Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής
Πρόγραμμα Μεταπτυχιακών Σπουδών
«Πληροφορική»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	**Red Teaming με χρήση του ARMITAGE και Metasploit.** **Red Teaming using ARMITAGE and Metasploit.**
Ονοματεπώνυμο Φοιτητή	**Τσιλίκας Νικόλαος**
Πατρώνυμο	**Χρήστος**
Αριθμός Μητρώου	**ΜΠΠΛ 19061**
Επιβλέπων	**Κωνσταντίνος Πατσάκης, Αναπληρωτής Καθηγητής**

Ημερομηνία Παράδοσης **Ιούνιος 2023**

Τριμελής Εξεταστική Επιτροπή

Κωνσταντίνος Πατσάκης	Ευθύμιος Αλέπης	Ευάγγελος Σακκόπουλος
Αναπληρωτής Καθηγητής	Αναπληρωτής Καθηγητής	Αναπληρωτής Καθηγητής

Πίνακας περιεχομένων

Πίνακας Εικόνων

Περίληψη

Σε αυτή την πτυχιακή εργασία θα εξεταστούν οι λειτουργίες, τα βασικά συστατικά και η διαχείριση του εργαλείου Armitage, ενός γραφικού περιβάλλοντος για το Metasploit Framework, το οποίο χρησιμοποιείται κυρίως σε δοκιμές διείσδυσης (penetration testing) και εκτίμησης τρωτότητας (vulnerability assessment) υπολογιστικών συστημάτων. Επίσης θα δοθούν παραδείγματα, για το πώς χρησιμοποιούνται τα παραπάνω σε επιχειρήσεις Red Teaming, χρησιμοποιώντας το Armitage.

Με τους όρους «έλεγχος ασφαλείας», «δοκιμές διείσδυσης» και «εκτίμηση τρωτότητας», εννοείται η εξέταση και δοκιμή των μέτρων και της πολιτικής ασφαλείας ενός δικτύου ή ενός συστήματος κάποιου οργανισμού και η δυνατότητα του να αντιμετωπίζει και να ανταπεξέρχεται σε τυχόν απειλές. Επίσης, εφόσον υπάρχουν κενά ασφαλείας, πως μπορεί κάποιος να τα εκμεταλλευτεί με την πρόθεση να προκαλέσει σοβαρά προβλήματα στο σύστημα.

Αρχικά, παρουσιάζεται η θεωρητική ανάλυση της διαδικασίας δοκιμών διείσδυσης συστημάτων, καθώς και οι λόγοι για τους οποίους κρίνεται όλο και περισσότερο αναγκαία με την πάροδο των χρόνων. Στη συνέχεια περιγράφεται η λειτουργία και οι δυνατότητες του Armitage καθώς και του Metasploit Framework. Έπειτα θα παρουσιαστούν κάποιες τεχνικές και μεθοδολογίες που χρησιμοποιούνται από τις Red Teams, με σκοπό να εξάγουν τις ζητούμενες πληροφορίες, σχετικές με την ασφάλεια, κάνοντας χρήση του Armitage και του Metasploit Framework σε ένα σύστημα με λειτουργικό σύστημα Metasploitable 2 καθώς και σε ένα σύστημα με Windows 10. Τέλος, παρουσιάζονται θεωρητικά οι σύγχρονες τεχνικές επίθεσης και επίτευξης μη ανιχνευσιμότητας, όπως η κοινωνική μηχανική (Social Engineering), οι Advanced Persistent Threats (APT) και οι τεχνικές Antivirus Evasion, καθώς επίσης και τεχνικές εντοπισμού και αντιμετώπισης αυτών των απειλών, όπως τα λογισμικά Endpoint Detection and Response (EDR).

1. Εισαγωγή

Σε αυτό το κεφάλαιο θα πραγματοποιηθεί μια θεωρητική ανάλυση της διαδικασίας δοκιμών διείσδυσης συστημάτων και θα παρουσιαστούν οι λόγοι για τους οποίους κρίνεται όλο και περισσότερο αναγκαία με την πάροδο των χρόνων.

1-1. Ορισμοί

Δοκιμή διείσδυσης θεωρείται μια διαδικασία κατά την οποία εξετάζεται το επίπεδο ασφάλειας ενός συστήματος και αναζητούνται ευπάθειες σε αυτό. Ο αναλυτής σε αυτή την διαδικασία αναλαμβάνει το ρόλο του επιτιθέμενου, κακόβουλου χρήστη, με σκοπό να αποδείξει ότι είναι εφικτό να προκαλέσει τη μεγαλύτερη δυνατή ζημιά στο σύστημα. Έτσι λοιπόν με αυτόν τον τρόπο ξεχωρίζουν τα μέρη του συστήματος τα οποία χρειάζονται ενίσχυση και περαιτέρω προστασία. Βέβαια στα πλαίσια αυτής της διαδικασίας είναι απαραίτητο να υπάρχει η συγκατάθεση της οντότητας (οργανισμού, χρήστη κτλ.) του οποίου το σύστημα εξετάζεται. Ακόμη είναι σημαντικό ο αναλυτής, κατά τη διάρκεια της έρευνάς του, να μην προκαλέσει πραγματική βλάβη στο σύστημα, μόνο να αποδείξει πως κάτι τέτοιο είναι δυνατό.

1-2. Αναγκαιότητα - Χρησιμότητα

Όσο περνούσαν τα χρόνια, με την ραγδαία εξέλιξη του διαδικτύου και με την είσοδο όλο και περισσότερων συστημάτων σε αυτό, οι κίνδυνοι όσον αφορά στην ασφάλεια των δεδομένων πλήθαιναν. Ο ρυθμός αύξησης των διαδικτυακών επιθέσεων που στόχευαν στην ακεραιότητα, την εμπιστευτικότητα και/ή την λειτουργικότητα των δεδομένων ήταν εκθετικός. Παράλληλα άρχισαν να φαίνονται και οι οικονομικές επιπτώσεις των εξελίξεων στις εταιρίες και τους οργανισμούς, που είχαν σαν βάση των δραστηριοτήτων τους τα δεδομένα. Έτσι λοιπόν παρουσιάστηκε επιτακτική ανάγκη για βελτίωση της ασφάλειας των συστημάτων και με αργούς ρυθμούς, άρχισε να υιοθετείται η έννοια της δοκιμής διείσδυσης ως την κύρια διαδικασία ελέγχου και βελτίωσης της ασφάλειας των υπολογιστικών υποδομών μιας οντότητας.

Παρότι μεγάλοι οργανισμοί είχαν αποδεχτεί την διαδικασία αυτή, άλλες επιχειρήσεις έδειχναν μια άρνηση προς την αποδοχή των δοκιμών διείσδυσης. Λόγω αυτού λοιπόν, αλλά και για την ενημέρωση των αναπτυσσόμενων επιχειρήσεων που δεν έχουν ως κεντρικό πυλώνα της οικονομίας τους τα δεδομένα αλλά τα χρησιμοποιούν για την διευκόλυνση των πελατών τους και γενικότερα του ευρέος κοινού, άρχισαν να βγαίνουν πρότυπα από τους μεγαλύτερους οργανισμούς ως προς το γιατί είναι αναγκαίο στη σημερινή εποχή για μια επιχείρηση να ρίξει βάρος στην ασφάλεια των δεδομένων. Μετά από έρευνα βρέθηκε το πρότυπο που δημοσίευσε ο οργανισμός SANS (SANS, 2006) (SANS, 2002) (PCI SSC, 2015), το οποίο θεωρείται και το πιο διαδεδομένο.

Σε αυτό το πρότυπο αναφέρεται πως οποιοδήποτε υπολογιστικό σύστημα ή λογισμικό είχε, έχει και θα έχει κενά ασφαλείας. Το γεγονός όμως πως δεν θα μπορέσει ποτέ να γίνει απαραβίαστο δεν υποβαθμίζει, σε καμία περίπτωση, την αναγκαιότητα των δοκιμών διείσδυσης αλλά, αντιθέτως, την ενισχύει. Ακόμη κατηγοριοποιεί τα οφέλη μιας δοκιμής διείσδυσης ως εξής:

- Ανακάλυψη προβλημάτων ασφαλείας σε μια υπολογιστική υποδομή, πριν από κάποιον κακόβουλο χρήστη. Με αυτόν τον τρόπο επιτυγχάνεται η άμυνα του συστήματος, κατά την ανάπτυξή του αφήνοντάς το ευάλωτο κατά το λιγότερο δυνατό τρόπο.
- Αναφορά προβλημάτων στον προϊστάμενο ασφαλείας για δημιουργία μελλοντικού πλάνου κατεύθυνσης. Το πλάνο αυτό σχεδιάζεται βάση παραγόντων οι οποίοι αναλύονται λεπτομερώς στο επόμενο κεφάλαιο.
- Επιβεβαίωση των ρυθμίσεων των συστημάτων που είχαν ως αρχικό στόχο την διατήρηση της ασφάλειας. Με αυτόν τον τρόπο γίνεται όχι μόνο αξιολόγηση της ασφάλειας, αλλά και του προσωπικού που είναι υπεύθυνο για αυτήν.
- Βελτίωση και/ή εκπαίδευση του προσωπικού σε σχέση με περιστατικά ασφαλείας. Αυτό γίνεται κυρίως σε οργανισμούς με μεγάλο ανθρώπινο δυναμικό.
- Εύρεση κενών ασφαλείας που δημιουργήθηκαν από άγνοια ή συμβιβασμό. Αυτά συνήθως προκύπτουν από προγραμματιστικά λάθη ή από εκτελεστική επιλογή για την μείωση του κόστους.
- Έλεγχος νέων τεχνολογιών/λογισμικών προϊόντων της επιχείρησης πριν αυτά βγουν στην παραγωγή. Με αυτόν τον τρόπο προβλέπονται καταστροφικές καταστάσεις, βελτιώνεται το κύρος του οργανισμού και μειώνεται το κόστος για μελλοντική συντήρηση και η ανάγκη για δημιουργία επέκτασης του

λογισμικού (patch).

1-3. Έλεγχος Διείσδυσης Penetration Testing.

Ο έλεγχος διείσδυσης (penetration testing) είναι η μέθοδος αξιολόγησης ασφάλειας Πληροφοριακών Συστημάτων (Η/Υ), προσομοιώνοντας κυβερνοεπιθέσεις από "κακόβουλους" χρήστες. Αφορά επίθεση για απόκτηση πρόσβασης σε υπηρεσίες, δεδομένα ή συστήματα χωρίς διαπιστευτήρια (username/password). Αν η εστίαση αυτού του ελέγχου είναι σε κάποιο πληροφοριακό σύστημα, τότε μια επιτυχής διείσδυση, συνοδεύεται από την απόκτηση πρόσβασης σε εμπιστευτικές πληροφορίες, όπως έγγραφα και βάσεις δεδομένων. Ποια είναι όμως η ειδοποιός διαφορά μεταξύ ενός κακόβουλου χρήστη από έναν Penetration Tester; Η άδεια του εκάστοτε ιδιοκτήτη ενός πληροφοριακού συστήματος το οποίο βρίσκεται υπό έλεγχο, επίσης οι ελεγκτές είναι υπεύθυνοι να παραδώσουν μια ενδελεχή έκθεση για τα ευρήματα τους. Αυτό έχει ως σκοπό την αύξηση του επιπέδου ασφαλείας του συστήματος που δοκιμάζεται. Σε κάποιες περιπτώσεις ο pen-tester έχει πρόσβαση στο σύστημα σαν απλός χρήστης με περιορισμένες δυνατότητες , ο στόχος σε αυτές τις περιπτώσεις είναι να επιτύχει την αύξηση των δικαιωμάτων του, και ως εκ τούτου να αποκτήσει πρόσβαση σε δεδομένα που κανονικά δεν έχει εξουσιοδότηση. Συνήθως οι ελεγκτές δεν σταματούν με την εύρεση του πρώτου τρωτού σημείου στο σύστημα, αλλά συνεχίζουν να ψάχνουν για ευπάθειες στο σύστημα. Είναι σημαντικό για έναν ελεγκτή διείσδυσης να κρατά αναλυτικές σημειώσεις κατά την διάρκεια του ελέγχου, ούτως ώστε τα ευρήματά του να επαληθευτούν και να επιδιορθωθούν. Οι εκάστοτε οργανισμοί, πρέπει να έχουν κατά νουν ότι είναι σχεδόν αδύνατον να αναγνωριστούν όλες οι ευπάθειες στο σύστημα από έναν έλεγχο. Για παράδειγμα, μετά το τέλος ενός ελέγχου, κάποια εταιρεία μπορεί να βγάλει μια αναβάθμιση στο λογισμικό της μετά το πέρας του ελέγχου, και ίσως αυτή να είναι τρωτή σε κάποια επίθεση, και ένα μήνα μετά κάποιος άλλος τρίτος να δώσει κάποια τρωτή αναβάθμιση στα λογισμικά που χρησιμοποιούνται. Η διατήρηση ενός ασφαλούς δικτύου απαιτεί συνεχή επαγρύπνηση.

1-4. Φάσεις ενός ελέγχου διείσδυσης (Penetration Testing)

Προετοιμασία (Pre-engagement).

Ο έλεγχος αρχίζει με αυτή τη φάση, η οποία περιλαμβάνει συζητήσεις με τον πελάτη ούτως ώστε οι δύο πλευρές να βρίσκονται στην ίδια σελίδα σχετικά με τον έλεγχο. Μια παρεξήγηση μεταξύ του δοκιμαστή και του πελάτη, ο οποίος επιζητά μια απλή σάρωση ευπαθειών μπορεί να οδηγήσει σε μια δύσκολη κατάσταση, γιατί οι έλεγχοι διεισδύσεως είναι πολύ ποιο έντονοι όπως έχει προαναφερθεί.

Στην παρούσα φάση, ο εκάστοτε pen-tester πρέπει να κατανοήσει τους εταιρικούς στόχους του πελάτη για την επερχόμενη διαδικασία διείσδυσης.

Άλλα σημαντικά θέματα που πρέπει να αποφασιστούν από κοινού πριν αρχίσει ο πραγματικός έλεγχος είναι τα ακόλουθα:

- **Πεδίο Δράσης:** Ποιές διευθύνσεις IP ή κεντρικοί υπολογιστές (hosts) είναι εντός του πεδίου δράσης και ποιοι όχι. Ποιες ενέργειες θα επιτρέψει στον pen-tester ο πελάτης; Επιτρέπεται η χρήση λογισμικών εκμετάλλευσης ευπαθειών (exploits), τα οποία πιθανόν να οδηγήσουν στην κατάρρευση κάποιας υπηρεσίας; Ή πρέπει να περιοριστεί στην απλή αναγνώριση των ευπαθειών; Ο πελάτης κατανοεί ότι, μια απλή σάρωση για ανοικτές θύρες στο δίκτυο, μπορεί να οδηγήσει στον τερματισμό της λειτουργίας κάποιου διακομιστή ή δρομολογητή; Το social-engineering είναι μια επιλογή;
- **Παράθυρο Ελέγχου:** Η εταιρία ίσως θελήσει οι έλεγχοι να εκτελούνται σε συγκεκριμένες ώρες ή ημέρες.
- **Επικοινωνία**: Με ποιον πρέπει να έρθει σε επαφή ο pen-tester αν ανακαλύψει κάτι κρίσιμο; Ο πελάτης θα έχει κάποιον σε αναμονή επι εικοσιτετραώρου βάσεως; Προτιμά την χρήση κρυπτογραφημένης ηλεκτρονικής αλληλογραφίας;

- **Μια κάρτα "get out of jail free":** Ο pen-tester πρέπει να είναι σίγουρος ότι είναι εξουσιοδοτημένος να εκτελέσει έναν έλεγχο σε κάποιο στόχο. Αν αυτός δεν ανήκει στον πελάτη (π.χ. κάποια υπηρεσία φιλοξενείται (hosted) από κάποιον τρίτο), ο πελάτης πρέπει να έχει επίσημη έγκριση από το τρίτο μέρος για να εκτελέσει το penetration test. Ανεξαρτήτως αυτού, ο pen-tester πρέπει να είναι σίγουρος ότι στο συμβόλαιο υπάρχει μια δήλωση η οποία περιορίζει την ευθύνη του σε περίπτωση που συμβεί κάτι απροσδόκητο, και να πάρει γραπτή έγκριση για να εκτελέσει τον έλεγχο.
- **Όροι πληρωμής**: Πως και πότε θα πληρωθεί, και πόσο;

Τέλος, στο συμβόλαιο πρέπει να υπάρχει μια ρήτρα Μη-Δημοσιοποίησης των ευρημάτων. Ο πελάτης θα εκτιμήσει ιδιαίτερα την γραπτή δέσμευση του pen- tester για εμπιστευτικότητα.

Συλλογή πληροφοριών (Information Gathering)

Είναι η πρώτη φάση του πραγματικού ελέγχου. Κατά την διάρκεια του, ο pen-tester είναι ελεύθερος να αναλύσει διαθέσιμες πηγές πληροφοριών, μια διαδικασία γνωστή ως open source intelligent (OSINT). Επίσης, εδώ αρχίζει η χρήση εργαλείων όπως σαρωτές θυρών (port scanners) για να πάρει μια ιδέα για τα συστήματα που υπάρχουν στο διαδίκτυο ή στο εσωτερικό εταιρικό δίκτυο καθώς επίσης και τι λογισμικά τρέχουν σε αυτά.

Μοντελοποίηση Απειλής (Threat Modeling)

Εφοδιασμένος με τις γνώσεις που απόκτησε κατά την προηγούμενη φάση ο pen- tester προχωρά στο Threat Modeling. Εδώ λειτουργεί-σκέφτεται ως κακόβουλος χρήστης και αναπτύσσει το πλάνο της επίθεσης βασισμένο στις πληροφορίες που μάζεψε προηγουμένως. Για παράδειγμα, αν ο πελάτης αναπτύσσει ιδιόκτητα λογισμικά, ένας επιτιθέμενος μπορεί να καταστρέψει τον οργανισμό αποκτώντας πρόσβαση στο εσωτερικό σύστημα ανάπτυξης λογισμικών, το σημείο στο οποίο αναπτύσσεται και δοκιμάζεται ο πηγαίος κώδικας , στην συνέχεια μπορεί να πωλήσει αυτά τα εταιρικά μυστικά σε κάποιον ανταγωνιστή. Εν κατακλείδι, σε αυτή την φάση ο pen-tesrter αναπτύσσει πλάνα και στρατηγικές για να διεισδύσει στο σύστημα του πελάτη.

Ανάλυση συστήματος για εντοπισμό αδυναμιών/ευπαθειών (Vulnerabilities)

Στη συνέχεια, ο pentester αρχίζει την ενεργή ανεύρεση ευπαθειών για να προσδιορίσει πόσο επιτυχής θα είναι η στρατηγική εκμετάλλευσης ευπαθειών του. Η αποτυχία ενός exploit μπορεί να οδηγήσει στην κατάρρευση κάποιας υπηρεσίας, να ενεργοποίηση μηχανισμούς ανίχνευσης εισβολών, ή αλλιώς να καταστρέψει τις ευκαιρίες του για μια επιτυχημένη εκμετάλλευση των ευπαθειών (exploitation). Συχνά κατά την διάρκεια αυτής της φάσης, ο ελεγκτής τρέχει σαρωτές ευπαθειών (vulnerability scanners), οι οποίοι χρησιμοποιούν διάφορες βάσεις δεδομένων ευπαθειών και μια σειρά από ενεργούς ελέγχους για να κάνουν την καλύτερη πρόβλεψη σχετικά με τί ευπάθεια υπάρχει, αν υπάρχει, στο σύστημα του πελάτη. Αλλά, παρότι αυτοί οι σαρωτές είναι ισχυρά εργαλεία, δεν μπορούν να αντικαταστήσουν την κριτική σκέψη, άρα πρέπει να εκτελεστούν και μη αυτοματοποιημένες αναλύσεις για να επιβεβαιωθούν τα αποτελέσματα των σαρωτών απότον ίδιο τον pen-tester.

Εκμετάλλευση (Exploitation) αδυναμιών/ευπαθειών

Στην παρούσα φάση, εκτελούνται τα κατάλληλα λογισμικά εκμετάλλευσης ευπαθειών (exploits) κατά των αναγνωρισμένων ευπαθειών (χρησιμοποιώντας εργαλεία όπως το Metasploit και το Armitage), ο pentester προσπαθεί να αποκτήσει πρόσβαση στο σύστημα.

Post Exploitation

Κάποιοι λένε ότι ο έλεγχος αρχίζει πραγματικά μετά την φάση του explotation, σε αυτήν εδώ την φάση. Εισέβαλε στο σύστημα, αλλά τι πραγματικά σημαίνει αυτό για τον οργανισμό; Αν ο pentester εισέβαλε σε ένα απαρχαιωμένο μη ενημερωμένο σύστημα το οποίο δεν είναι μέρος του τομέα (domain) ή αλλιώς, δικτυωμένο με στόχους υψηλής αξίας, και αυτό το σύστημα δεν περιέχει καμία χρήσιμη πληροφορία σε κάποιον επιτιθέμενο, το ρίσκο αυτής της ευπάθειας είναι πάρα πολύ χαμηλότερο από το αν ο pentester εκμεταλλευτεί μια ευπάθεια σε έναν domain controller ή στο σύστημα ανάπτυξης του οργανισμού.

Κατά την διάρκεια αυτής της φάσης, οι pentesters μαζεύουν πληροφορίες σχετικά με το σύστημα που επιτέθηκαν, ψάχνουν για ενδιαφέροντα αρχεία, προσπαθούν να αυξήσουν τα δικαιώματά τους σε αυτό, όπου είναι αναγκαίο κ.ο.κ. Για παράδειγμα, μπορεί να πάρουν μια λίστα με κατακερματισμένους (hashed) κωδικούς για να δουν αν μπορούν να τους ανακτήσουν, ή να τους χρησιμοποιήσουν για να αποκτήσουν πρόσβαση σε επιπλέον συστήματα της εταιρίας. Επίσης ίσως

χρησιμοποιήσουν το σύστημα που πλέον ελέγχουν για να επιτεθούν σε κάποιο άλλο που πριν δεν ήταν διαθέσιμο.

Αναφορά (Reporting)

Η τελική φάση του ελέγχου διείσδυσης είναι η αναφορά. Εδώ μεταβιβάζονται τα ευρήματα στον πελάτη με τρόπο κατανοητό σε αυτόν. Ενημερώνονται σε ποια σημεία είναι σωστοί, και που υπάρχει ανάγκη για βελτιώσεις στην ασφάλεια, πως εισέβαλε στο σύστημα, τι βρήκε σε αυτό, πως θα κλείσει το κενό ασφάλειας κ.α.

Η συγγραφή μιας καλής αναφοράς του ελέγχου είναι μια τέχνη που χρειάζεται πολύ εμπειρία για να τελειοποιηθεί. Χρειάζεται τα ευρήματα να μεταφερθούν σαφέστατα σε όλους τους ενδιαφερομένους, από τον επικεφαλής και την ομάδα του τμήματος πληροφοριών που θα επιδιορθώσουν τα κενά, μέχρι τα ανώτερα στρώματα της διοίκησης που θα υπογράψουν σχετικά με τις αλλαγές. Ένας τρόπος για να επικοινωνήσει μαζί του είναι να του παρουσιάσει τα προσωπικά δεδομένα στα οποία κατάφερε να αποκτήσει πρόσβαση μέσω αυτού. Η αναφορά πρέπει να περιλαμβάνει μια τεχνική έκθεση καθώς επίσης και μια περίληψη των κυριότερων σημείων.

- **Περίληψη των κυριότερων σημείων**: Περιγράφει τους στόχους του ελέγχου και προσφέρει μια επισκόπηση υψηλού επιπέδου των πορισμάτων. Προορίζεται γιατους επικεφαλής του οργανισμού.
- **Τεχνική έκθεση:** Σε αυτό το κομμάτι της αναφοράς προσφέρεται μια πιο αναλυτική αναφορά με περισσότερα τεχνικά στοιχεία. Αυτή προορίζεται για τοΤμήμα Πληροφοριών.

2. Εργαλεία λογισμικού δοκιμών διείσδυσης Metasploit και Armitage.

Στο κεφάλαιο αυτό θα μελετηθούν τα πιο διαδεδομένα εργαλεία δοκιμών διείσδυσης, το Metasploit Framework καθώς και το Armitage. Θα γίνει αναφορά στην σκοπιμότητα και στις λειτουργίες τους, καθώς και στους λόγους για τους οποίους θεωρούνται τα επικρατέστερα. Η ανάλυση αυτή θα γίνει σε θεωρητικό επίπεδο, καθώς ο τρόπος χρήσης τους σε πρακτικό επίπεδο μελετάται σε επόμενο κεφάλαιο.

2-1. Metasploit Framework

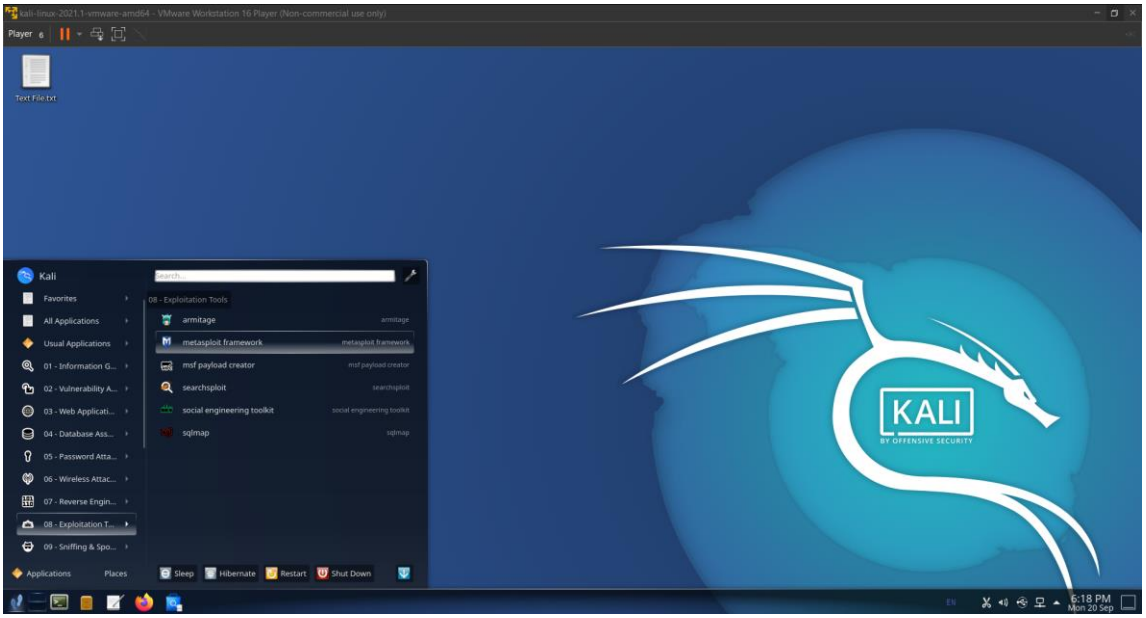
Το Metasploit Framework είναι ίσως και το πιο διαδεδομένο εργαλείο δοκιμών διείσδυσης που υπάρχει στις μέρες μας. Είναι ένα πολυεργαλείο που χρησιμοποιείται από αναλυτές ασφαλείας, με σκοπό τη συλλογή πληροφοριών για ευπάθειες σε, απομακρυσμένα ή τοπικά, συστήματα-στόχους και για την εκμετάλλευσή τους με την δημιουργία κατάλληλου φορτίου κώδικα (payload). Δημιουργήθηκε εν έτει 2003 από τον H. D. Moore, έναν αναλυτή ασφαλείας, και ξεκίνησε ως ένα έργο ανοιχτού κώδικα. Στην συνέχεια αποκτήθηκε από την εταιρία ασφαλείας Rapid7, η οποία συνέχισε να παρέχει μια βασική έκδοση του προγράμματος δωρεάν, ενώ δημιούργησε και μια πιο εξειδικευμένη, επί πληρωμή έκδοση, το Metasploit Pro. Αυτή τη στιγμή είναι διαθέσιμο για τα λειτουργικά συστήματα Linux και Windows.

Οι λειτουργίες του εργαλείου για την επίτευξη των στόχων του είναι αμέτρητες. Οι πιο σημαντικές είναι οι εξής:

- Διατηρεί μια βάση δεδομένων με ευπάθειες σε παλαιότερες εκδόσεις προγραμμάτων και λειτουργικών συστημάτων, μέσω των οποίων μπορεί να αναλύσει τον εκάστοτε στόχο μιας επίθεσης.
- Για τις ευπάθειες αυτές μπορεί αυτοματοποιημένα να παράγει φορτίο κώδικα (payload/shellcode) με την δυνατότητα να εκμεταλλευτεί την εκάστοτε ευπάθεια.
- Δίνει τη δυνατότητα σε ερευνητές να παράγουν φορτίο κώδικα για να εκμεταλλευτούν και μη γνωστές ευπάθειες.
- Ελέγχει όλα τα γνωστά επίπεδα ενός στόχου (δικτύωση, κρυπτογραφία, προγραμματιστικά λάθη κοκ).

Είναι ακόμη αξιοσημείωτο ότι κάνει χρήση τεχνολογιών και αλγορίθμων, έτσι ώστε τα φορτία κώδικα που παράγει να είναι μη ανιχνεύσιμα και περίπλοκα στην ανάλυση της λειτουργικότητάς τους, σε περίπτωση ανίχνευσης.

Στο περιβάλλον του Kali Linux το Metasploit Framework εντοπίζεται στην κατηγορία Exploitation Tools του μενού εφαρμογών.



Εικόνα 1. Metasploit στο Kali Linux

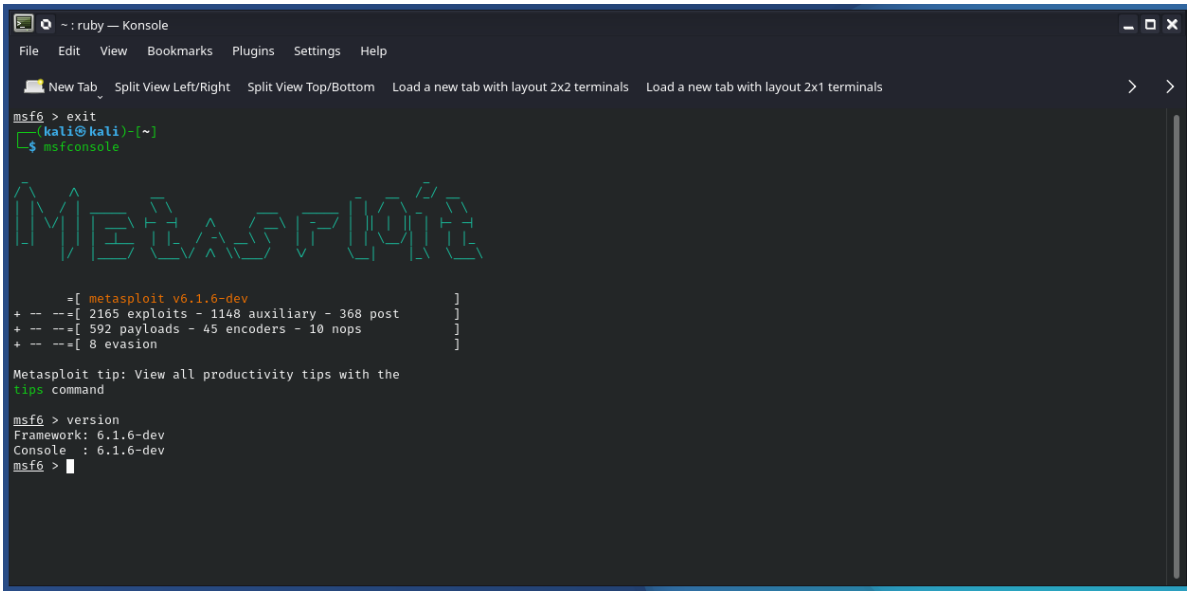
Με την έναρξή του, ο ερευνητής έχει πρόσβαση στη γραμμή εντολών του Metasploit και η διαχείρισή του είναι εφικτή μέσα από τις πολυπληθείς εντολές που υποστηρίζει αυτή. Η επεξήγηση όλων αυτών των εντολών ξεφεύγει από τα πλαίσια αυτής της εργασίας, ωστόσο θα παρουσιαστούν οι πιο βασικές για τον χειρισμό του εργαλείου.

- **help** - Τυπώνονται όλες οι διαθέσιμες εντολές που παρέχει το εργαλείο. Σχηματίζοντας τον συνδυασμό στην μορφή 'help command' τυπώνονται περισσότερες πληροφορίες για την συγκεκριμένη εντολή (command).
- **search** - Με αυτήν την εντολή παρέχεται η δυνατότητα αναζήτησης κώδικα για την εκμετάλλευση αδυναμιών που ήδη έχει στην βάση δεδομένων του το Metasploit. Η αναζήτηση μπορεί να γίνει βάσει κάποιων χαρακτηριστικών, όπως ο τύπος λειτουργικού συστήματος του στόχου, το όνομα του ευπαθούς προγράμματος, ο μοναδικός αναγνωριστικός κωδικός που χαρακτηρίζονται οι αδυναμίες, καθώς και ο συνδυασμός όλων μαζί.
- **db_import** - Εισάγει τα αποτελέσματα μιας σάρωσης δικτύου ή αδυναμιών από εργαλεία όπως το nMap, στην τοπική βάση δεδομένων, ώστε να γίνει η ανάλυσή τους για πιθανά κενά ασφαλείας, τα οποία μπορεί να αναγνωρίσει το Metasploit χρησιμοποιώντας τα δεδομένα που ήδη έχει ενσωματωμένα για αδυναμίες και την εκμετάλλευση αυτών.
- **services** - Τυπώνει όλες τις πληροφορίες που διαθέτει στη βάση δεδομένων για τις υπηρεσίες και τους υπολογιστές που έχουν προηγουμένως εισαχθεί από κάποιο υποστηριζόμενο εργαλείο σαρωτή.
- **use** - Δηλώνεται ποιο άρθρωμα (module) του Metasploit θα χρησιμοποιηθεί. Χαρακτηριστικό παράδειγμα τέτοιου αρθρώματος είναι κώδικας που διαθέτει το εργαλείο για την εκμετάλλευση συγκεκριμένης αδυναμίας. Συνήθως τα αρθρώματα χρειάζονται κάποιου είδους ρύθμιση, ώστε να ανταποκρίνονται στις εκάστοτε περιπτώσεις χρήσης τους.

Οι εντολές που χρησιμοποιούνται για την παραμετροποίηση ενός αρθρώματος είναι οι:

- **show options** - Δείχνει τις παραμέτρους που πρέπει να ρυθμιστούν.
- **set** - Θέτει την τιμή μιας παραμέτρου.

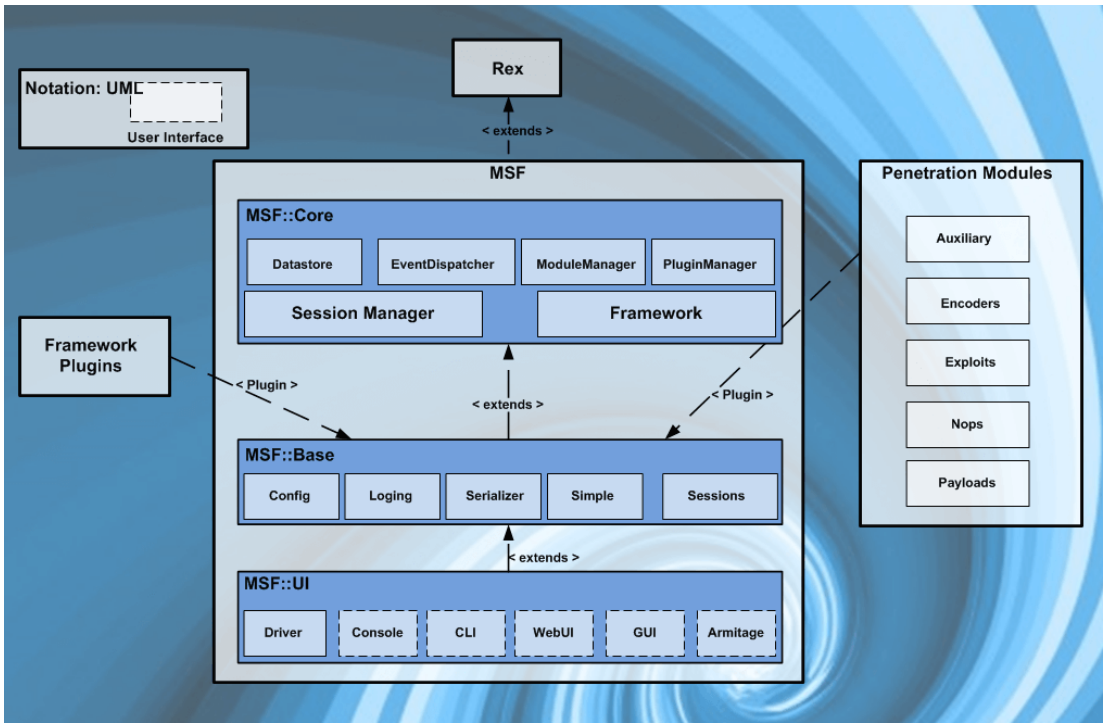
Χρησιμοποιώντας τις παραπάνω εντολές και πληροφορίες, ο ερευνητής μπορεί να αναζητήσει στο διαδίκτυο αν υπάρχει κάποια γνωστή αδυναμία στις υπηρεσίες και τις εφαρμογές που εμφανίζονται στο δίκτυο που ερευνά. Όσον αφορά στις γνωστές αδυναμίες, το εργαλείο αυτό παρέχει την δυνατότητα εκμετάλλευσής τους ενσωματωμένη σε αυτό, χωρίς να χρειάζεται κάποιος ερευνητής ασφαλείας να αναζητήσει στο διαδίκτυο για προγράμματα που προσφέρουν αυτή τη λειτουργία.



Εικόνα 2. Περιβάλλον Metasploit

Η παρούσα έκδοση είναι η 6.1.6-dev και όπως φαίνεται στην Εικόνα 2, περιλαμβάνει 2165 exploits, 1148 auxiliary, 368 post, 592 payloads, 45 encoders, 10 nops και 8 evasion.

Το Metasploit είναι γραμμένο σε Ruby και βρίσκεται σε εξέλιξη για πολλά χρόνια. Με την πρώτη ματιά, το μέγεθος του έργου μπορεί να είναι τρομακτικό, αλλά σπάνια θα χρειαστεί να εμβαθύνουμε στην αρχιτεκτονική του, η οποία φαίνεται στην Εικόνα 3.



Εικόνα 3. Γράφημα Αρχιτεκτονικής Metasploit

2-1-1. Modules

Το Metasploit, όπως παρουσιάζεται στον χρήστη, αποτελείται από modules. Ένα module (μία μονάδα ή ενότητα), είναι ένα κομμάτι λογισμικού που μπορεί να χρησιμοποιηθεί από το Metasploit Framework. Μερικές φορές ίσως χρειάζεται ένα exploit module, δηλαδή ένα τμήμα λογισμικού το οποίο πραγματοποιεί μία επίθεση και ορίζεται ως ένα module που χρησιμοποιεί payloads. Άλλες φορές ένα auxiliary module, το οποίο ορίζεται ως ένα exploit χωρίς payload, μπορεί να χρειαστεί για σάρωση ή απαρίθμηση ενός συστήματος. Αυτά τα εναλλάξιμα δομοστοιχεία αποτελούν τον πυρήνα που κάνει το πλαίσιο τόσο ισχυρό.

Σχεδόν όλη η αλληλεπίδραση με το Metasploit γίνεται μέσω των πολλών modules, τα οποία αναζητά σε δύο τοποθεσίες. Οι τοποθεσίες αυτών των modules είναι:

Κύριο «δέντρο» των modules:

- **/usr/share/metasploit-framework/modules/**

User-Specified Module «δέντρο»:

- **~/.msf4/modules/**

Εκεί μπορεί ο tester να αποθηκεύει τα δικά του module sets.

Exploits

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 overflows, 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 active exploits θα εκμεταλλευτούν έναν συγκεκριμένο υπολογιστή, θα τρέξουν μέχρι να ολοκληρωθούν και μετά θα σταματήσουν. Έχουν επίσης τη δυνατότητα να τρέχουν στο παρασκήνιο, 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.

Auxiliary

Τα auxiliary modules περιλαμβάνουν port scanners, fuzzers, sniffers, και άλλα.

Payloads

Το payload (ωφέλιμο φορτίο ελληνιστί), 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» (to bind, εξ ου και το όνομα) 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.

Υπάρχουν τρία διαφορετικά είδη payload modules στο Metasploit, τα Singles, Stagers και Stages και ξεχωρίζουν από το πόσες καθέτους «/» έχουν στον τίτλο τους. Δηλαδή, το «windows/shell_bind_tcp» είναι single payload χωρίς stage (στάδιο), ενώ το «windows/shell/bind_tcp» έχει ένα Stager (bind_tcp) και ένα stage (shell).

Τα Singles, είναι ανεξάρτητα και αυτοτελή payloads, μπορεί δηλαδή να είναι κάτι τόσο απλό όσο να προστεθεί ένας χρήστης στο δίκτυο-στόχος ή να εκτελεστεί ένα αρχείο (calc.exe). Τα Stagers εγκαθιδρύουν μια σύνδεση μεταξύ του επιτιθέμενου και του θύματος και είναι σχεδιασμένα κυρίως να έχουν μικρό μέγεθος και να είναι αξιόπιστα. Επειδή όμως αυτό είναι δύσκολο, υπάρχουν πολλοί παρόμοιοι Stagers. Τα Stages είναι συστατικά payload που «καταβάζονται» από τις μονάδες (modules) των Stagers. Τα πολλαπλά stages (στάδια) των payloads παρέχουν εξειδικευμένες λειτουργίες (features) χωρίς όρια μεγέθους, όπως το Meterpreter, το VNC Injection και πιο πρόσφατα το iPhone «ipwn» Shell.

Encoders

Οι Encoders διασφαλίζουν ότι τα ωφέλιμα φορτία φτάνουν στον προορισμό τους άθικτα. Ο παραγόμενος κώδικας κελύφους συνήθως μπορεί να είναι πλήρως λειτουργικός αλλά να έχει πολλούς null χαρακτήρες (x00s και xffs), έτσι όταν τον τρέχουν άλλα προγράμματα, αυτοί σηματοδοτούν το τέλος μιας συμβολοσειράς και αυτό προκαλεί τον κώδικα να τερματίζει πριν την ολοκλήρωση του, με αποτέλεσμα να καταστρέφει το payload (ωφέλιμο φορτίο). Επιπλέον, όταν ο κώδικας τρέχει μέσα στο δίκτυο απροκάλυπτα είναι πολύ πιθανόν να εντοπιστεί από προγράμματα ασφαλείας. Για αυτό το λόγο υπάρχουν οι encoders, οι οποίοι βοηθούν στην αποφυγή κακών ή μηδενικών (null) χαρακτήρων και του εντοπισμού από προγράμματα ασφαλείας (Intrusion Detect Systems – IDSs), κωδικοποιώντας έτσι το payload με τέτοιο τρόπο ώστε να μην περιλαμβάνει «κακούς» χαρακτήρες.

Το Metasploit προσφέρει ένα σύνολο διαφορετικών κωδικοποιητών για συγκεκριμένες περιπτώσεις. Ορισμένοι χρησιμοποιούνται για περιπτώσεις που μόνο αλφαριθμητικοί χαρακτήρες μπορούν να χρησιμοποιηθούν σαν payload, ή μόνο εκτυπώσιμοι χαρακτήρες επιτρέπονται σαν είσοδος, αλλά υπάρχουν και άλλοι που μπορούν να πετύχουν σχεδόν σε όλες τις περιπτώσεις. Βέβαια καθώς το πλαίσιο δεν μένει ποτέ στάσιμο, οι κωδικοποιητές αυτοί αλλάζουν και βγαίνουν συχνά νέοι και καλύτεροι.

Nops

Τα Nops διατηρούν τα μεγέθη ωφέλιμου φορτίου σταθερά κατά τις προσπάθειες εκμετάλλευσης.

2-1-2. Ορολογία

Έχοντας εξηγήσει όλα τα παραπάνω μένουν δύο ακόμα βασικά όροι τους οποίους θα πρέπει να γνωρίζει ένας tester και χρησιμοποιούνται συχνά στο Metasploit.

Shellcode

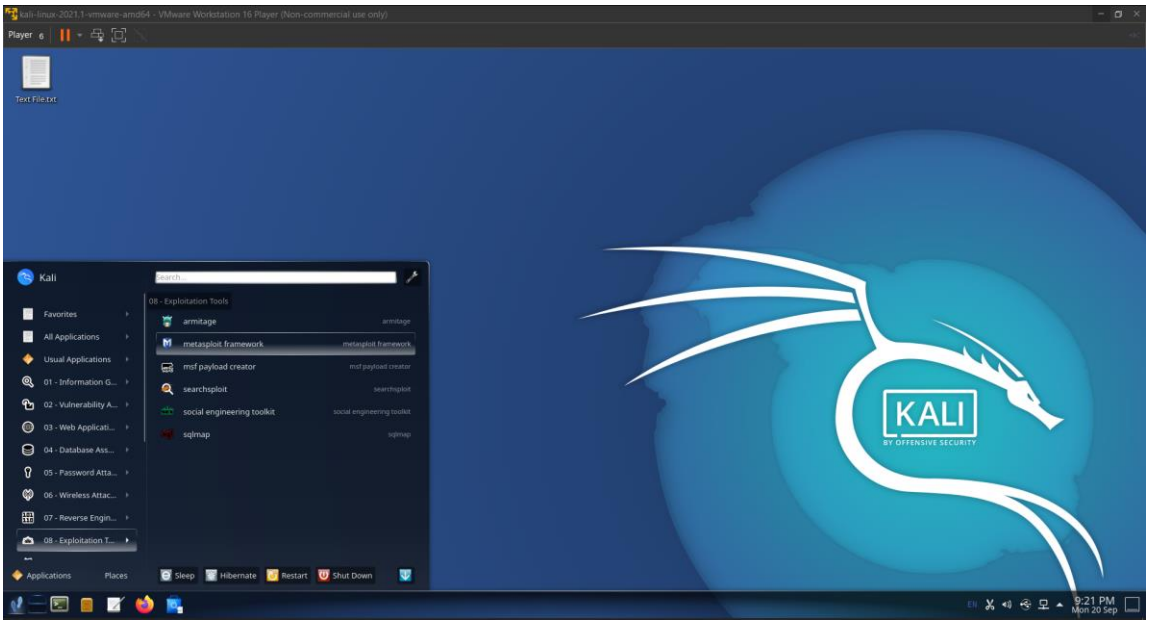
Shellcode είναι ένα σύνολο οδηγιών που χρησιμοποιούνται ως ένα ωφέλιμο φορτίο, όταν συμβαίνει η εκμετάλλευση. Συνήθως είναι γραμμένο σε assembly. Στις περισσότερες περιπτώσεις μία εντολή κελύφους (command shell) θα παρέχεται όταν η σειρά οδηγιών θα έχει διεξαχθεί από το μηχάνημα-στόχο.

Listener

Ένας listener (ακροατής) είναι ένα στοιχείο μέσα στο Metasploit που περιμένει για κάποιου είδους σύνδεση. Για παράδειγμα, αφού το μηχάνημα στόχος έχει γίνει exploit, μπορεί να καλέσει το επιτιθέμενο μηχάνημα μέσω διαδικτύου. Ο listener χειρίζεται αυτήν την σύνδεση, αναμένοντας το επιτιθέμενο μηχάνημα να έρθει σε επαφή με το υπονομευμένο σύστημα.

2-2. Armitage

Το Armitage είναι το GUI (Graphical User Interface) του Metasploit, δημιουργημένο από τον Raphael Mudge. Η διεπαφή είναι καλοφτιαγμένη, πλούσια σε λειτουργίες και το κυριότερο δωρεάν. Καθιστά το πλαίσιο πιο εύχρηστο στους συνηθισμένους σε γραφικά περιβάλλοντα, αλλά πριν χρησιμοποιηθεί πρέπει να υπάρχει μια βασική κατανόηση του τρόπου λειτουργίας του.



Εικόνα 4. Armitage στο Kali Linux

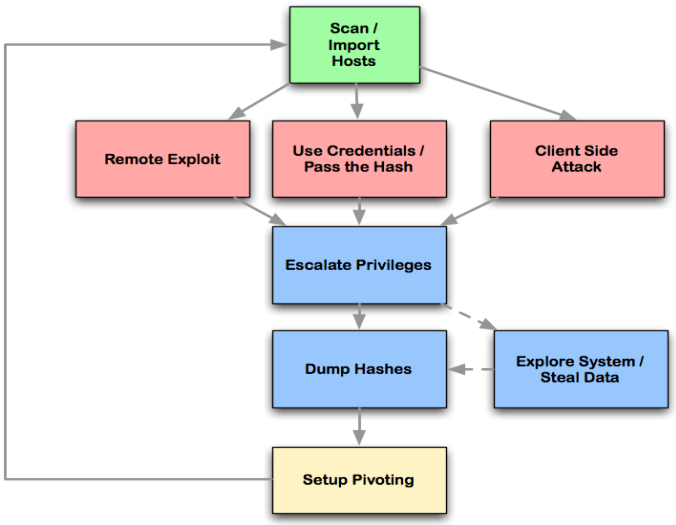
Χαρακτηρίζεται ως ένα εργαλείο συνεργασίας Red Team για Metasploit που απεικονίζει στόχους, προτείνει ευπάθειες προς εκμετάλλευση και διαθέτει τις προηγμένες post-exploitation δυνατότητες στο framework.

Μέσω μιας εκτέλεσης του Metasploit, η Red Team μπορεί:

- Να χρησιμοποιεί τις ίδιες συνεδρίες.
- Να μοιράζεται hosts, καταγεγραμμένα δεδομένα και ληφθέντα αρχεία.
- Να επικοινωνεί μέσω ενός κοινού αρχείου καταγραφής συμβάντων.
- Να εκτελεί bots για να αυτοματοποιήσει τις εργασίες της κόκκινης ομάδας.

2-2-1. Διαχείριση των επιθέσεων με το Armitage

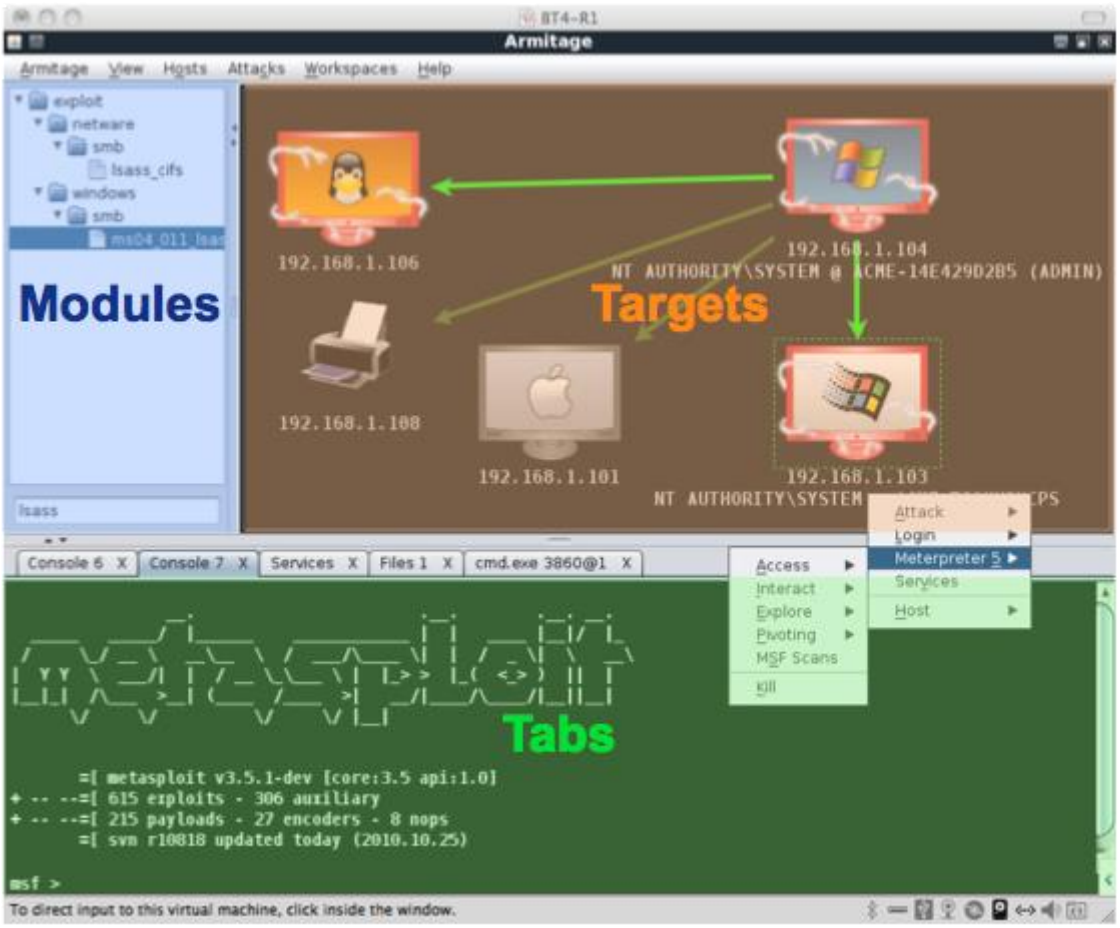
Το Armitage οργανώνει τις δυνατότητες του Metasploit γύρω από τη διαδικασία hacking. Υπάρχουν λειτουργίες για ανακάλυψη, πρόσβαση, post-exploitation και κίνηση μέσα στο δίκτυο.



Εικόνα 5. Cyber Attack Management

Κατά την λειτουργία του το περιβάλλον του Armitage μέσω του plug-in του Nmap στο Metasploit, προσφέρει στο χρήστη μια οπτικοποίηση των συστημάτων-στόχων, που έχουν σαρωθεί, προτείνει exploits με την βοήθεια των ενσωματωμένων auxiliary modules για την αναζήτηση ευπαθειών και χρησιμοποιεί το Meterpreter το οποίο, όταν εφαρμοστεί επιτυχώς ένα exploit, παρέχει επιλογές από post-modules απόκτησης δικαιωμάτων διαχειριστή, πρόσβασης σε password hash codes, ανοίγματος ενός terminal, κλπ.

Αξιοσημείωτο είναι το γεγονός ότι ακόμα και αν ο έλεγχος για την αποτελεσματικότητα των exploits αποτύχει, το Armitage υποστηρίζει την ονομαζόμενη “Hail Mary” επίθεση που εξαπολύει ένα αυτοματοποιημένο σύστημα εκμετάλλευσης εναντίον των στόχων. Βρίσκει exploits που σχετίζονται με τους στόχους, φιλτράρει τα exploits χρησιμοποιώντας γνωστές πληροφορίες και, στη συνέχεια, τις ταξινομεί σε μια βέλτιστη σειρά. Αυτή η δυνατότητα δεν θα βρει κάθε πιθανό shell, αλλά είναι μια καλή επιλογή αν δεν γνωρίζουμε τι άλλο να δοκιμάσουμε.



Εικόνα 6. Armitage User Interface

2-2-2. Διεπαφή Armitage

Η διεπαφή του Armitage αποτελείται από 3 κύρια παράθυρα:

Modules

Περιηγητής μέσα από τον οποίο μπορεί να γίνει επιλογή των exploit, payload, auxiliary ή post modules για κάθε στόχο. Υπάρχει επίσης η δυνατότητα αναζήτησης με κάποια λέξη κλειδί. Ένα από τα μεγαλύτερα πλεονεκτήματα του Armitage είναι η δυνατότητα να εκτελεστεί κάποιο module εναντίον πολλαπλών στόχων ταυτόχρονα.

Targets

Γραφικό κομμάτι του Armitage που αναπαριστά μια χαρτογράφηση του δικτύου προς εξέταση με τη μορφή οθονών υπολογιστή όπου κάτω από κάθε υπολογιστή αναγράφεται η IP του. Όταν το λειτουργικό σύστημα ενός υπολογιστή έχει γνωστοποιηθεί αναγράφεται στην οθόνη του, ενώ αν έχει παραβιαστεί, η οθόνη κοκκινίζει δηλώνοντας ότι επιτεύχθηκε πρόσβαση μέσω ενός exploit και αποκτήθηκε ο έλεγχός του.

Tabs

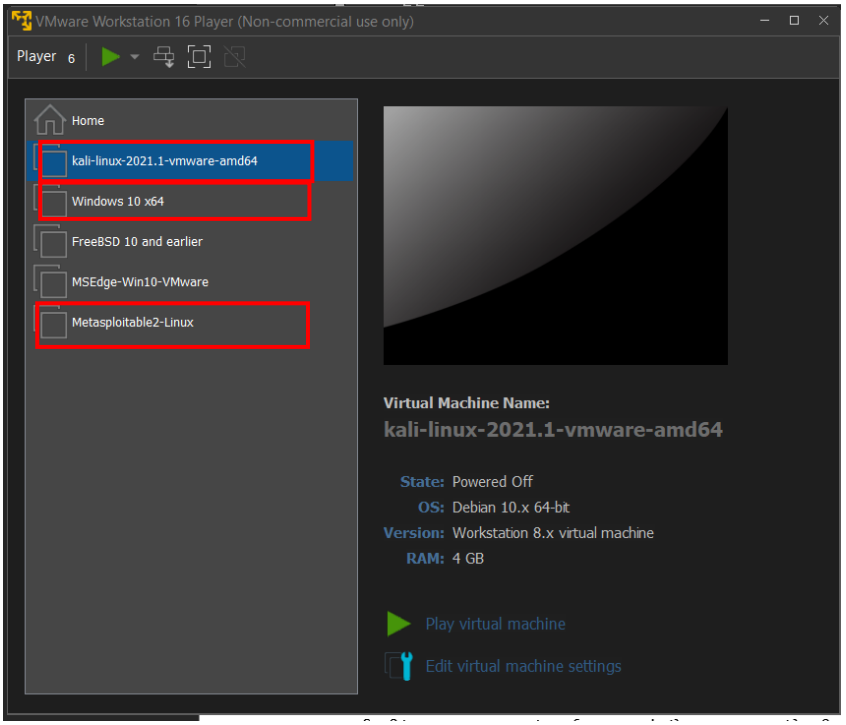
Ο χώρος που έχει δεσμευτεί στο Armitage για την διαχείριση της κονσόλας του Metasploit, των shells που αποκτούνται, των πινάκων από hosts, κ.α. Η κονσόλα του Metasploit παρουσιάζει ιδιαίτερο ενδιαφέρον καθώς μεταβάλλεται δυναμικά ανάλογα με τις ενέργειες που πραγματοποιούνται μέσω του Armitage, αλλά και αντίστροφα (π.χ. η επιλογή ενός exploit από το παράθυρο modules θα εμφανίσει αυτόματα την αντίστοιχη εντολή στην κονσόλα).

3. Περιβάλλον εργασίας

Για τις ανάγκες της παρούσας διατριβής χρησιμοποιήθηκε το πρόγραμμα VMWare Workstation 16 σε περιβάλλον Windows 10, 64-bit. Δημιουργήθηκαν οι εξής εικονικοί υπολογιστές:

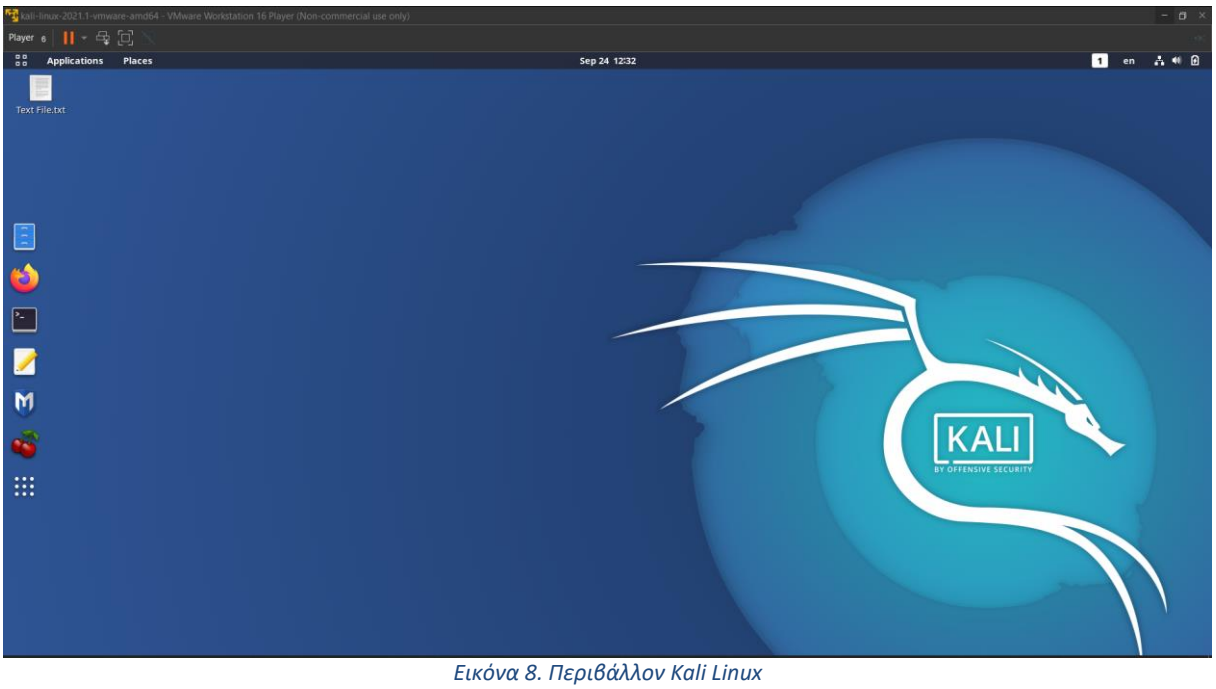
- Kali Linux 2021.1 (4GB RAM)
- Metasploitable 2 (512MB RAM)
- Windows 10 (2 GB RAM)

Το VM Kali Linux έπαιξε τον ρόλο του επιτιθέμενου ενώ το Metasploitable 2 και το Windows 10 τον ρόλο του στόχου. Στα επόμενα κεφάλαιο θα παρουσιαστεί η εφαρμογή του Armitage στα δύο VM-στόχους, το Metasploitable 2 και το Windows 10.



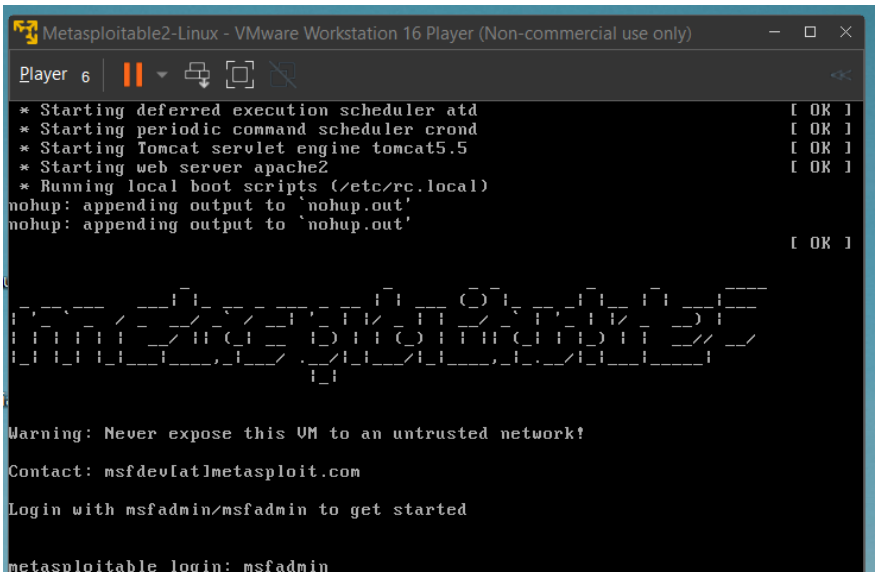
Εικόνα 7. Virtual Machines σε VMWare

Το Kali Linux είναι μια διανομή Linux βασισμένη στο Debian με κύριο σκοπό τη χρήση σε δοκιμές διείσδυσης. Αποτελεί τον διάδοχο του BackTrack Linux και όμοια με τον προκάτοχό του, είναι αφιερωμένη στην ηλεκτρονική ασφάλεια και περιέχει ορισμένα από τα σημαντικότερα εργαλεία που χρησιμοποιούν hackers και pentesters διεθνώς.



Εικόνα 8. Περιβάλλον Kali Linux

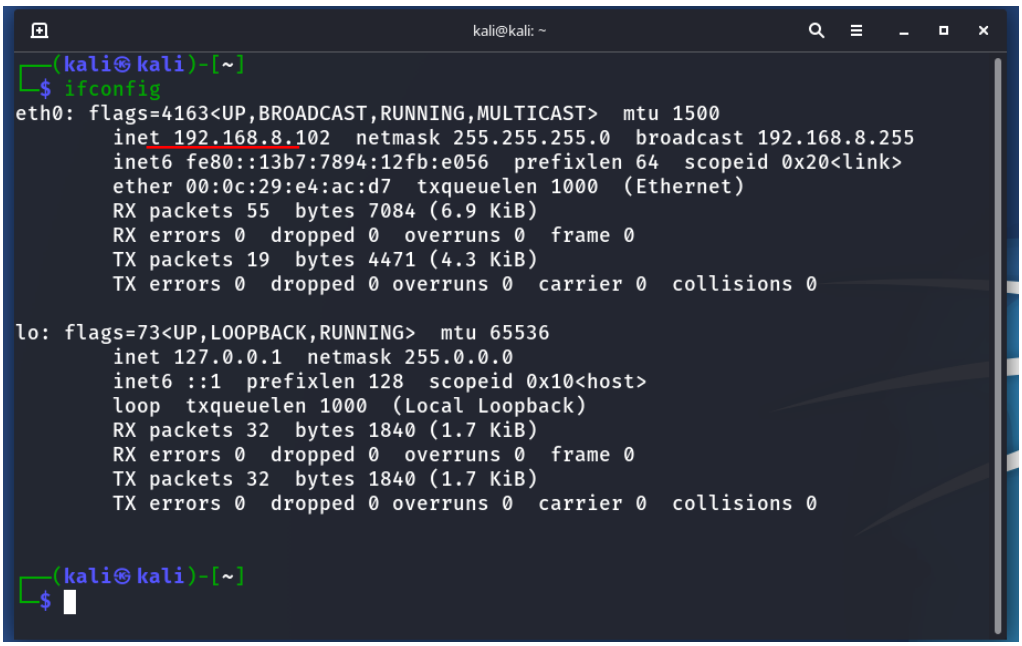
Το Metasploitable 2 είναι μια εσκεμμένα ευπαθής Linux εικονική μηχανή, σχεδιασμένη ακριβώς για το σκοπό της δοκιμής εργαλείων διείσδυσης και εξάσκησης τεχνικών διείσδυσης.



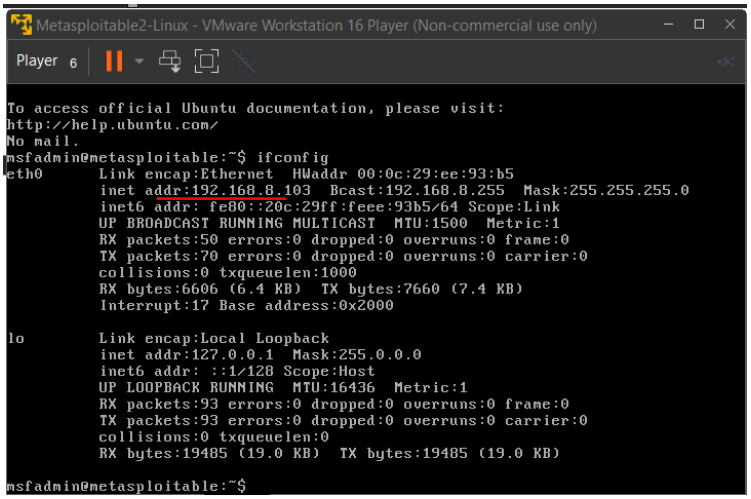
Εικόνα 9. Περιβάλλον Metasploitable 2

Για την επικοινωνία των συστημάτων ορίστηκαν οι αντίστοιχοι network adapter των VM σε Bridged, έτσι είναι συνδεδεμένοι απευθείας στο τοπικό δίκτυο μέσω ενός router. Με την εντολή -ifconfig στο terminal κάθε VM καταγράφηκαν οι IP διευθύνσεις τους:

- Kali Linux: 192.168.8.102
- Metasploitable 2: 192.168.8.103
- Windows 10: 192.168.8.104



Εικόνα 10. Διεύθυνση IP Kali Linux

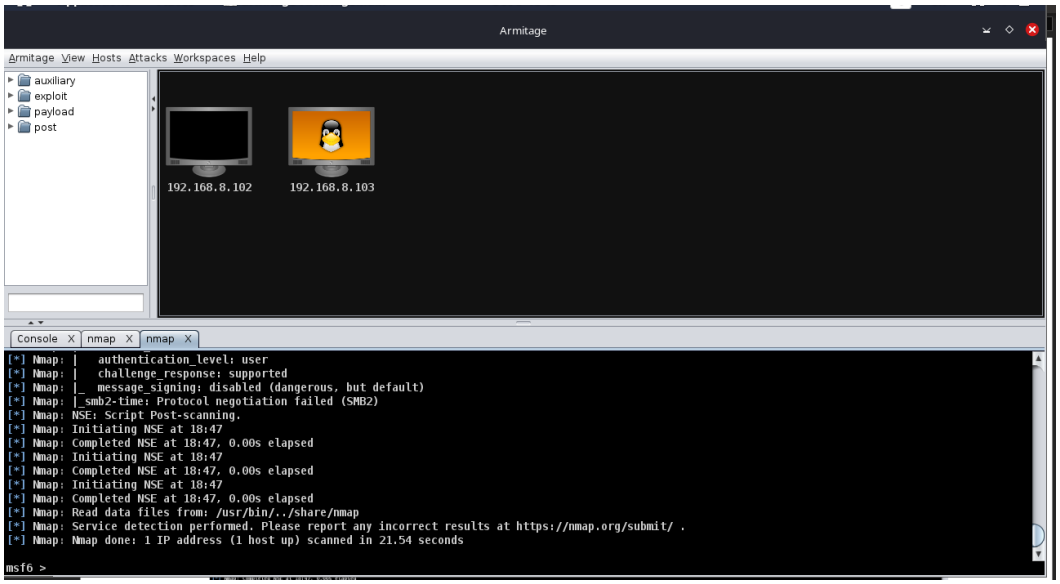


Εικόνα 11. Διεύθυνση IP Metasploitable 2

4. Εφαρμογή του Armitage στο Metasploitable 2

Η Έκδοση του Armitage που χρησιμοποιήθηκε είναι η 1.4.11.

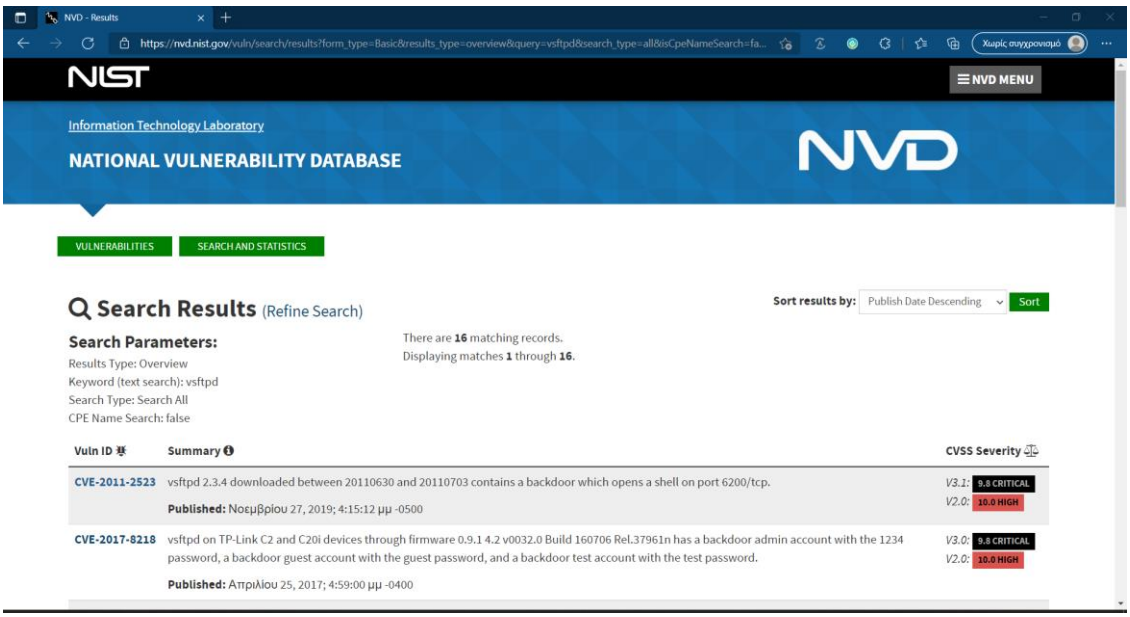
Ξεκινώντας μια επίθεση με το Armitage, η πρώτη κίνηση είναι να πραγματοποιηθεί μια σάρωση θυρών με το Nmap ώστε να εντοπιστούν οι ενεργοί υπολογιστές και να συλλεχθούν οι απαραίτητες πληροφορίες για τις ενεργές θύρες τους και τις υπηρεσίες που τρέχουν. Η επιλογή γίνεται από την καρτέλα Hosts -> Nmap Scan -> Intense Scan εισάγοντας την IP του Metasploitable 2 (192.168.8.103). Όταν ολοκληρωθεί η σάρωση παρατηρείται ότι το σύστημα-στόχος έχει προστεθεί στους targets και ότι έχει λειτουργικό σύστημα Linux 2.6.X.



Εικόνα 12. Χαρτογράφηση δικτύου Kali Linux - Metasploitable 2

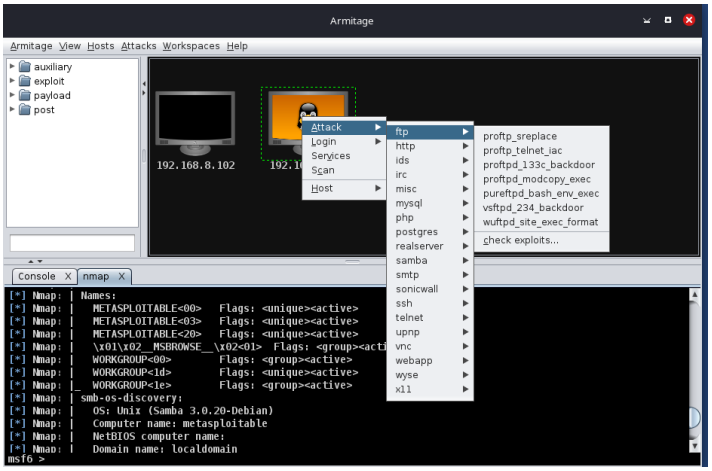
Για την εύρεση των μεθόδων επιθέσεων που αντιστοιχούν σε αδυναμίες των υπηρεσιών που βρέθηκαν ότι τρέχουν στον στόχο επιλέγεται από την καρτέλα Attacks -> Find Attacks. Αφού τελειώσει η ανάλυση επιθέσεων, πατώντας δεξί κλικ στο στόχο, εμφανίζεται η επιλογή “Attack” που περιέχει όλα τα exploits χωρισμένα σε κατηγορίες, βάση πρωτοκόλλων κυρίως, τα οποία έχει ανακαλύψει η παραπάνω σάρωση. Το γεγονός ότι έχουν βρεθεί τα exploits δε σημαίνει όμως ότι μπορούν να εκτελεστούν και με επιτυχία.

Ένας τρόπος για να αποφευχθεί η δοκιμή των exploits ένα προς ένα, είναι να συμβουλευτεί κανείς ένα εργαλείο σάρωσης ευπαθειών, και βάση της ύπαρξης exploit που προτείνεται για το Metasploit και την επικινδυνότητα της ευπάθειας, να γίνει η επιλογή του κατάλληλου. Αναζητώντας μια ευπάθεια για το “vsftpd” στην ιστοσελίδα nvd.nist.gov, ανακαλύπτουμε ότι η χρήση του exploit “VSFTPD v2.3.4 Backdoor Command Execution” έχει υψηλή αποτελεσματικότητα.

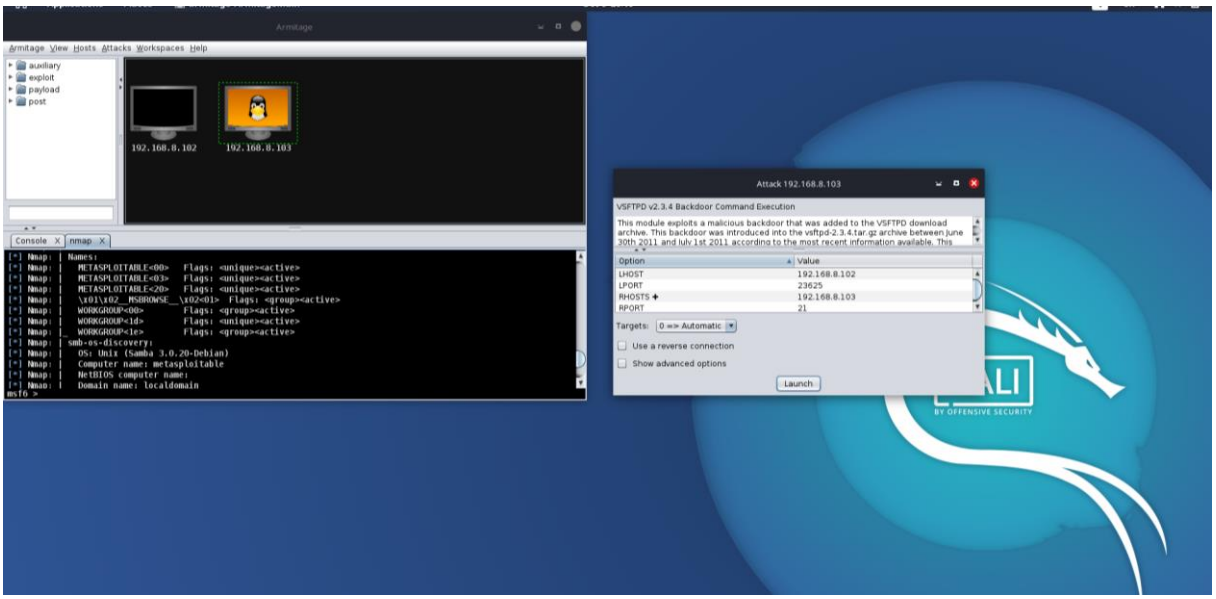


Εικόνα 13. Εύρεση ευπαθειών στο nvid.nist.gov

Επιλέγοντάς την, ανοίγει το παράθυρο το παράθυρο με τις επιλογές/στοιχεία για local (LHOST) και remote hosts (RHOST) που έχουν συμπληρωθεί αυτόματα.



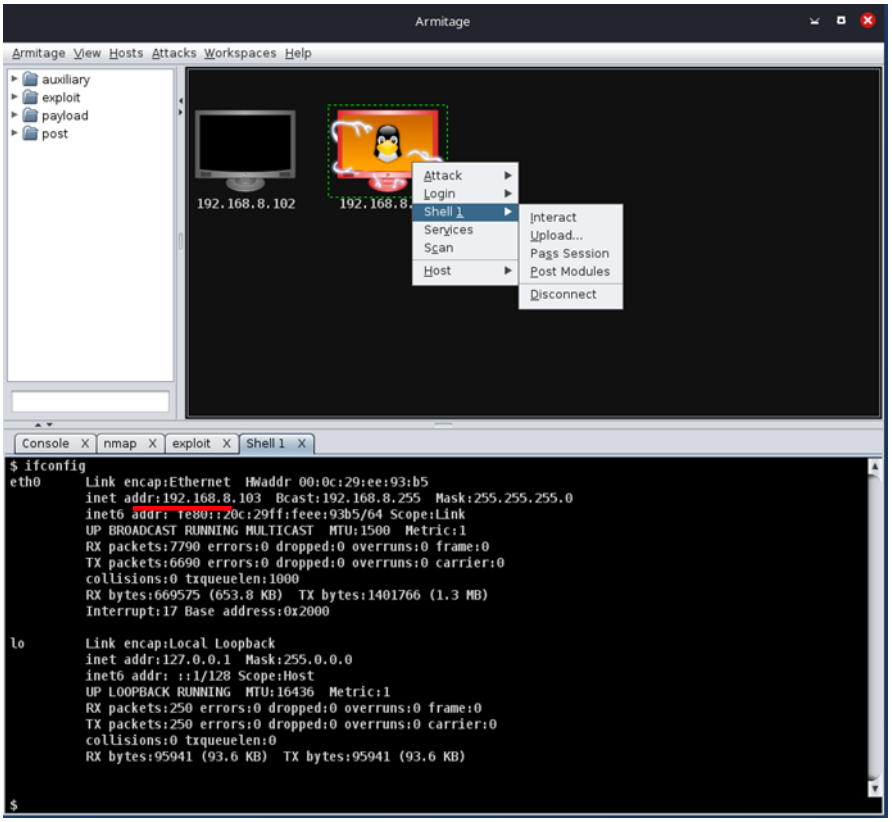
Εικόνα 14. Επιλογή exploit vsftp για το Metasploitable 2



Εικόνα 15. Εξαπώληση επίθεσης vsftp στο Metasploitable 2

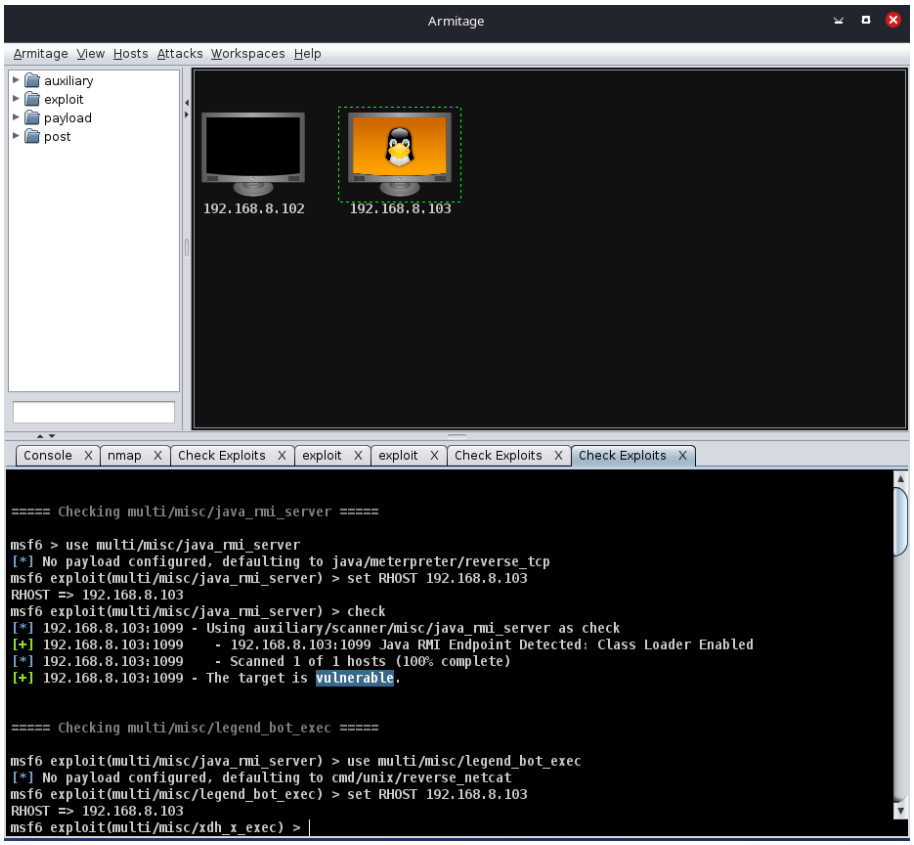
Πατώντας “Launch”, το exploit εκτελείται με επιτυχία και παρατηρούμε ότι αλλάζει το χρώμα της οθόνης του συστήματος-στόχου σε κόκκινο με κεραυνούς, κάτι το οποίο υποδεικνύει ότι υπάρχει πλέον πλήρης πρόσβαση ως root.

Για τον έλεγχο του συστήματος και την εκτέλεση εντολών στο shell που αποκτήθηκε, πατώντας δεξί κλικ στην οθόνη του στόχου στο παράθυρο Targets, επιλέγεται το “Shell 1”, το οποίο τελικά ανοίγει ένα tab με το terminal του Metasploitable 2. Επιβεβαιώνουμε ότι έχουμε συνδεθεί στο Metasploitable 2 , δίνοντας στο Shell 1 την εντολή ifconfig και βλέπουμε την IP του Metasploitable 2.



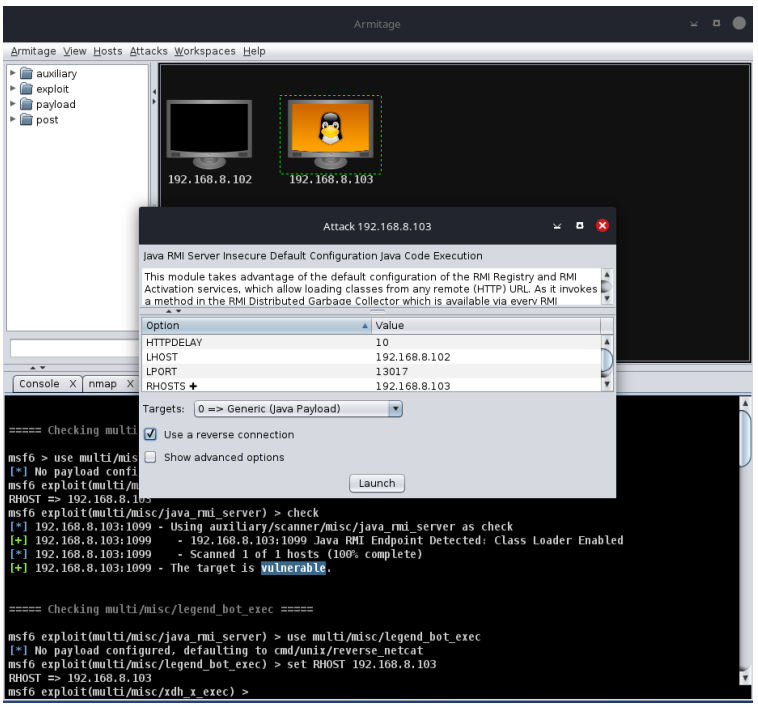
Εικόνα 16. Απόκτηση πρόσβασης στο Metasploitable 2

Ένας άλλος τρόπος εύρεσης του exploit που θα παρέχει πρόσβαση σε ένα σύστημα είναι ο έλεγχος των exploits που έχουν βρεθεί κάνοντας χρήση της εντολής “check” του Metasploit μέσω της επιλογής “check exploits” πατώντας δεξί κλικ στην οθόνη του στόχου και ανατρέχοντας στην επιθυμητή κατηγορία exploits. Με αυτή τη μέθοδο το Armitage εξετάζει αυτόματα τα exploits ένα προς ένα ανοίγοντας ένα tab όπου φαίνεται ο έλεγχος του κάθε exploit με τον αντίστοιχο χαρακτηρισμό. Συγκεκριμένα εάν το exploit είναι κατάλληλο η κονσόλα επιστρέφει το μήνυμα “The target is vulnerable” ενώ αν δεν είναι επιστρέφει το μήνυμα “The target is not exploitable”. Στην παρακάτω περίπτωση έγινε αναζήτηση των exploits για την κατηγορία “misc” και το exploit που βρέθηκε κατάλληλο είναι το “multi/misc/java_rmi_server”.



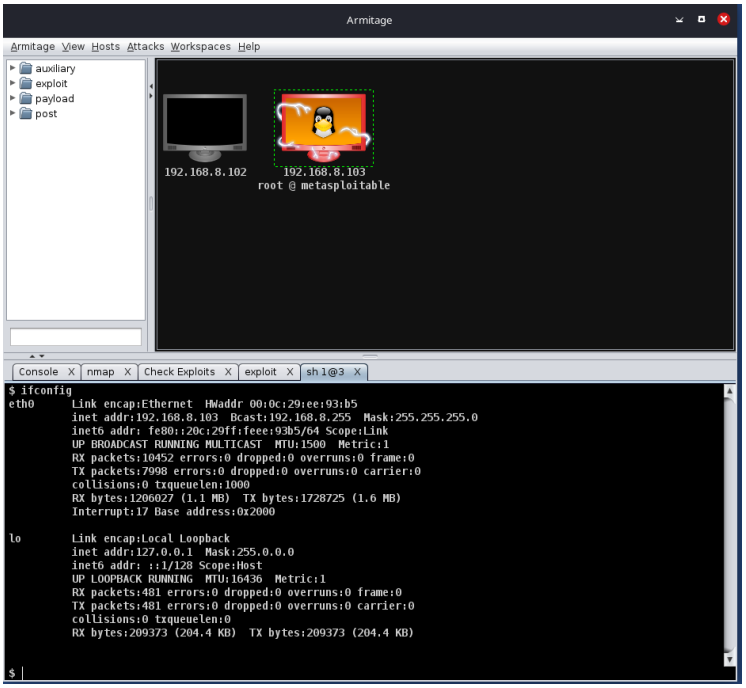
Εικόνα 17. Εύρεση ευπάθειας με την επιλογή check exploits.

Επιλέγοντας το exploit αυτό (misc/java_rmi_server), στις επιλογές ζητάμε να γίνει μια αντίστροφη σύνδεση (reverse connection) από το Metasploitable 2 στον επιτιθέμενο, και πατώντας Launch το exploit εκτελείται με επιτυχία.



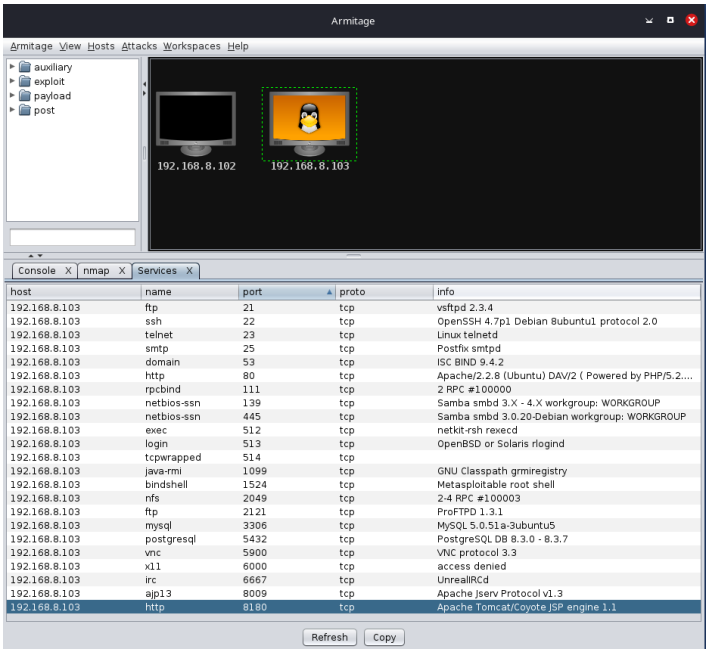
Εικόνα 18. Επιλογή exploit java_rmi_server

Επιβεβαιώνουμε ανοίγοντας ένα command shell με Meterpreter, και βλέπουμε πως έχουμε συνδεθεί στο Metasploitable 2 δικαιώματα root.



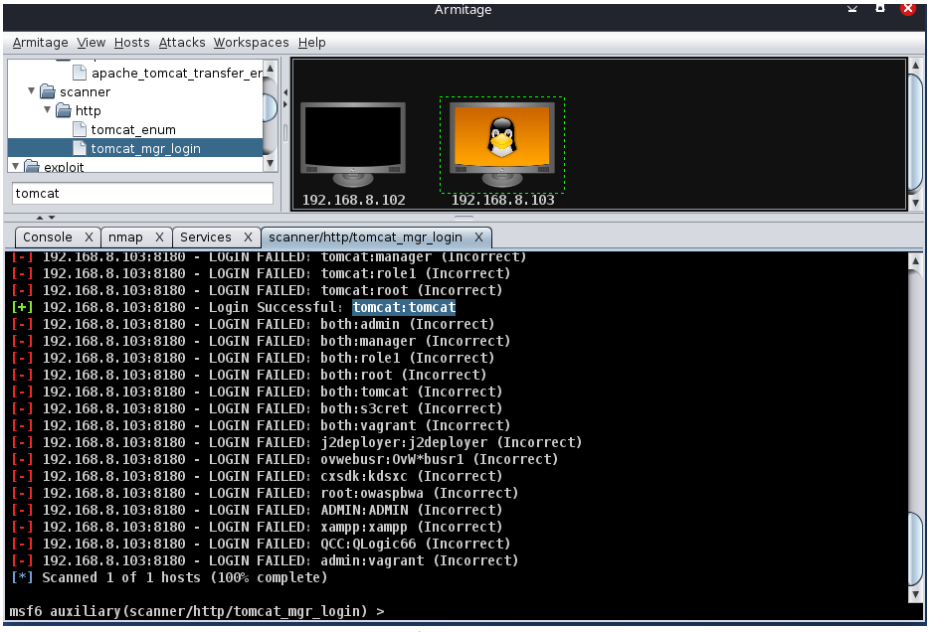
Εικόνα 19. Απόκτηση πρόσβασης root στο Metasploitable 2

Εξετάζοντας τέλος μια λίγο πιο σύνθετη και λιγότερο αυτοματοποιημένη μέθοδο εκμετάλλευσης του συστήματος, επιχειρήθηκε μια επίθεση Brute Force. Πατώντας δεξί κλικ στο σύστημα Metasploitable 2 και επιλέγοντας “services” ανοίγει ένα tab με όλες τις ανοιχτές θύρες που βρέθηκαν από το nmap. Σε μια από αυτές ακούει ο Apache Tomcat Server (port 8180).



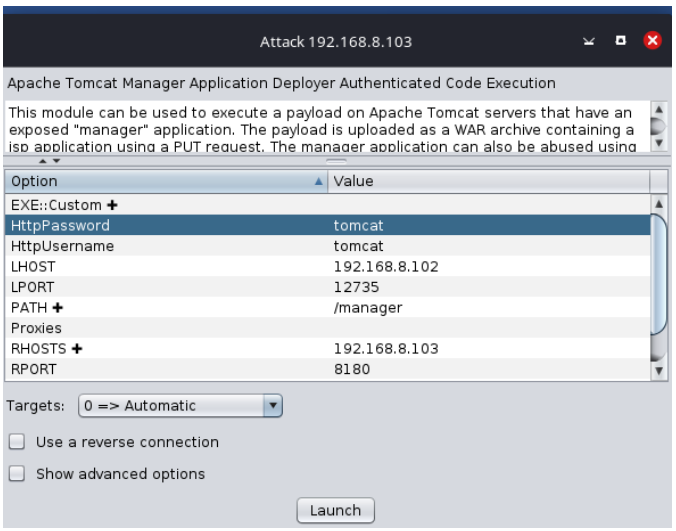
Εικόνα 20. Services στο Metsploitable 2

Κάνοντας αναζήτηση “tomcat” στο παράθυρο modules, βρίσκομε ένα auxiliary module στην κατηγορία scanners, “tomcat_mgr_login”. Επιλέγοντας και παραμετροποιώντας βάσει της παραπάνω ανοιχτής θύρας, κάνοντας κλικ στο “launch” ξεκινά η Brute Force επίθεση. Παρατηρείται στα αποτελέσματα στην κονσόλα ότι δοκιμάζονται διάφορα passwords, όπου ένας συνδυασμός είναι επιτυχής επιστρέφοντας μήνυμα “192.168.8.103:8180 - LOGIN SUCCESSFUL : tomcat:tomcat”.



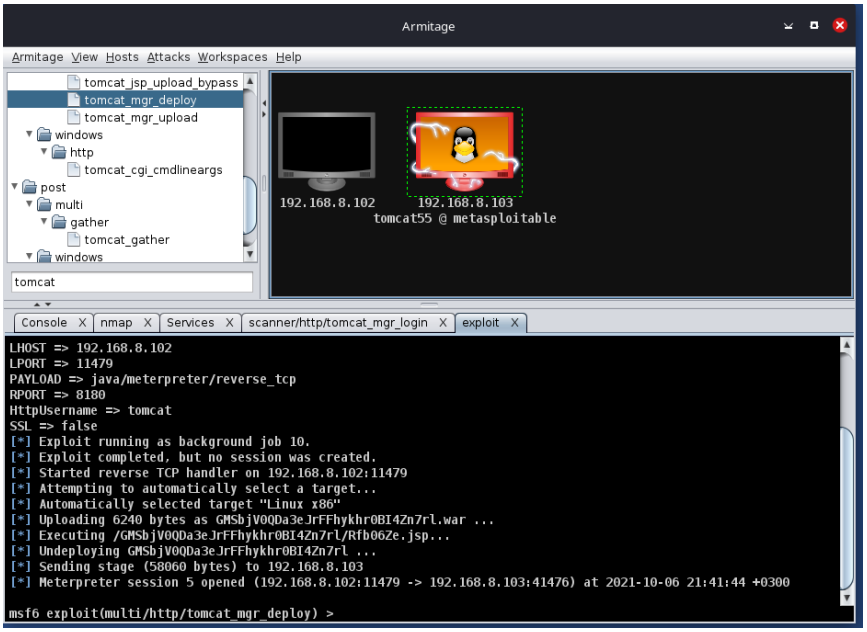
Εικόνα 21. Εύρεση username/password Tomcat Server με Brute Force

Γνωρίζοντας πλέον το σωστό username και password, γίνεται επιλογή ενός exploit module που βρέθηκε νωρίτερα από την αναζήτηση: “tomcat_mgr_deploy”. Στις επιλογές αρκεί να εισαχθούν τα username και password και η θύρα 8180.



Εικόνα 22. Παραμετροποίηση tomcat_mgr_deploy

Πατώντας “Launch” το exploit τρέχει με επιτυχία επιστρέφοντας ένα Meterpreter session προς το Metasploitable 2.

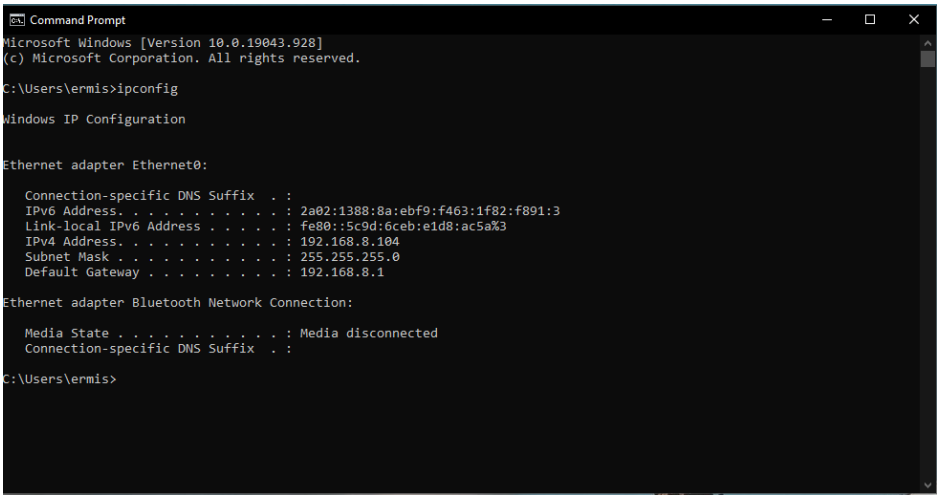


Εικόνα 23. Απόκτηση πρόσβασης στο Metasploitable 2 μέσω του Tomcat Server

5. Εφαρμογή του Armitage και Metasploit σε Windows 10

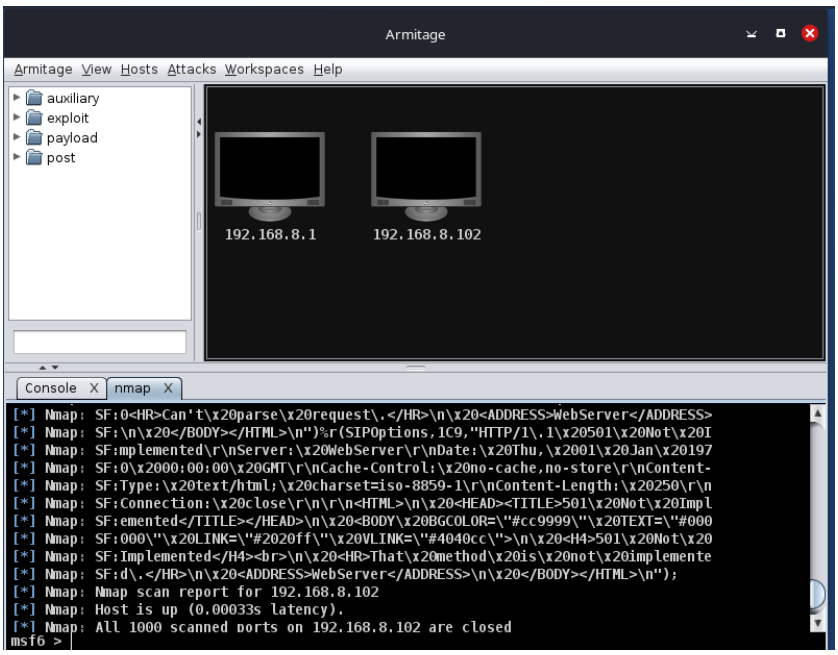
5-1. Σάρωση δικτύου για τον εντοπισμό host Windows 10 με ενεργοποιημένο Firewall

Όπως είδαμε παραπάνω, το περιβάλλον εργασίας που έχει δημιουργηθεί είναι το ίδιο, η IP του VM με Kali Linux παραμένει η 192.168.8.102 ενώ του VM με περιβάλλον Windows 10 είναι η 192.168.2.104.

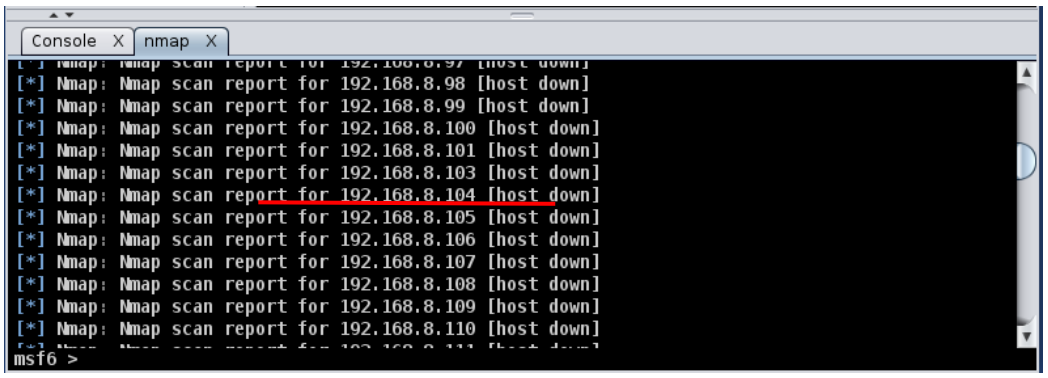


Εικόνα 24. Διεύθυνση IP Windows 10

Ξεκινώντας μια επίθεση με το Armitage, η πρώτη κίνηση είναι να πραγματοποιηθεί μια σάρωση θυρών με το Nmap ώστε να εντοπιστούν οι ενεργοί υπολογιστές και να συλλεχθούν οι απαραίτητες πληροφορίες για τις ενεργές θύρες τους και τις υπηρεσίες που τρέχουν. Η επιλογή γίνεται από την καρτέλα Hosts -> Nmap Scan -> Intense Scan εισάγοντας την IP του δικτύου που θέλουμε να σαρώσουμε (192.168.8.0/24), θεωρώντας ότι δεν γνωρίζουμε σε ποιες IP υπάρχουν ενεργοί Η/Υ. Όταν ολοκληρωθεί η σάρωση παρατηρείται ότι έχουν εντοπιστεί ενεργοί hosts με IP 192.168.8.1 και 192.168.8.102, ενώ το σύστημα-στόχος Windows 10 με IP 192.168.8.104 που μας ενδιαφέρει χαρακτηρίζεται ως ανενεργός, παρόλο που λειτουργεί κανονικά και είναι ενεργός και συνδεδεμένος στο δίκτυο. Οι hosts που βρέθηκαν ενεργοί είναι το router (192.168.8.1), στο οποίο βρέθηκαν ενεργές θύρες 53 (dns) και 80 (web server), και το VM με Kali Linux (192.168.8.102) στο οποίο δεν βρέθηκαν ενεργές θύρες, πλην όμως ελήφθη απάντηση στο ping scan και χαρακτηρίστηκε ως ενεργός.

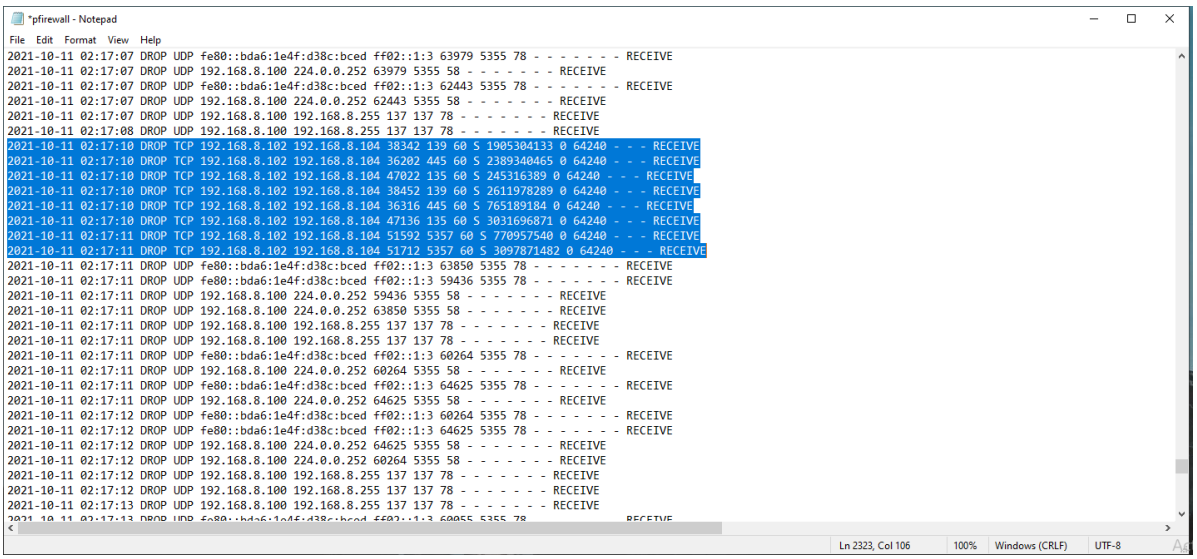


Εικόνα 25. Ενεργοί Hosts μετά από Intense Scan



Εικόνα 26. Χαρακτηρισμός Windows 10 ως ανενεργός.

Επαναλαμβάνουμε την σάρωση, αυτή την φορά επιλέγοντας Hosts -> Nmap Scan -> Intense Scan, no ping, με την οποία θα θεωρηθούν όλοι οι hosts ενεργοί και θα ελεγχθούν όλες οι θύρες όλων των πιθανών hosts. Γι' αυτό τον λόγο η σάρωση αυτή διαρκεί μεγαλύτερο χρονικό διάστημα και ενημερωνόμαστε μέσα από την καρτέλα του nmap για το ποσοστό ολοκλήρωσης της διαδικασίας. Μετά την ολοκλήρωση της διαδικασίας, παρατηρούμε ότι ο host 192.168.8.104 χαρακτηρίζεται ως ενεργός, πλην όμως δεν βρέθηκε κάποια θύρα ανοιχτή. Ελέγχοντας το αρχείο καταγραφής (logfile) του Firewall των Windows 10, παρατηρούμε ότι τα πακέτα που εστάλησαν από το την IP 192.168.8.102, δηλαδή από τον επιτιθέμενο, στις θύρες 135, 139, 445 και 5357, απορρίφθηκαν.

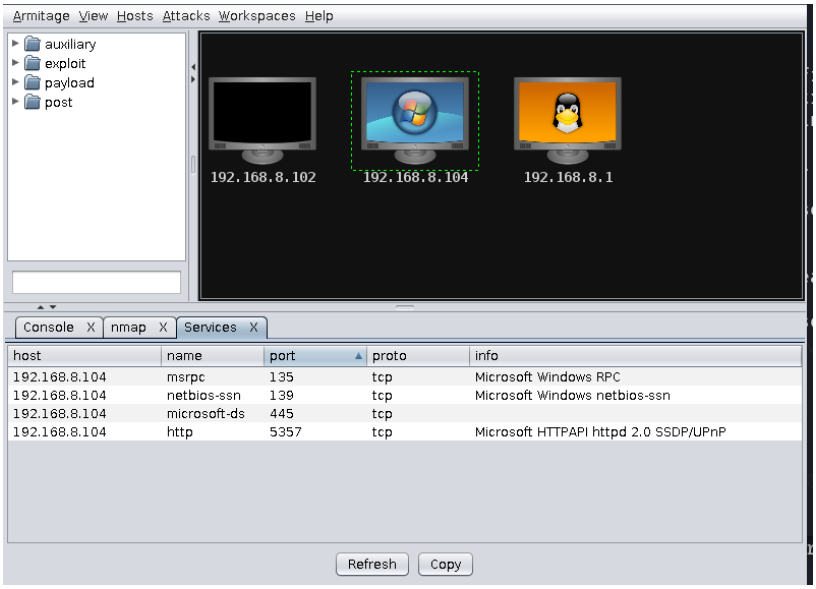


Εικόνα 27. Απόρριψη πακέτων από το Firewall των Windows 10.

Επαναλαμβάνουμε την σάρωση, αυτή την φορά επιλέγοντας Hosts -> Nmap Scan -> MSF Scan. Αυτή η λειτουργία θα σαρώσει μερικές θύρες. Στη συνέχεια απαριθμεί αρκετές γνωστές υπηρεσίες χρησιμοποιώντας auxiliary modules του Metasploit που έχουν δημιουργηθεί για το σκοπό αυτό. Το αποτέλεσμα είναι το ίδιο, δεν βρέθηκαν ανοιχτές θύρες και το Firewall απέρριψε πάλι τα πακέτα.

5-2. Σάρωση δικτύου και προσπάθεια εκμετάλλευσης αδυναμιών host Windows 10 χωρίς Firewall.

Εκτελώντας εκ νέου μια σάρωση Intense scan, no ping στο δίκτυο 192.168.8.0/24, αυτή την φορά έχοντας απενεργοποιήσει το Firewall στο VM των Windows 10, παρατηρούμε ότι ο host βρέθηκε ενεργός, αναγνωρίστηκε ως λειτουργικό σύστημα κάποια έκδοση των Windows, και βρέθηκαν ενεργές θύρες 135, 139, 445 και 5357 με τις αντίστοιχες υπηρεσίες όπως φαίνεται στον πίνακα των services.



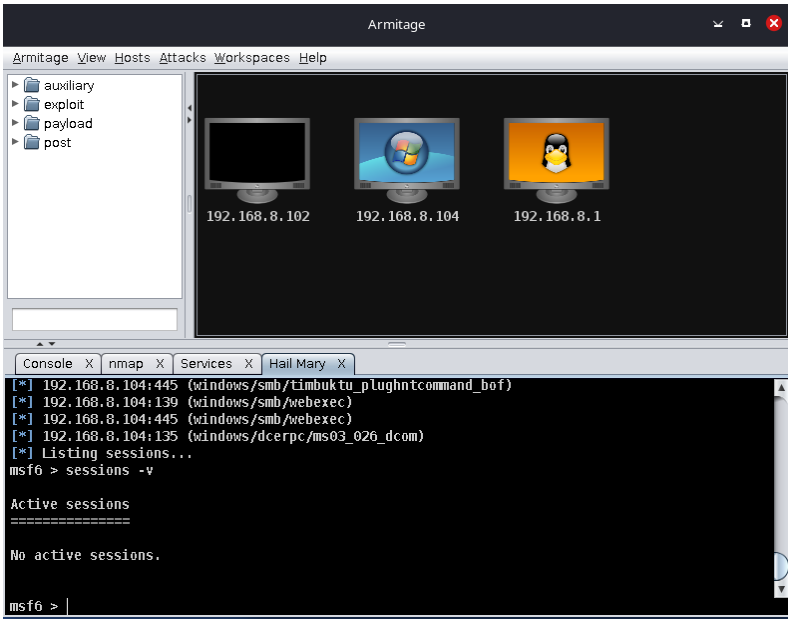
Εικόνα 28. Εντοπισμός Host Windows 10 και ενεργών υπηρεσιών.

Η σχολαστική σάρωση και ο εντοπισμός των ενεργών υπηρεσιών σε έναν host, αποτελούν

την πιο σημαντική εργασία που θα καθορίσει και την συνέχεια της εύρεσης ευπαθειών και της διείσδυσης στο σύστημα. Κατανοούμε λοιπόν την σημαντική χρησιμότητα των Firewall, τα οποία αποτρέπουν τον εντοπισμό των ενεργών υπηρεσιών και θυρών.

Στην συνέχεια, επιλέγεται από την καρτέλα Attacks -> Find Attacks. Αφού τελειώσει η ανάλυση επιθέσεων, πατώντας δεξί κλικ στο στόχο, εμφανίζεται η επιλογή “Attack” που περιέχει όλα τα exploits χωρισμένα σε κατηγορίες, βάση πρωτοκόλλων κυρίως, τα οποία έχει ανακαλύψει η παραπάνω σάρωση. Δοκιμάζοντας τις προτεινόμενες επιθέσεις, καμία από αυτές δεν είναι αποτελεσματική.

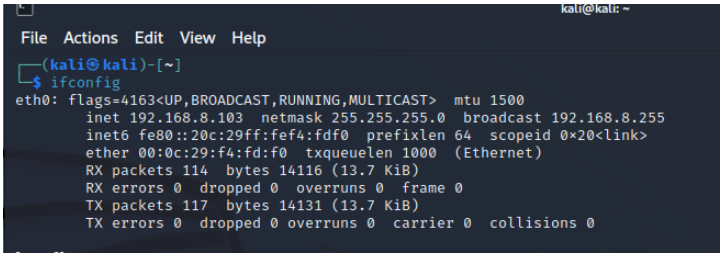
Τέλος, πραγματοποιούμε μια αυτοματοποιημένη Hail Mary επίθεση, η οποία όμως δεν εντόπισε κάποια ευπάθεια και δεν ενεργοποίησε κάποια συνεδρία.



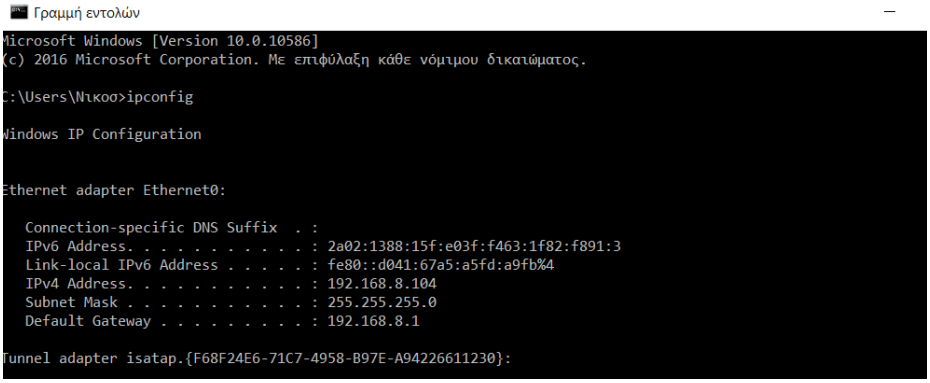
Εικόνα 29. Hail Mary επίθεση.

5-3. Επίθεση σε Windows 10 με χρήση κακόβουλου λογισμικού.

Σε αυτό το κεφάλαιο, θα δούμε πώς το Metasploit μπορεί να χρησιμοποιηθεί για επίθεση σε υπολογιστή Windows 10. Αυτό θα το κάνουμε μέσω ενός κακόβουλου εκτελέσιμου αρχείου, χρησιμοποιώντας το Shellter. Το περιβάλλον εργασίας που έχει δημιουργηθεί είναι το ίδιο, η IP του VM με Kali Linux είναι η 192.168.8.103 ενώ του VM με περιβάλλον Windows 10 είναι η 192.168.2.104.



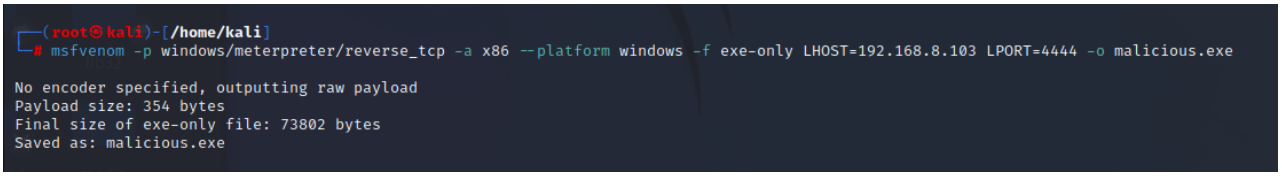
Εικόνα 30. Διεύθυνση IP Kali Linux



Εικόνα 31. Διεύθυνση IP Windows 10

5-3-1. Δημιουργία κακόβουλου αρχείου .exe

Για να δημιουργήσουμε το εκτελέσιμο αρχείο, θα χρησιμοποιήσουμε το msfvenom όπως φαίνεται στην παρακάτω εντολή:



Εικόνα 32. Δημιουργία κακόβουλου αρχείου malicious.exe

Η εντολή δίνει εντολή στο msfvenom να δημιουργήσει ένα εκτελέσιμο αρχείο των Windows 32-bit που υλοποιεί μια αντίστροφη σύνδεση TCP για το ωφέλιμο φορτίο. Η μορφή πρέπει να οριστεί ως τύπου .exe και πρέπει να οριστούν ο τοπικός κεντρικός υπολογιστής (LHOST) και η τοπική θύρα (LPORT). Στην περίπτωσή μας, το LHOST είναι η διεύθυνση IP του επιτιθέμενου υπολογιστή μας Kali Linux και το LPORT είναι η θύρα για να «ακούμε» μια σύνδεση από τον στόχο αφού έχει παραβιαστεί.

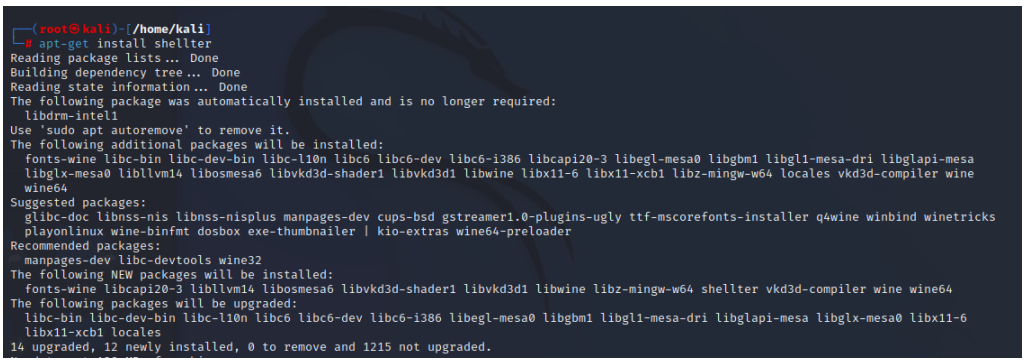
Τα αντιικά προγράμματα λειτουργούν ανιχνεύοντας κακόβουλες υπογραφές μέσα σε εκτελέσιμα αρχεία. Έτσι, το αρχείο μας θα επισημανθεί ως κακόβουλο μόλις εισαχθεί στο περιβάλλον των Windows. Πρέπει να βρούμε έναν τρόπο να το τροποποιήσουμε για να παρακάμψουμε τον εντοπισμό προστασίας από ιούς. Θα το κωδικοποιήσουμε για να το καταστήσουμε πλήρως μη ανιχνεύσιμο ή FUD (Fully Undetectable).

5-3-2. Κάνοντας το εκτελέσιμο FUD (εντελώς μη ανιχνεύσιμο)

Για να κωδικοποιήσουμε το εκτελέσιμο αρχείο μας, θα χρησιμοποιήσουμε το Shellter. Το Shellter λειτουργεί αλλάζοντας τις υπογραφές του εκτελέσιμου από την προφανώς κακόβουλη σε μια εντελώς νέα και μοναδική που μπορεί να παρακάμψει τον εντοπισμό.

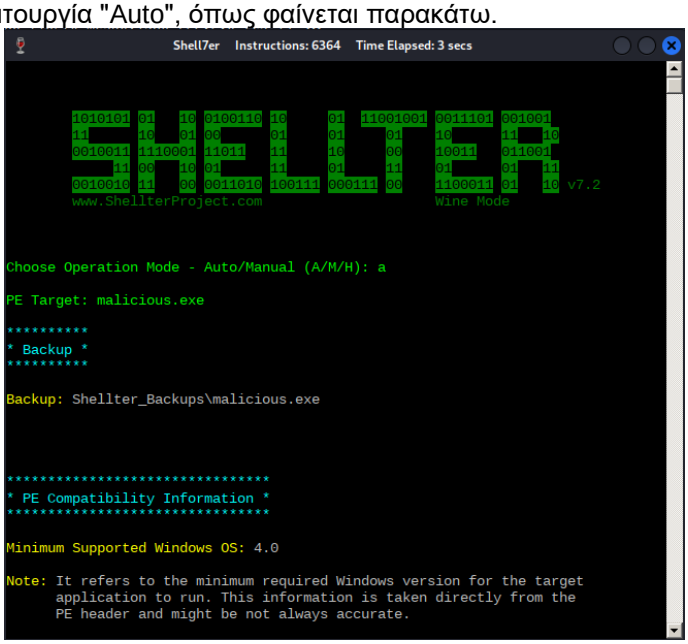
Να σημειώσουμε ότι τα antivirus ελέγχουν επίσης τη συμπεριφορά των εκτελέσιμων και χρησιμοποιούν τεχνικές όπως η ευρετική σάρωση, επομένως δεν περιορίζονται μόνο στον έλεγχο για υπογραφές. Κατά τη διάρκεια των εργαστηριακών δοκιμών, ανακαλύφθηκε ότι το Windows Defender (το οποίο αποστέλλεται από προεπιλογή με τα Windows 10) επεσήμανε το εκτελέσιμο αρχείο έξι από τις δέκα φορές που χρησιμοποιήσαμε το Shellter για την εκτέλεση της κωδικοποίησης. Αυτό συμβαίνει παρά το γεγονός ότι τα Windows 10 είναι μια νέα λήψη με τις πιο πρόσφατες ενημερώσεις κώδικα που εφαρμόζονται! Θα είναι καλύτερο να γίνεται χρήση του Shellter Pro (ή οποιοδήποτε pro crypter) ή να γράφουμε τον δικό μας crypter για να αποφεύγουμε την επισήμανση του προγράμματος προστασίας από ιούς των εκτελέσιμων αρχείων μας.

Εγκαθιστούμε το Shellter στον υπολογιστή μας Kali Linux.



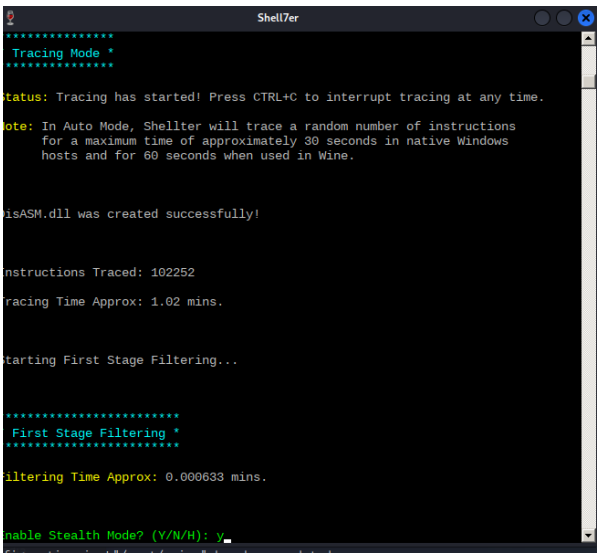
Εικόνα 33. Εγκατάσταση Shellter

Για να εκκινήσουμε το Shellter, απλώς πληκτρολογούμε shellter στο τερματικό. Θα μας ζητηθεί να εισαγάγουμε την απόλυτη διαδρομή προς το εκτελέσιμο για να το κάνουμε FUD. Βεβαιωνόμαστε ότι έχουμε επιλέξει τη λειτουργία "Auto", όπως φαίνεται παρακάτω.



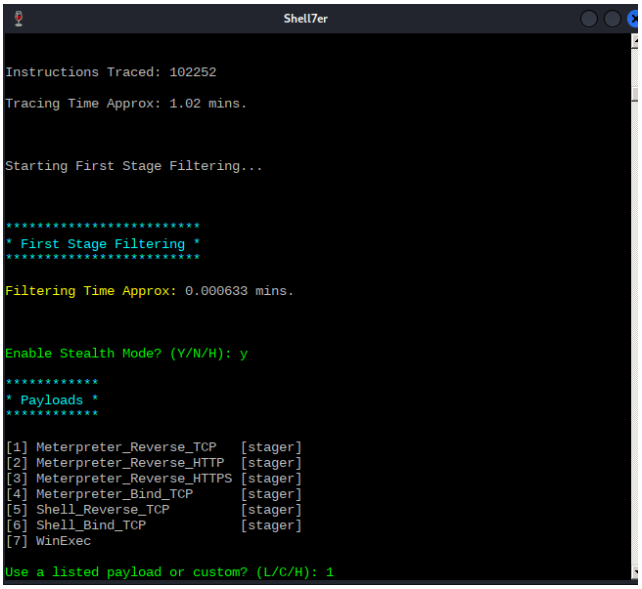
Εικόνα 34. Εκτέλεση Shellter

Στη συνέχεια, το Shellter θα προετοιμάσει και θα εκτελέσει ορισμένους ελέγχους. Στη συνέχεια, θα μας ρωτήσει εάν επιθυμούμε να εκτελέσει τη λειτουργία stealth. Επιλέγουμε "Y" για ναι.



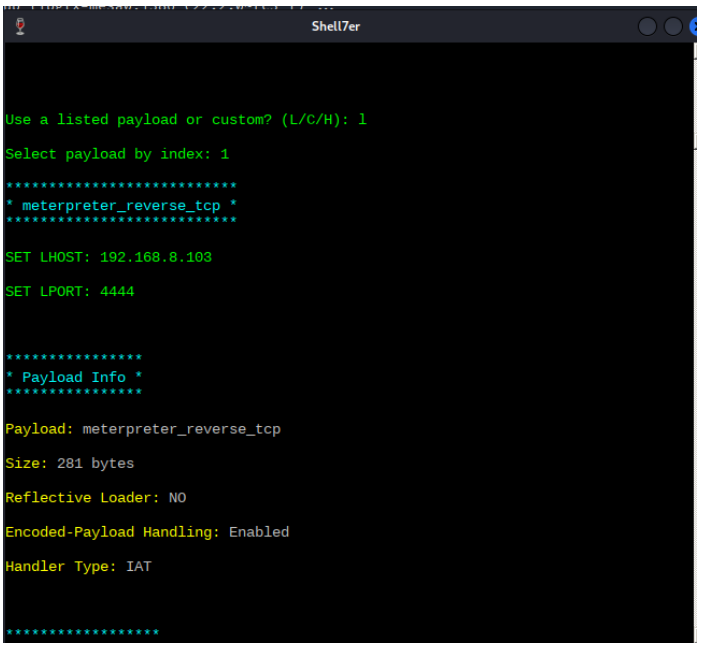
Εικόνα 35. Επιλογή stealth mode

Η επόμενη ερώτηση θα μας ζητήσει να εισαγάγουμε το payload, είτε προσαρμοσμένο είτε καταχωρισμένο. Θα πρέπει να επιλέξουμε ένα από τη λίστα πληκτρολογώντας "L", εκτός εάν θέλουμε να συνεχίσουμε με το δικό μας προσαρμοσμένο payload. Χρειαζόμαστε ένα Meterpreter_Reverse_TCP, επομένως θα πρέπει να επιλέξουμε το "1".



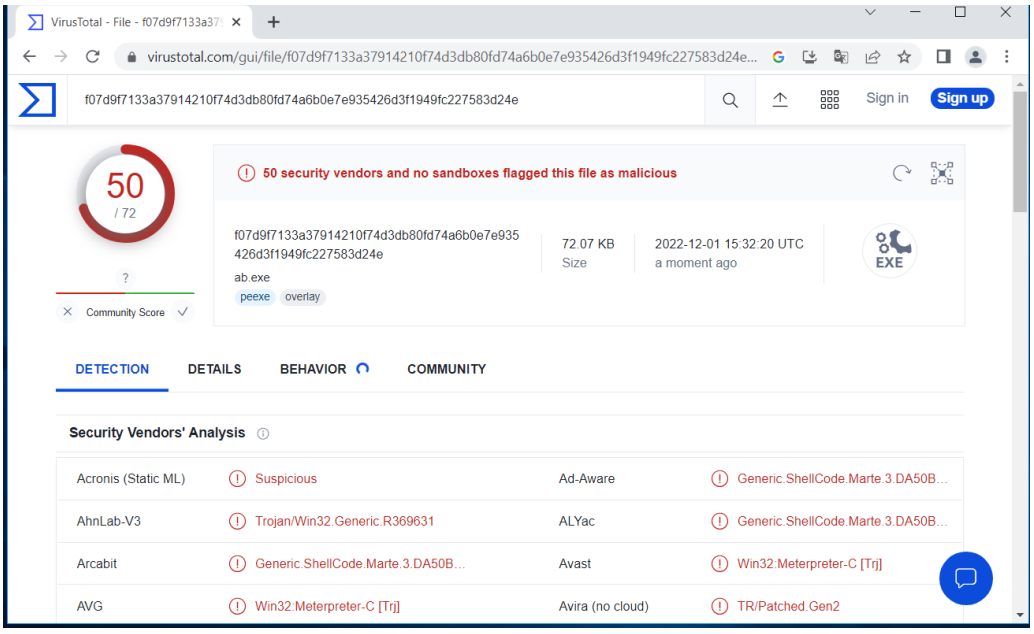
Εικόνα 36. Επιλογή Payload

Πληκτρολογούμε LHOST και LPORT και πατάμε Enter. Το Shellter θα ολοκληρωθεί και θα μας ζητήσει να πατήσουμε Enter.



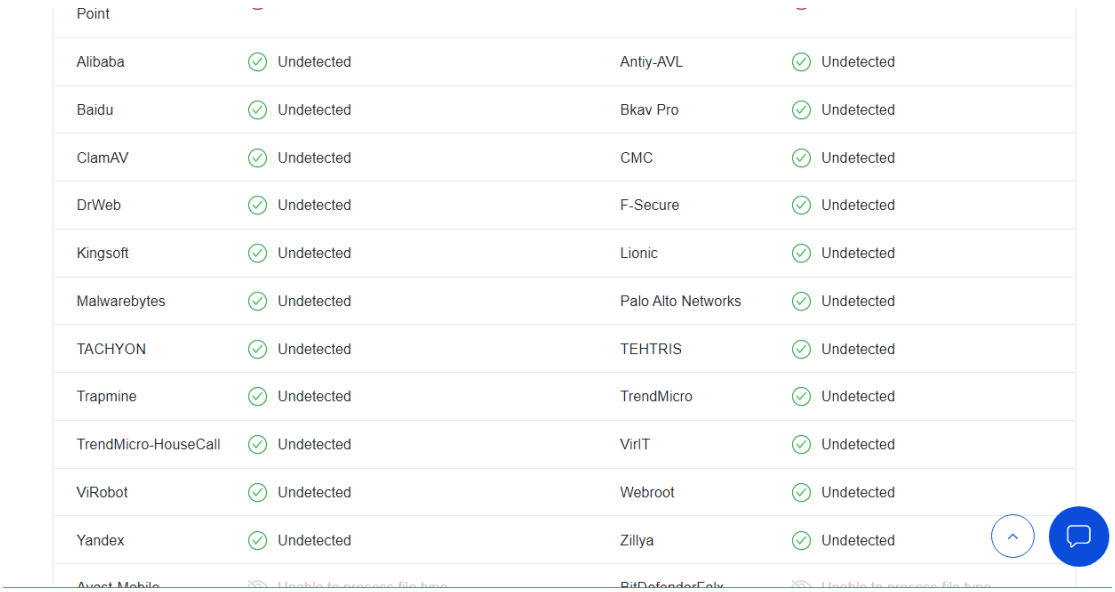
Εικόνα 37. Ρύθμιση shelter

Σε αυτό το σημείο, το εκτελέσιμο αρχείο που παρείχαμε θα έχει γίνει μη ανιχνεύσιμο από antivirus. Για να το ελέγξουμε, ανεβάζουμε το αρχείο στην ιστοσελίδα https://www.virustotal.com και εκτελούμε μια σάρωση. Παρατηρούμε ότι 50 από τα 72 Antivirus το εντόπισαν ως κακόβουλο λογισμικό.



Εικόνα 38. Σάρωση εκτελέσιμου στο virustotal.

Ένας σημαντικός αριθμός Antivirus (22) δεν εντόπισαν το εκτελέσιμο ως κακόβουλο, όπως φαίνεται στην παρακάτω εικόνα. Παρόλαυτά, θα είναι καλύτερο να γίνεται χρήση του Shellter Pro (ή οποιοδήποτε pro crypter) ή να γράφουμε τον δικό μας crypter για να αποφεύγουμε την επισήμανση του προγράμματος προστασίας από ιούς των εκτελέσιμων αρχείων μας.



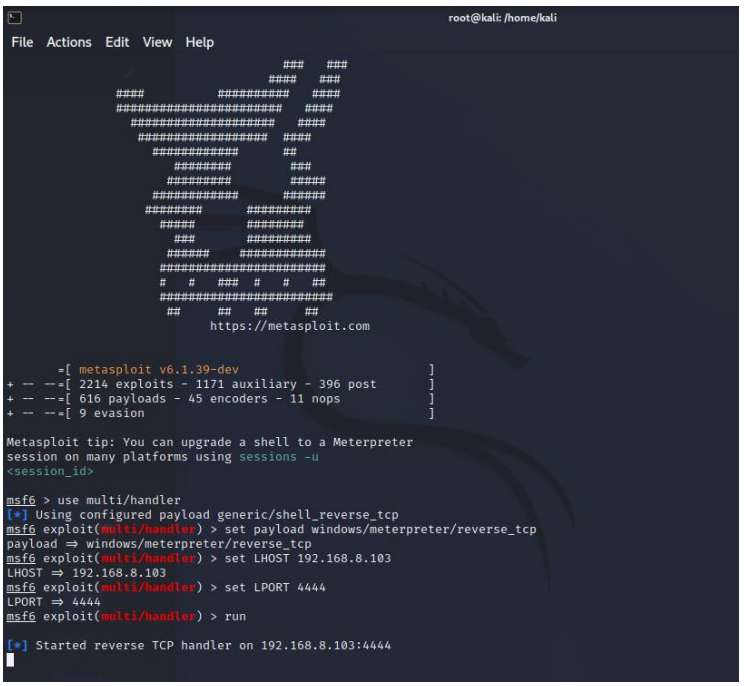
Point			
Alibaba	Undetected	Antiy-AVL	Undetected
Baidu	Undetected	Bkav Pro	Undetected
ClamAV	Undetected	CMC	Undetected
DrWeb	Undetected	F-Secure	Undetected
Kingsoft	Undetected	Lionic	Undetected
Malwarebytes	Undetected	Palo Alto Networks	Undetected
TACHYON	Undetected	TEHTRIS	Undetected
Trapmine	Undetected	TrendMicro	Undetected
TrendMicro-HouseCall	Undetected	VirIT	Undetected
ViRobot	Undetected	Webroot	Undetected
Yandex	Undetected	Zillya	Undetected
Avast Mobile	Unable to process file type	BitDefenderFalx	Unable to process file type

Εικόνα 39. Antivirus που δεν εντόπισαν το κακόβουλο εκτελέσιμο.

5-3-3. Εκτελώντας το payload

Τώρα πρέπει να ρυθμίσουμε έναν listener στη θύρα που καθορίσαμε στο εκτελέσιμο αρχείο. Αυτό το κάνουμε εκκινώντας το Metasploit, χρησιμοποιώντας την εντολή msfconsole στο τερματικό Kali Linux.

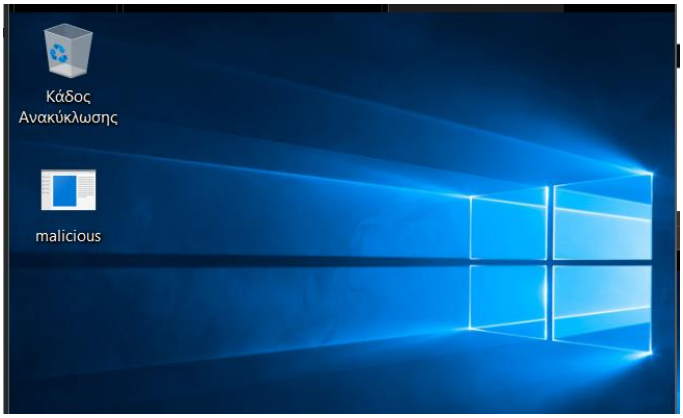
Το παρακάτω στιγμιότυπο οθόνης δείχνει ποιες εντολές πρέπει να δοθούν στο Metasploit. Αρχικά, θα πούμε στο Metasploit να χρησιμοποιήσει τον generic payload handler "multi/handler" χρησιμοποιώντας την εντολή use multi/handler. Στη συνέχεια, θα ρυθμίσουμε το payload ώστε να ταιριάζει με τις ρυθμίσεις του εκτελέσιμου αρχείου χρησιμοποιώντας την εντολή set payload windows/meterpreter/reverse_tcp. Στη συνέχεια, θα ρυθμίσουμε τα LHOST και LPORT με αυτόν τον τρόπο — LHOST 192.168.8.103 και LPORT 4444. Μόλις τελειώσουμε, πληκτρολογούμε "run" ή "exploit" και πατάμε Enter. Ο reverse tcp handler εκκινεί και αναμένει για σύνδεση.



Εικόνα 40. Ρύθμιση listener στο Metasploit

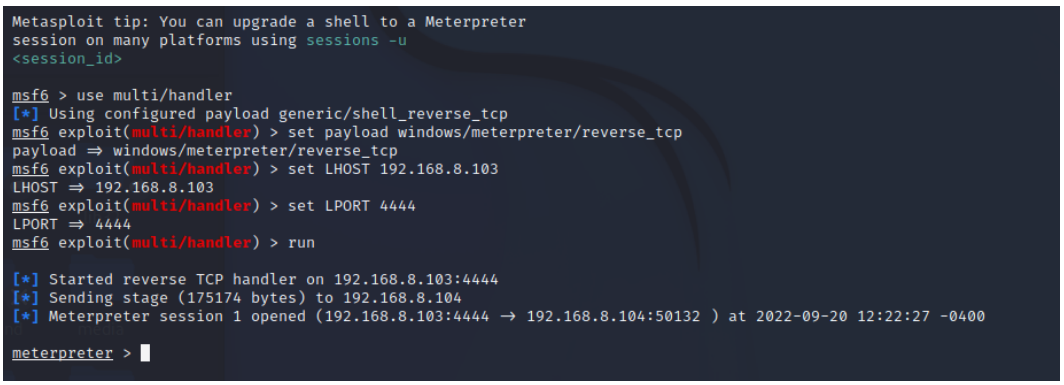
Το επόμενο βήμα είναι να εκτελέσουμε το κακόβουλο αρχείο από τα Windows. Σε μια πραγματική πρακτική κατάσταση, αυτό θα απαιτήσει δεξιότητες κοινωνικής μηχανικής, ώστε να αποσταλεί το αρχείο και να πειστεί ο χρήστης να το εκτελέσει. Ωστόσο, αντιγράφουμε το malicious.exe στο σύστημα Windows 10 στο ίδιο δίκτυο με το σύστημα Kali.

Έχουμε το κακόβουλο αρχείο στον υπολογιστή Windows 10 και το εκτελούμε.



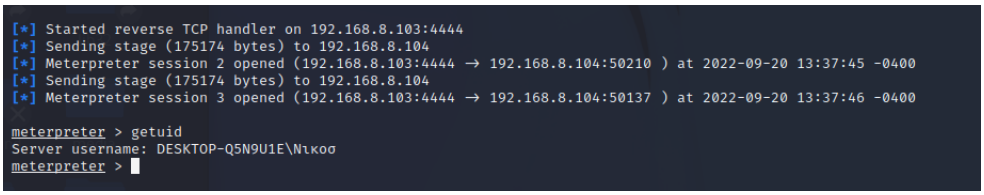
Εικόνα 41. Εκτέλεση κακόβουλου αρχείου στα Windows 10

Το εκτελέσιμο κάνει το payload να εκτελεστεί και να συνδεθεί ξανά με το μηχάνημα επίθεσης (Kali Linux). Αμέσως, λαμβάνουμε ένα Meterpreter session στο Kali Linux μας.



Εικόνα 42. Άνοιγμα Meterpreter session

Δεδομένου ότι το αρχείο δεν εκτελέστηκε ως "διαχειριστής", υπάρχουν εντολές του Meterpreter που δεν μπορούν να εκτελεστούν καθώς θα οδηγούσαν σε μια απάντηση "access denied". Αυτό μπορεί να επιβεβαιωθεί εκτελώντας την εντολή getuid, η οποία μας λέει ότι τρέχουμε ως χρήστης «Νικος».



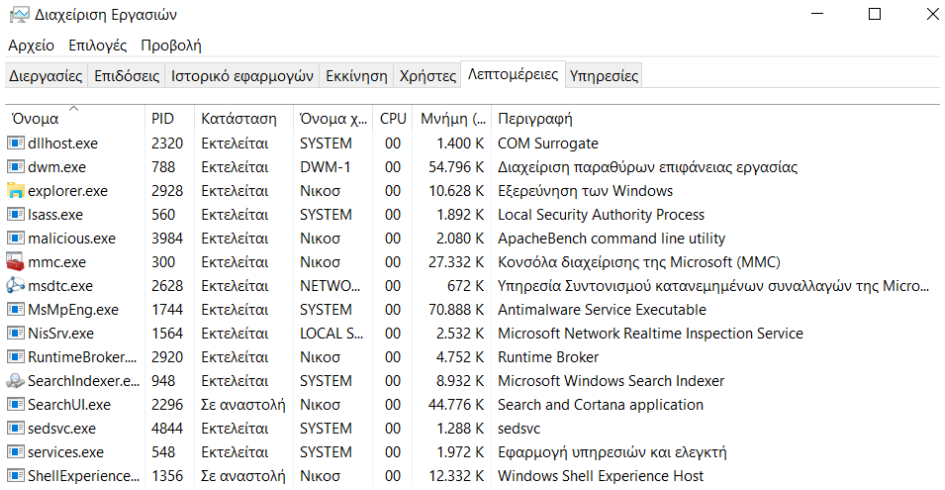
Εικόνα 43. Getuid

5-3-4. Έλεγχος του υπολογιστή του θύματος

Θεωρώντας ότι έχουμε καταφέρει, με τεχνικές social engineering, να πείσουμε τον χειριστή του Η/Υ να εκτελέσει το κακόβουλο λογισμικό και εφόσον έχουμε καταφέρει να το καταστήσουμε μη ανιχνεύσιμο από τα antivirus, είδαμε ότι δημιουργήθηκε ένα session Meterpreter με τα δικαιώματα του τρέχοντος χρήστη.

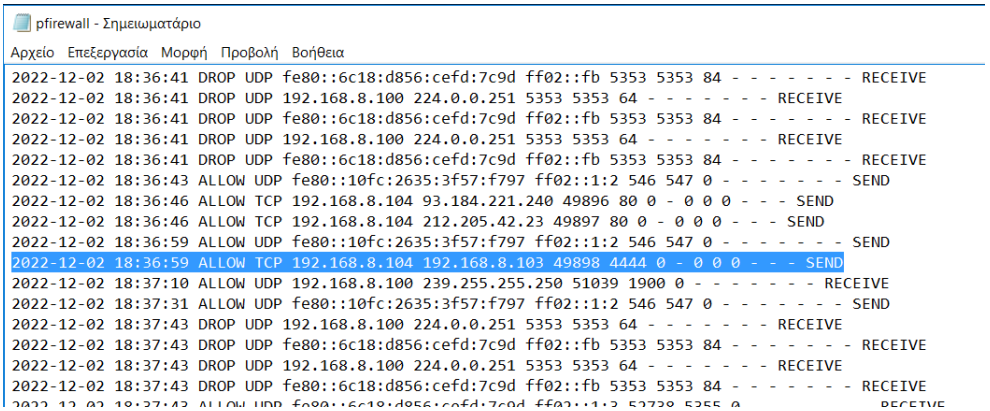
Από την πλευρά του θύματος, οι ενέργειες που εκτελέστηκαν από το κακόβουλο λογισμικό μπορούν να γίνουν αντιληπτές εξετάζονται τις τρέχουσες διεργασίες, το αρχείο καταγραφής του τείχους προστασίας καθώς και τις ενεργές συνδέσεις.

Εξετάζοντας τις τρέχουσες διεργασίες στο Task Manager των Windows 10, παρατηρούμε ότι εκτελείται το κακόβουλο λογισμικό malicious.exe με Process ID (PID) 3984 από τον χρήστη «Νικος» ο οποίος είναι ο συνδεδεμένος χρήστης που εκτέλεσε το λογισμικό. Η περιγραφή του κακόβουλου λογισμικού παρουσιάζεται ως “ApacheBench command line utility”, το οποίο είναι ένα εργαλείο load testing και συγκριτικής αξιολόγησης για διακομιστή πρωτοκόλλου μεταφοράς υπερκειμένου (HTTP). Μπορεί να εκτελεστεί από τη γραμμή εντολών και είναι πολύ απλό στη χρήση. Καταλαμβάνει στην μνήμη χώρο 2080 Kb.



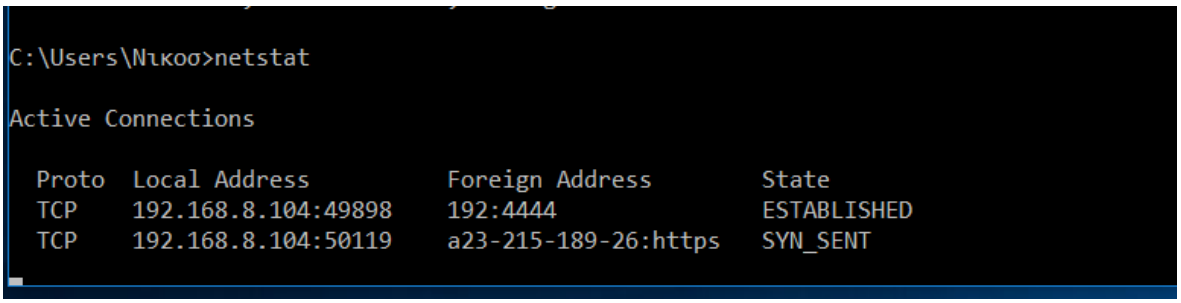
Εικόνα 44. Task Manager Windows 10

Εάν ελέγξουμε το αρχείο καταγραφής του τείχους προστασίας του Windows 10 (pfirewall.log) παρατηρούμε ότι την στιγμή που εκτελέστηκε το κακόβουλο λογισμικό πραγματοποιήθηκε η αποστολή πακέτων TCP από τον υπολογιστή Windows 10 (IP 192.168.8.104) προς τον επιτιθέμενο υπολογιστή (IR 192.168.8.103) στην θύρα 4444 που είχαμε καθορίσει. Η σύνδεση επετράπη από το τείχος προστασίας καθώς είναι εξερχόμενη.



Εικόνα 45. Καταγραφή της σύνδεσης στο αρχείο καταγραφής του firewall.

Από την γραμμή εντολών του Windows 10, με την εντολή netstat βλέπουμε τις ενεργές συνδέσεις την παρούσα στιγμή. Βλέπουμε ότι έχει εγκατασταθεί σύνδεση από τον υπολογιστή Windows 10 (IP 192.168.8.104) προς τον επιτιθέμενο στην θύρα 4444.



Εικόνα 46. Ενεργές συνδέσεις με την εντολή netstat.

6. Σύγχρονες τεχνικές επίθεσης και επίτευξης μη ανιχνευσιμότητας

6-1. Κοινωνική Μηχανική (Social Engineering)

Η κοινωνική μηχανική είναι η προσπάθεια απόκτησης πληροφοριών, πρόσβασης ή εισαγωγής μη εξουσιοδοτημένου λογισμικού στο περιβάλλον μέσω της χειραγώγησης των τελικών χρηστών. Αναφέρεται σε μια σειρά μη τεχνολογικών μεθόδων επίθεσης που χρησιμοποιούνται από κυβερνοεγκληματίες για να χειραγωγήσουν τους χρήστες προκειμένου να παρακάμψουν τα πρωτόκολλα ασφάλειας ή άλλες διαδικασίες, μέσω της εκτέλεσης επιβλαβών ενεργειών ή να τους αναγκάσουν να αποκαλύψουν ευαίσθητες πληροφορίες.

Σύμφωνα με έρευνα του 2019 που διεξήγαγε η Zogby Analytics για λογαριασμό της US National Cyber Security Alliance οι μικρομεσαίες επιχειρήσεις γίνονται όλο και περισσότερο στόχοι των κυβερνοεγκληματιών. Σχεδόν οι μισές (44%) εταιρείες με 251-500 υπαλλήλους δήλωσαν ότι είχαν υποστεί παραβίαση δεδομένων τους τελευταίους 12 μήνες. Η έρευνα διαπίστωσε ότι το 88% των μικρών επιχειρήσεων πιστεύουν ότι είναι τουλάχιστον «κάπως πιθανό» να αποτελέσουν στόχο κυβερνοεγκληματιών, ενώ σχεδόν οι μισές (46%) θεωρούν ότι είναι «πολύ πιθανοί» στόχοι.

Η ζημιά είναι πραγματική και εκτεταμένη. Το FBI εκτιμά ότι, μόνο το 2018, οι αμερικανικές εταιρείες έχασαν περισσότερα από 2,7 δισεκατομμύρια δολάρια λόγων κυβερνοεπιθέσεων, συμπεριλαμβανομένων 1,2 δισεκατομμυρίων δολαρίων που οφείλονται σε υποκλοπές εταιρικών emails (BEC) και υποκλοπές λογαριασμών email (EAC) που επέτρεψαν μη εξουσιοδοτημένες μεταφορές χρημάτων.

Οι περισσότερες τεχνικές κοινωνικής μηχανικής δεν απαιτούν ειδικές τεχνικές δεξιότητες εκ μέρους του εισβολέα, πράγμα που σημαίνει ότι οποιοσδήποτε μπορεί να δράσει σε αυτόν τον χώρο. Στην παραπάνω επίθεση που εκτελέσαμε με χρήση κακόβουλου λογισμικού, θα έπρεπε με κάποια τεχνική κοινωνικής μηχανικής να αποστείλουμε το κακόβουλο αρχείο στον χρήστη και να τον πείσουμε να το εκτελέσει. Μια τέτοια τεχνική θα μπορούσε να είναι το spearphishing, όπως θα δούμε παρακάτω.

6-1-1. Τεχνικές Social Engineering

Υπάρχουν πολλές τεχνικές που εμπίπτουν στον όρο της κοινωνικής μηχανικής στο χώρο της κυβερνοασφάλειας, μεταξύ των πιο γνωστών είναι το spam και το ηλεκτρονικό ψάρεμα (phishing).

Spam

Το Spam αποτελεί οποιαδήποτε μορφή ανεπιθύμητης επικοινωνίας που αποστέλλεται μαζικά. Τις περισσότερες φορές, το spam είναι ένα email που αποστέλλεται σε όσο το δυνατόν περισσότερους παραλήπτες, αλλά μπορεί επίσης να παραδοθεί μέσω άμεσων μηνυμάτων, SMS και κοινωνικών μέσων. Το Spam δεν είναι το ίδιο ένα εργαλείο κοινωνικής μηχανικής, αλλά ορισμένες από τις καμπάνιες χρησιμοποιούν τεχνικές κοινωνικής μηχανικής όπως phishing, spearphishing, vishing, smishing ή διάδοση κακόβουλων επισυναπτόμενων αρχείων ή συνδέσμων.

Phishing

Το ηλεκτρονικό ψάρεμα (phishing) είναι μια μορφή διαδικτυακής επίθεσης στην οποία ο εγκληματίας πλαστοπροσωπεί μια αξιόπιστη οντότητα για να αποσπάσει ευαίσθητες πληροφορίες από το θύμα. Αυτά τα είδη απάτης συνήθως προσπαθούν να δημιουργήσουν μια αίσθηση επείγοντος ή να χρησιμοποιήσουν τακτικές τρόμου για να εξαναγκάσουν το θύμα να συμμορφωθεί με τα αιτήματα του επιτιθέμενου. Οι καμπάνιες ηλεκτρονικού ψαρέματος μπορούν να στοχεύουν μεγάλο αριθμό ανώνυμων χρηστών ή συγκεκριμένου θύματος/θυμάτων.

Spearphishing

Το Spearphishing είναι μια στοχευμένη μορφή ηλεκτρονικού ψαρέματος στο οποίο ο εισβολέας στέλνει προσαρμοσμένα μηνύματα σε μια περιορισμένη ομάδα ανθρώπων, ή ακόμη και σε ένα άτομο, με σκοπό την υποκλοπή των δεδομένων τους ή τη χειραγώγησή τους με σκοπό την εκτέλεση επιβλαβών ενεργειών.

Vishing, Smishing

Το Vishing και το Smishing είναι τεχνικές κοινωνικής μηχανικής που μοιάζουν με το

ηλεκτρονικό ψάρεμα (phishing), αλλά πραγματοποιούνται με άλλα μέσα. Το Vishing (φωνητικό ηλεκτρονικό ψάρεμα - voice phishing) χρησιμοποιεί ψευδείς τηλεφωνικές κλήσεις, ενώ το Smishing (SMS phishing) χρησιμοποιεί μηνύματα κειμένου SMS που περιέχουν κακόβουλους συνδέσμους ή περιεχόμενο.

Πλαστοπρωσοπία

Η πλαστοπροσωπία στην κυβερνοασφάλεια έχει παρόμοιο νόημα με την αντίστοιχη στον φυσικό κόσμο. Οι κυβερνοεγκληματίες ενεργούν στο όνομα ενός αξιόπιστου προσώπου και εξαπατούν τα θύματα για να προβούν σε ενέργειες που βλάπτουν τον εαυτό τους ή τον οργανισμό τους. Ένα τυπικό παράδειγμα είναι ένας εισβολέας που πλαστοπροσωπεί τον Διευθύνοντα Σύμβουλο μιας εταιρείας - όταν ο ίδιος βρίσκεται εκτός γραφείου - παραγγέλλει και εγκρίνει δόλιες συναλλαγές.

Technical Support Scams

Οι απάτες τεχνικής υποστήριξης (Technical Support Scams) είναι συνήθως ψευδείς τηλεφωνικές κλήσεις ή web διαφημίσεις στις οποίες οι επιτιθέμενοι προσφέρουν στα θύματα υποτιθέμενες υπηρεσίες τεχνικής υποστήριξης. Στην πραγματικότητα, οι κυβερνοεγκληματίες προσπαθούν να αποκομίσουν οικονομικά οφέλη από την πώληση ψεύτικων υπηρεσιών και την άρση ανύπαρκτων προβλημάτων.

Scareware

Το Scareware είναι λογισμικό που χρησιμοποιεί διάφορες τεχνικές με σκοπό να προκαλέσει άγχος στα θύματα προκειμένου να τα χειραγωγήσει ώστε να εγκαταστήσουν περαιτέρω κακόβουλο κώδικα στις συσκευές τους, ενώ συνήθως αποσπά πληρωμές για μη λειτουργικό ή κακόβουλο λογισμικό. Ένα τυπικό παράδειγμα είναι ένα ψεύτικο προϊόν προστασίας από ιούς που έχει σχεδιαστεί για να εξαπατήσει τους χρήστες ώστε να πιστεύουν ότι οι συσκευές τους έχουν παραβιαστεί και ότι πρέπει να εγκαταστήσουν ειδικό (συνήθως επιβλαβές) λογισμικό για την εξάλειψη του προβλήματος.

Cyberscams

Οι κυβερνοαπάτες (cyberscams) είναι δόλιες διαδικασίες που χρησιμοποιούν συχνά μία ή ακόμα και πολλές από τις τεχνικές κοινωνικής μηχανικής που περιγράφονται εδώ.

6-1-2. Προστασία από Social Engineering

Υπάρχουν πολλά σημάδια τα οποία μπορεί να υπονοούν μια επίθεση κοινωνικής μηχανικής. Η κακή γραμματική και η ορθογραφία είναι το πιο συνηθισμένο. Το ίδιο και η αίσθηση ότι πρέπει να απαντήσουμε αμέσως που προκύπτει από τα γραφόμενα. Οποιοδήποτε αίτημα για αποκάλυψη ευαίσθητων δεδομένων πρέπει αμέσως να κινήσει υποψίες. Οι αξιόπιστες εταιρείες συνήθως δεν ζητούν κωδικούς πρόσβασης ή προσωπικά δεδομένα μέσω email ή μηνυμάτων κειμένου. Μερικά από τα χαρακτηριστικά σημάδια που υποδηλώνουν κοινωνική μηχανική είναι τα εξής:

- **Φτωχό και γενικόλογο λεξιλόγιο**: Συνήθως, οι εισβολείς δεν δίνουν μεγάλη προσοχή στη λεπτομέρεια, στέλνοντας μηνύματα γεμάτα τυπογραφικά λάθη, με λέξεις να λείπουν και κακή γραμματική. Ένα άλλο γλωσσικό στοιχείο που μπορεί να σηματοδοτήσει μια απόπειρα επίθεσης είναι γενικοί χαιρετισμοί και διατυπώσεις, όπως "Αγαπητέ παραλήπτη" ή "Αγαπητέ χρήστη".
- **Παράξενη διεύθυνση αποστολέα**: Οι περισσότεροι spammers δεν αφιερώνουν χρόνο για να πλαστογραφήσουν το όνομα ή το domain του αποστολέα προκειμένου να φανούν αξιόπιστοι. Επομένως, εάν ένα μήνυμα ηλεκτρονικού ταχυδρομείου προέρχεται από μια διεύθυνση που είναι συνδυασμός τυχαίων αριθμών και χαρακτήρων ή είναι άγνωστη στον παραλήπτη, θα πρέπει να τοποθετηθεί απευθείας στον φάκελο spam και να αναφερθεί στο τμήμα IT.
- **Αίσθηση επείγοντος**: Οι εγκληματίες πίσω από τις εκστρατείες κοινωνικής μηχανικής συχνά προσπαθούν να τρομάξουν τα θύματα με φράσεις που προκαλούν άγχος όπως "στείλτε μας τα στοιχεία σας αμέσως, ή το δέμα σας θα απορριφθεί" ή "εάν δεν ενημερώσετε το προφίλ σας τώρα, θα κλείσουμε τον λογαριασμό σας". Οι τράπεζες, οι εταιρείες αποστολής δεμάτων, τα δημόσια ιδρύματα, αλλά και τα εσωτερικά τμήματα των επιχειρήσεων επικοινωνούν συνήθως με ουδέτερο και πραγματικό τρόπο. Επομένως, εάν το μήνυμα προσπαθεί να ωθήσει τον παραλήπτη να ενεργήσει γρήγορα, είναι πιθανώς κακόβουλο και πιθανώς κρύβει μια επικίνδυνη απάτη.
- **Αιτήματα για ευαίσθητες πληροφορίες**: Οι οργανισμοί, αλλά ακόμη και τα υπόλοιπα τμήματα της εταιρείας συνήθως δεν ζητούν ευαίσθητες πληροφορίες μέσω email ή τηλεφώνου - εκτός εάν η επαφή ξεκίνησε από τον ίδιο τον υπάλληλο.

Κάποιοι από τους τρόπους για την προστασία από επιθέσεις κοινωνικής μηχανικής είναι:

- **Τακτική εκπαίδευση** γύρω από την κυβερνοασφάλεια του συνόλου των υπαλλήλων, συμπεριλαμβανομένων των ανώτατων στελεχών και του προσωπικού του τμήματος IT. Μια τέτοια εκπαίδευση πρέπει να δείχνει ή να προσομοιώνει πραγματικά σενάρια. Η μάθηση πρέπει να περιλαμβάνει δράση εκ μέρους των υπαλλήλων, οι οποίοι θα πρέπει να δοκιμάζονται ενεργά έξω από την αίθουσα εκπαίδευσης: οι τεχνικές κοινωνικής μηχανικής βασίζονται στη χαμηλή επίγνωση της κυβερνοασφάλειας των στόχων τους.
- **Σάρωση για αδύναμους κωδικούς πρόσβασης** που θα μπορούσαν ενδεχομένως να αποτελέσουν ένα backdoor στο δίκτυο του οργανισμού. Επιπλέον, θα πρέπει να προστατέβονται οι κωδικοί πρόσβασης με άλλο ένα επίπεδο ασφάλειας εφαρμόζοντας έλεγχο ταυτότητας πολλών παραγόντων.
- **Εφαρμογή τεχνικών λύσεων** για την αντιμετώπιση της απάτης στις **επικοινωνίες**, έτσι ώστε τα μηνύματα spam και phishing να εντοπίζονται, να τοποθετούνται σε καραντίνα, να εξουδετερώνονται και να διαγράφονται.
- **Δημιουργία κατανοητών πολιτικών ασφάλειας** που μπορούν να χρησιμοποιούν οι εργαζόμενοι και που να τους βοηθούν να εντοπίσουν τι ακριβώς πρέπει να κάνουν όταν συναντήσουν κάποια επίθεση κοινωνικής μηχανικής.
- **Χρήση λύσεων ασφαλείας και εργαλείων διαχείρισης**, για την προστασία των endpoints και των δικτύων του οργανισμού που προσφέρει στους διαχειριστές πλήρεις δυνατότητες επισκόπησης, εντοπισμού και μετριασμού πιθανών κυβερνοαπειλών

6-2. Antivirus Evasion

6-2-1. Στατική – Δυναμική μέθοδοι

Οι συντάκτες κακόβουλου λογισμικού, οι penetration testers, οι red teamers και άλλοι ερευνητές του infosec χρησιμοποιούν τεχνικές antivirus evasion για να παρακάμψουν εφαρμογές προστασίας από ιούς, ώστε ο κώδικας, το script ή το εκτελέσιμο αρχείο να μην επισημανθεί ως κακόβουλο. Οι μέθοδοι διαφυγής μπορούν να κατηγοριοποιηθούν στις ακόλουθες κατηγορίες:

- Στατικές
- Δυναμικές

Οι στατικές είναι μέθοδοι αποφυγής που χρησιμοποιούνται για την παράκαμψη των αλγορίθμων σάρωσης των antivirus. Οι δυναμικές είναι μέθοδοι που χρησιμοποιούνται για την αποφυγή της ανίχνευσης όταν εκτελούνται το εκτελέσιμο αρχείο ή το σενάριο. Οι στατικές μέθοδοι συνήθως περιλαμβάνουν την αλλαγή του δυαδικού αρχείου προκειμένου να αποφευχθεί η αντιστοίχιση μοτίβων, τα CRC, οι υπογραφές κατακερματισμού ή ασαφούς κατακερματισμού ή το γράφημα του κώδικα προκειμένου να δημιουργηθεί σύγχυση στον εντοπισμό υπογραφών βάσει γραφήματος. Οι δυναμικές μέθοδοι περιλαμβάνουν την αλλαγή της λειτουργίας κακόβουλου λογισμικού όταν εκτελείται μέσα σε ένα sandbox ή έναν εξομοιωτή προστασίας από ιούς, ώστε να συμπεριφέρεται σαν να ήταν ένα μη κακόβουλο αρχείο.

6-2-2. Διαίρει και βασίλευε

Ένα σημαντικό μέρος του antivirus evasion είναι να μάθουμε πώς εντοπίζεται ένα συγκεκριμένο κακόβουλο δείγμα. Είναι σημαντικό να ανακαλύψουμε εάν το κακόβουλο λογισμικό εντοπίστηκε μέσω στατικών μέσων ή λόγω ύποπτης συμπεριφοράς κατά την εκτέλεση, ποιος τύπος υπογραφής χρησιμοποιήθηκε, εάν εντοπίστηκε λόγω του γραφήματος κώδικα κ.λπ. Μια μέθοδος για να επιτευχθεί αυτό είναι το «Διαίρει και βασίλευε».

Σε αυτή τη μέθοδο, το δείγμα κακόβουλου λογισμικού χωρίζεται σε μικρότερα αρχεία και στη συνέχεια όλα τα παραγόμενα αρχεία αναλύονται για να βρεθεί το συγκεκριμένο τμήμα που ενεργοποίησε τον εντοπισμό. Για παράδειγμα, εάν έχουμε ένα κακόβουλο αρχείο .gif, θα μπορούσε να χωριστεί σε μέρη 256 byte και, στη συνέχεια, το πρόγραμμα προστασίας από ιούς να σαρώσει όλα τα αρχεία για να βρει πού ακριβώς βρίσκεται ο εντοπισμός. Ωστόσο, αυτή η μέθοδος θα πρέπει να προσαρμοστεί ανάλογα με τον τύπο αρχείου που αναλύουμε. Εάν, για παράδειγμα, έχουμε ένα φορητό εκτελέσιμο αρχείο, ο διαχωρισμός του σε αρχεία ίσου μεγέθους (π.χ. 256 byte) πιθανότατα θα μπέρδευε την κεφαλίδα του PE. Σε αυτή την περίπτωση, αυτή η μέθοδος θα πρέπει να χρησιμοποιείται διαφορετικά.

Το αρχείο πρέπει να χωριστεί σε αρχεία αυξανόμενου μεγέθους (μετατόπιση 0 – 256 byte, μετατόπιση 0 – 512 byte κ.λπ.). Όταν το πρόγραμμα προστασίας από ιούς επισημαίνει ένα συγκεκριμένο αρχείο ως κακόβουλο, γνωρίζουμε σε ποια μετατόπιση ταιριάζει η υπογραφή. Εάν ελέγξουμε το αρχείο με ένα πρόγραμμα επεξεργασίας hex, μπορούμε να βρούμε ποια ακολουθία byte ενεργοποιεί την ανίχνευση, ώστε να μπορούμε να τροποποιήσουμε το δείγμα ανάλογα για να παρακάμψουμε το πρόγραμμα προστασίας από ιούς. Η πιο συνηθισμένη περίπτωση θα ήταν μια στατική υπογραφή που βασίζεται σε αντιστοίχιση προτύπων ή CRC, ίσως σε συνδυασμό με κάποιο αλγόριθμο ασαφούς λογικής.

6-2-3. Signature evasion

Οι υπογραφές δεν είναι πάντα τόσο εύκολο να παρακαμφθούν όπως περιγράφεται στην προηγούμενη ενότητα. Οι μορφές αρχείων των οποίων οι είναι γνωστές, όπως αυτές για αρχεία PE ή OLE2 ή PDF δεν μπορούν να παρακαμφθούν με απλή τροποποίηση του δείγματος σε μια συγκεκριμένη μετατόπιση, επομένως πρέπει να προσεγγιστούν διαφορετικά για μια επιτυχημένη αποφυγή.

Κάθε πρόγραμμα προστασίας από ιούς πρέπει να υποστηρίζει αναλυτές για έναν τεράστιο αριθμό μορφών αρχείων. Ορισμένες μορφές μπορεί να είναι εξαιρετικά περίπλοκες ή κλειστού κώδικα χωρίς αρκετή τεκμηρίωση. Ως αποτέλεσμα, η λειτουργία των αναλυτών μορφής αρχείων μπορεί να διαφέρει μεταξύ των προγραμμάτων προστασίας από ιούς και η πολυπλοκότητα και ο χρόνος εφαρμογής τους θα αυξανόταν πολύ, βοηθώντας ακούσια τους δημιουργούς κακόβουλου λογισμικού.

Για να παρακάμψουμε την ανίχνευση για μια συγκεκριμένη μορφή αρχείου, ένα σημαντικό βήμα είναι να κατανοήσουμε τη μορφή αρχείου για να την τροποποιήσετε με επιτυχία χωρίς να την καταστρέψουμε. Ορισμένες γενικές μέθοδοι για ορισμένες κοινές μορφές αρχείων περιγράφονται στις ακόλουθες ενότητες.

Portable Executable files

Τα φορητά εκτελέσιμα αρχεία (PE) χρησιμοποιούνται πολύ συχνά από συντάκτες κακόβουλου λογισμικού, καθώς είναι αυτοτελή και δεν χρειάζονται προγράμματα κεντρικού υπολογιστή για να εκτελεστούν. Μερικοί τρόποι για επιτυχή τροποποίηση είναι οι εξής:

- **Section names**: Τα ονόματα των ενοτήτων ενός αρχείου PE μπορούν να αλλάξουν χωρίς φόβο ότι θα καταστραφεί το αρχείο, εφόσον είναι μικρότερα ή ίσα με το μέγεθος του πεδίου (8 χαρακτήρες). Ορισμένα προγράμματα προστασίας από ιούς χρησιμοποιούν τα ονόματα των ενοτήτων για να ελέγξουν για συγκεκριμένες ομάδες κακόβουλου λογισμικού, επομένως η αλλαγή τους θα μπορούσε ενδεχομένως να οδηγήσει σε επιτυχή διαφυγή.

- **TimeDateStamp:** Το TimeDateStamp είναι ένα απλό πεδίο χρονικής σφραγίδας σε αρχεία PE. Αυτό δεν απαιτείται από το λειτουργικό σύστημα, επομένως μπορεί να αλλάξει ή ακόμα και να διαγραφεί. Ορισμένα προγράμματα προστασίας από ιούς ελέγχουν τη χρονική σήμανση για να συσχετίσουν αρχεία με συγκεκριμένες οικογένειες κακόβουλου λογισμικού, έτσι ώστε η αλλαγή ή η διαγραφή να έχει το επιθυμητό αποτέλεσμα.

- **Major/MinorLinkerVersion,Major/MinorOperatingSystemVersion, Major/MinorImageVersion**: Αυτά τα πεδία μπορούν να τροποποιηθούν ακριβώς ως TimeDateStamp.

- **AddressOfEntryPoint:** Αυτό το πεδίο μπορεί επίσης να αλλάξει σε NULL έτσι ώστε το σημείο εισόδου του προγράμματος να είναι σε μετατόπιση 0x00. Η ρύθμισή του σε NULL θα μπορούσε να οδηγήσει σε επιτυχή διαφυγή.

- **Μήκος αρχείου**: Η αύξηση του μεγέθους του αρχείου θα μπορούσε επίσης να οδηγήσει σε διαφυγή, καθώς πολλές ευρετικές μηχανές προστασίας από ιούς συχνά απορρίπτουν μεγάλα αρχεία (τα περισσότερα αρχεία κακόβουλου λογισμικού είναι μικρά) προκειμένου να μην υποβαθμίσουν την απόδοση του συστήματος.

Scripts

Γλώσσες scripting όπως το PowerShell των αρχείων JavaScript στο πρόγραμμα περιήγησης μπορούν να χρησιμοποιηθούν για να περιέχουν κακόβουλο κώδικα. Μερικές τεχνικές παράκαμψης ανίχνευσης είναι οι ακόλουθες:

- **Κωδικοποίηση συμβολοσειρών – Obfuscation**: Ένα script μπορεί να παρακάμψει την αποφυγή

απλώς κωδικοποιώντας τους χαρακτήρες συμβολοσειράς ή αναθέτοντάς τους σε μεταβλητές. Στην JavaScript, για παράδειγμα, οι συναρτήσεις διαφυγής ή χωρίς διαφυγή μπορούν να χρησιμοποιηθούν ως εξής: χωρίς διαφυγή ("alert%28%221%22%29") καταλήγει σε "alert('1')". Για το PowerShell, ένα script όπως το Invoke-Obfuscation από τον Daniel Bohannon χρησιμοποιεί διάφορες τεχνικές για να μπερδέψει ένα πλήρες κακόβουλο σενάριο.

- **Εκτέλεση κώδικα on the fly**: Στη JavaScript, εάν ο κώδικας τεθεί ως όρισμα στη συνάρτηση eval, θα εκτελεστεί σαν να καλείται απευθείας. Μια άλλη συνάρτηση, το document.write, μπορεί να χρησιμοποιηθεί για τη δυναμική εγγραφή κώδικα HTML και JavaScript.

- **Junk Code:** Σε πολλές γλώσσες scripting, μια άλλη μέθοδος παράκαμψης του antivirus είναι η χρήση ανεπιθύμητου κώδικα. Ονόματα μεταβλητών και συναρτήσεων με αθώα εμφάνιση, άχρηστες υποθέσεις που κάνουν τη ροή του προγράμματος να φαίνεται πιο περίπλοκη, timeouts και οι περίοδοι αναμονής μπορούν να χρησιμοποιηθούν για την αποφυγή εντοπισμού.

PDF

Το PDF είναι ένα πολύ περίπλοκο πρότυπο μορφής που καθιστά εύκολη την τροποποίηση προκειμένου να παρακάμψει τον εντοπισμό. Για παράδειγμα, ένα αρχείο PDF που περιέχει κακόβουλο JavaScript έχει ετικέτες /JS ή /JavaScript που περιέχουν τα Αντικείμενα JavaScript. Οι χαρακτήρες σε αυτές τις ετικέτες μπορούν να αντικατασταθούν με τις δεκαεξαδικές τιμές τους, όπως /JavaScript -> /#4a#61#76#61#53#63#72#69#70#74. Μια άλλη μέθοδος είναι η επανάληψη αντικειμένων.

Η επανάληψη αντικειμένων έχει ως αποτέλεσμα τη χρήση μόνο του τελευταίου αντικειμένου. Έτσι, η προσθήκη αντικειμένων με τον ίδιο αριθμό θα μπορούσε να οδηγήσει σε evasion. Οι ροές μπορούν επίσης να συμπιεστούν και να κωδικοποιηθούν πολλές φορές με διαφορετικούς κωδικοποιητές και συμπιεστές, προκειμένου να παρακαμφθεί το πρόγραμμα προστασίας από ιούς.

6-2-4. Scanner Evasion

Σε σύγκριση με την Signature Evasion, η Scanner Evasion σημαίνει παράκαμψη της μηχανής προστασίας από ιούς αντί της υπογραφής συγκεκριμένης μορφής. Οι σαρωτές μπορεί να είναι στατικοί (σαρώνουν μόνο αρχεία που είναι γραμμένα στο δίσκο) ή δυναμικοί (ελέγχουν τη συμπεριφορά ενός προγράμματος ή εκτελούν ανάλυση μνήμης). Μια βασική τεχνική για να αποφύγουμε το πρόγραμμα προστασίας από ιούς είναι, όπως αναφέρθηκε προηγουμένως, η αλλαγή του μεγέθους του αρχείου. Ορισμένοι σαρωτές Antivirus απορρίπτουν τον έλεγχο μεγάλων αρχείων για να μην υποβαθμίσουν την απόδοση. Μια άλλη μέθοδος είναι η απενεργοποίηση της ανάλυσης συγκεκριμένου τύπου από το σαρωτή. Εάν ένας αναλυτής PDF, για παράδειγμα, δεν μπορεί να αναλύσει ένα αρχείο, θα το αποκλείσει από όλους τους ελέγχους συγκεκριμένου τύπου υπογραφής και μόνο ίσως επιβάλει γενικό έλεγχο CRC. Μια ακόμη τεχνική είναι η εκτέλεση μη έγκυρων εντολών στον emulator ή η εύρεση οδηγιών που δεν έχουν εφαρμοστεί στον disassembly engine. Σε αυτήν την περίπτωση, η ανάλυση του αρχείου δεν θα είναι δυνατή.

Anti-Emulation

Μια χρήσιμη τεχνική anti-emulation είναι η emulator fingerprinting. Οι εξομοιωτές συνήθως υλοποιούν μόνο τις πιο κοινές λειτουργίες του λειτουργικού συστήματος. Όλες οι άλλες λειτουργίες υλοποιούνται ως στελέχη που επιστρέφουν κωδικοποιημένες τιμές ή δεν υλοποιούνται καθόλου. Είναι επίσης πιθανό μια συνάρτηση να μην εφαρμόζεται σωστά σε έναν εξομοιωτή, οπότε όταν καλείται με έγκυρα ορίσματα, επιστρέφει ένα σφάλμα ή ένα μη έγκυρο αποτέλεσμα.

Anti-disassembling

Μια άλλη αποτελεσματική μέθοδος για την παράκαμψη ενός προγράμματος προστασίας από ιούς είναι η προσπάθεια διακοπής των disassemblers. Τα περισσότερα προγράμματα προστασίας από ιούς χρησιμοποιούν τους δικούς τους ειδικά δημιουργημένους disassemblers ή παλιές εκδόσεις από το diStorm disassembler. Οι CPU σήμερα υποστηρίζουν μεγάλο αριθμό συνόλων εντολών, πολλά από αυτά εν μέρει ή καθόλου τεκμηριωμένα. Ως αποτέλεσμα, οι περισσότεροι disassemblers δεν τους υποστηρίζουν με αποτέλεσμα να αποτυγχάνουν όταν καλούνται. Ένας χρήσιμος τρόπος αξιοποίησης αυτού του γεγονότος είναι να ρυθμίσουμε έναν παλιό disassembler distorm και να προσπαθήσουμε να βρούμε ποιες λειτουργίες δεν υποστηρίζονται. Εάν χρησιμοποιήσουμε τις μη υποστηριζόμενες λειτουργίες με τρόπο που δεν καταστρέφουν το αρχείο κακόβουλου λογισμικού ή δεν επηρεάζουν τη λειτουργικότητα

του αρχείου κακόβουλου λογισμικού, το πρόγραμμα disassembler αποτυγχάνει επειδή δεν μπορεί να αποσυναρμολογήσει σωστά αυτές τις λειτουργίες και το κακόβουλο λογισμικό δεν επισημαίνεται ως κακόβουλο από το πρόγραμμα προστασίας από ιούς.

Anti-debugging

Τα antivirus συχνά συνδέονται σε μια ενεργή διεργασία για να διαβάσουν τη μνήμη της και να ελέγξουν για αντιστοίχιση υπογραφής κακόβουλου λογισμικού. Οι τεχνικές Anti-debugging χρησιμοποιούνται για να αποτρέψουν τους debuggers από την προσκόλληση στη διεργασία του κακόβουλου λογισμικού. Για παράδειγμα, στα Windows, για να συνδεθεί ο debugger σε μια διεργασία, πρέπει να δημιουργήσει ένα remote thread στη διεργασία. Κάθε φορά που δημιουργείται ένα remote thread, ο loader του λειτουργικού συστήματος καλεί callback Thread Local Storage (TLS). Σε αυτήν την περίπτωση, θα μπορούσαμε να ορίσουμε έναν προκαθορισμένο αριθμό νημάτων στην εφαρμογή και να εφαρμόσουμε ένα callback TLS που αυξάνει μια global μεταβλητή. Εάν η τιμή αυτής της μεταβλητής είναι μεγαλύτερη από τον προκαθορισμένο αριθμό νημάτων στην εφαρμογή, σημαίνει ότι δημιουργήθηκε ένα remote thread, επομένως πιθανότατα έχει προσαρτηθεί ένα debugger στη δειργασία. Σε αυτήν την περίπτωση, διατηρούμε τον κώδικα κακόβουλου λογισμικού ανενεργό για να αποφύγουμε τον εντοπισμό.

6-2-5. Heuristic Engines Evasion

Ένα σημαντικό συστατικό των antivirus είναι οι ευρετικές μηχανές (heuristic engines). Οι ευρετικές μηχανές χρησιμοποιούν ρουτίνες ανίχνευσης που αξιολογούν τη συμπεριφορά αντί για συγκεκριμένες υπογραφές για να ελέγξουν εάν κάποιο αρχείο ανήκει σε μια συγκεκριμένη ομάδα κακόβουλου λογισμικού ή μοιράζεται κοινές ιδιότητες με γνωστά αρχεία κακόβουλου λογισμικού. Επί του παρόντος, τα antivirus βασίζονται περισσότερο σε ευρετικές μεθόδους παρά στον κάπως παρωχημένο τρόπο ανίχνευσης υπογραφών. Υπάρχουν τρεις τύποι ευρετικών μηχανών:

- **Στατικά**, τα οποία προσπαθούν να εντοπίσουν στατικά κακόβουλο λογισμικό αποσυναρμολογώντας ή αναλύοντας τις κεφαλίδες του συγκεκριμένου αρχείου.
- **Δυναμικά**, που ελέγχουν τη συμπεριφορά του αρχείου συνδέοντας κλήσεις API ή εκτελώντας το πρόγραμμα σε έναν emulator. Οι δυναμικές ευρετικές μηχανές ονομάζονται επίσης συστήματα πρόληψης εισβολής κεντρικού υπολογιστή (Host Intrusion Prevention Systems - HIPS).
- **Υβριδικά**, τα οποία έχουν και στατικές και δυναμικές ιδιότητες.

Static heuristic engines bypassing

Υπάρχουν δύο προσεγγίσεις για την υλοποίηση static heuristic engine. Η μια χρησιμοποιεί αλγόριθμους μηχανικής μάθησης (π.χ. δίκτυα Bayesian) που ελέγχουν για ομοιότητες μεταξύ ομάδων κακόβουλου λογισμικού χρησιμοποιώντας δεδομένα που συλλέγονται από τα κιτ εργαλείων ομαδοποίησης. Αυτά χρησιμοποιούνται κυρίως σε εργαστήρια κακόβουλου λογισμικού λόγω του μεγάλου αριθμού ψευδώς θετικών στοιχείων και της ανάγκης για αυξημένη χρήση πόρων. Το άλλο χρησιμοποιεί έμπειρα συστήματα, τα οποία είναι ένα σύνολο αλγορίθμων που μιμούνται τη διαδικασία σκέψης και λήψης αποφάσεων ενός ανθρώπινου αναλυτή κακόβουλου λογισμικού (π.χ. ο τύπος αρχείου είναι ασυνήθιστος, ο κώδικας είναι ασαφής, το αρχείο είναι κρυπτογραφημένο, χρησιμοποιεί τεχνικές κατά της ανάλυσης και τα λοιπά.). Αυτή η προσέγγιση είναι πιο κοινή σε προϊόντα προστασίας από ιούς που βασίζονται σε επιτραπέζιους υπολογιστές, όπου απαιτείται καλύτερη απόδοση. Η αποσυναρμολόγηση των βιβλιοθηκών προστασίας από ιούς αποκαλύπτει τις λειτουργίες που είναι υπεύθυνες για την υλοποίηση των ευρετικών μμηχανών.

Dynamic heuristic engines bypassing

Οι dynamic heuristic engines χρησιμοποιούν hooks ή emulators API. Οι emulators αναφέρθηκαν στην προηγούμενη ενότητα. Τα hooks μπορούν να εγκατασταθούν είτε σε kernel-land είτε σε userland.

Τα Userland hooks λειτουργούν με τον ακόλουθο τρόπο:

- Εισάγουν μια βιβλιοθήκη στις διεργασίες του userland.
- Επιλύουν τις λειτουργίες API που πρέπει να παρακολουθούνται.
- Αλλάζουν τις πρώτες εντολές της συνάρτησης με ένα άλμα στον κωδικό προστασίας από ιούς.
- Αφού το πρόγραμμα προστασίας από ιούς ολοκληρώσει την παρακολούθηση, επιστρέφει στο API.

Αν αλλάξουμε τον κωδικό που εκτελεί το αρχικό άλμα στο antivirus, μπορούμε βασικά να

παρακάμψουμε όλους τις στατικές ευρετικές μηχανές που χρησιμοποιούν hooks. Αναλυτικά, για να αποφύγουμε τα hooks userland μπορούμε να επιλύσουμε τις hooked διευθύνσεις στην αρχική βιβλιοθήκη, να διαβάσουμε τα αρχικά byte των hooked συναρτήσεων και να γράψουμε τα byte πίσω στη μνήμη.

Τα kernel-land hooks μπορούν να χρησιμοποιήσουν τις ακόλουθες λειτουργίες για την εκτέλεση του αγκίστρωσης:

- PsSetCreateProcessNotifyRoutine
- PsSetCreateThreadNotifyRoutine
- PsSetLoadImageNotifyRoutine

Εάν το κακόβουλο λογισμικό εκτελείται σε επίπεδο πυρήνα, μπορεί να λάβει έναν δείκτη σε καθεμία από τις παραπάνω λειτουργίες, να βρει όλες τις εγκατεστημένες επιστροφές κλήσης και να αφαιρέσει το hook, έτσι ώστε το antivirus να μπορέσει να παρακολουθεί τη δημιουργία διεργασιών, τη δημιουργία νημάτων κ.λπ. Αφού αφαιρεθούν τα άγκιστρα και η παρακολούθηση έχει απενεργοποιηθεί, ο κακόβουλος κώδικας θα μπορούσε να ξεκινήσει να εκτελείται.

6-3. APT και EDR

Οι επιθέσεις στον κυβερνοχώρο εξελίσσονται συνεχώς τόσο σε πολυπλοκότητα όσο και σε κλίμακα, φτάνοντας σε τέτοιο βαθμό που το Παγκόσμιο Οικονομικό Φόρουμ θεωρεί ότι είναι ο δεύτερος πιο απειλητικός κίνδυνος για το παγκόσμιο εμπόριο την επόμενη δεκαετία. Η παραοικονομία που έχει δημιουργηθεί έχει γίνει τόσο τεράστια σε σημείο να είναι συγκρίσιμη με το μέγεθος των εθνικών οικονομιών. Σε αντίθεση με τις περισσότερες επιθέσεις στον κυβερνοχώρο που έχουν τρόπο λειτουργίας «hit and run», υπάρχουν προηγμένες επίμονες απειλές , οι οποίες είναι ευρέως γνωστές μέσω της συντομογραφίας APT.

Λόγω της φύσης και του αντίκτυπού τους, αυτές οι επιθέσεις έχουν λάβει μεγάλη ερευνητικό ενδιαφέρον, καθώς η ετερογένεια των επιθέσεων εισάγει πολλά ζητήματα για τους παραδοσιακούς μηχανισμούς ασφαλείας. Για παράδειγμα, λόγω του stealth χαρακτήρα τους, τα APT παρακάμπτουν τα antivirus. Ως εκ τούτου, απαιτούνται πιο προηγμένες μέθοδοι για την έγκαιρη ανίχνευσή τους. Τα Endpoint Detection and Response (EDR) παρέχουν μια πιο ολιστική προσέγγιση για την ασφάλεια ενός οργανισμού, καθώς πέρα από τις υπογραφές, τα EDR συσχετίζουν πληροφορίες και συμβάντα από πολλούς υπολογιστές ενός οργανισμού. Επομένως, μεμονωμένα συμβάντα από endpoints συλλέγονται, υποβάλλονται σε επεξεργασία και συσχετίζονται, παρέχοντας στις blue teams μια αναλυτική εικόνα των απειλών στις οποίες εκτίθεται η περίμετρος ενός οργανισμού.

Στην εργασίας των Κ. Πατσάκη και Γ. Καρατζά με τίτλο “An empirical assessment of endpoint detection and response systems against advanced persistent threats attack vectors.” [4] παρουσιάζεται μια προσομοίωση ομάδας APT και αξιολογείται η αποτελεσματικότητα έντεκα πιο προηγμένων EDRs έναντι του εντοπισμού και της πρόληψης των APT. Για το σκοπό αυτό, προσομοιώνονται τέσσερα ενδεικτικά σενάρια επιθέσεων APT σε ένα ελεγχόμενο περιβάλλον χρησιμοποιώντας ένα σύνολο scripted επιθέσεων, τεχνικές διανομής spear-phishing και κακόβουλου λογισμικού που ταιριάζουν με τον τυπικό τρόπο λειτουργίας αυτών των επιθέσεων. Στη συνέχεια εξετάζονται οι απαντήσεις που παράγονται από τα EDR. Τα αποτελέσματα υποδεικνύουν ότι υπάρχουν ακόμη πολλά περιθώρια βελτίωσης, καθώς τα EDR τελευταίας τεχνολογίας αποτυγχάνουν να αποτρέψουν και να καταγράψουν το μεγαλύτερο μέρος των επιθέσεων.

6-3-1. Ορισμός Endpoint Detection and Response Systems

Ο όρος Endpoint Detection and Response (EDR), επινοήθηκε από τον A. Chuvakin [5] το 2013. Τα EDR συλλέγουν δεδομένα από endpoints και τα στέλνουν για αποθήκευση και επεξεργασία σε μια κεντρική βάση δεδομένων. Εκεί, τα συλλεχθέντα συμβάντα, τα δυαδικά αρχεία κ.λπ., θα συσχετιστούν σε πραγματικό χρόνο για τον εντοπισμό και την ανάλυση ύποπτων δραστηριοτήτων στους υπολογιστές που παρακολουθούνται. Έτσι, τα EDR ενισχύουν τις δυνατότητες των SOC καθώς ανακαλύπτουν και ειδοποιούν τόσο τον χρήστη όσο και τις ομάδες απόκρισης έκτακτης ανάγκης για αναδυόμενες απειλές στον κυβερνοχώρο.

Οι κύριες λειτουργίες ενός συστήματος ασφαλείας EDR είναι:

- Παρακολούθηση και συλλογή δεδομένων δραστηριότητας από endpoints που θα μπορούσαν να

υποδηλώνουν απειλή.
- Ανάλυση αυτών των δεδομένων για να εντοπιστούν μοτίβα απειλών.
- Αυτόματη απάντηση σε εντοπισμένες απειλές για την αφαίρεση ή τον περιορισμό τους και ειδοποίηση προσωπικού ασφαλείας.
- Εγκληματολογία και εργαλεία ανάλυσης για την έρευνα των εντοπισμένων απειλών και αναζήτηση ύποπτων δραστηριοτήτων.

6-3-2. Βασικές δυνατότητες των EDR

Detection

Η ανίχνευση απειλών είναι μια θεμελιώδης ικανότητα των EDR. Το θέμα δεν είναι αν θα «επιτεθεί» μια προηγμένη απειλή, το θέμα είναι το πότε. Κατά την είσοδο στο περιβάλλον λειτουργίας, πρέπει να είμαστε σε θέση να ανιχνεύουμε με ακρίβεια την απειλή, ώστε να μπορούμε να την περιορίσουμε, να την αξιολογήσουμε και να την εξουδετερώσουμε. Αυτό δεν είναι εύκολη υπόθεση όταν αντιμετωπίζουμε εξελιγμένο κακόβουλο λογισμικό που μπορεί να είναι εξαιρετικά κρυφό και ικανό να μεταμορφωθεί από μια καλοήθη σε μια κακόβουλη κατάσταση αφού περάσει το σημείο εισόδου.

Με τη συνεχή ανάλυση αρχείων, το EDR μπορεί να επισημαίνει τα μολυσμένα αρχεία με την πρώτη ένδειξη κακόβουλης συμπεριφοράς. Εάν ένα αρχείο θεωρείται αρχικά ασφαλές, αλλά μετά από μερικές εβδομάδες αρχίσει να εμφανίζει δραστηριότητα ransomware, το EDR θα εντοπίσει το αρχείο και θα ξεκινήσει τη διαδικασία αξιολόγησης και ανάλυσης, ενώ θα ειδοποιήσει τον οργανισμό να ενεργήσει.

Containment

Μετά τον εντοπισμό ενός κακόβουλου αρχείου, το EDR πρέπει να μπορεί να περιορίσει την απειλή. Τα κακόβουλα αρχεία στοχεύουν να μολύνουν όσο το δυνατόν περισσότερες διεργασίες, εφαρμογές και χρήστες. Η τμηματοποίηση μπορεί να είναι μια εξαιρετική άμυνα στο data center για να αποφύγουμε την πλευρική μετακίνηση (lateral movement) προηγμένων απειλών. Ενώ η τμηματοποίηση είναι χρήσιμη, ένα ισχυρό EDR μπορεί να βοηθήσει να περιοριστεί ένα κακόβουλο αρχείο πριν ελέγξουμε τις άκρες των τμηματοποιημένων περιοχών του δικτύου. Το Ransomware είναι ένα τεράστιο παράδειγμα του γιατί πρέπει να περιορίζονται οι απειλές. Το ransomware μπορεί να είναι δύσκολο να αφαιρεθεί και από την στιγμή που έχει κρυπτογραφήσει τα δεδομένα, το εργαλείο EDR μπορεί να περιορίσει πλήρως το ransomware για να μετριάσει την βλάβη. Ως πρόσθετο στοιχείο, η ασφάλεια EDR παρέχει τη δυνατότητα απομόνωσης παραβιασμένων endpoints, αποτρέποντας περαιτέρω κρυπτογράφηση μέσω του δικτύου.

Investigation

Μόλις εντοπιστεί και περιοριστεί το κακόβουλο αρχείο, το EDR θα πρέπει να διερευνήσει το περιστατικό. Εάν το αρχείο πέρασε κρυφά στην περίμετρο με την πρώτη προσπάθεια, υπάρχει μια ευπάθεια. Είναι πιθανό η ομάδα πληροφοριών απειλών να μην έχει ξαναδεί τέτοιου είδους προηγμένη απειλή. Ίσως μια συσκευή ή μια εφαρμογή να είναι ξεπερασμένη και να πρέπει να ενημερωθεί. Χωρίς τις κατάλληλες δυνατότητες διερεύνησης, δεν θα αποκτηθεί εικόνα για το πώς πέρασε μια απειλή το δίκτυο. Ως αποτέλεσμα, το δίκτυο είναι πιθανό να αντιμετωπίσει ξανά τις ίδιες απειλές και προβλήματα. Το EDR παρέχει ανά περιστατικό το είδος της ανάλυσης που απαιτείται για την αποκάλυψη αυτών των ζητημάτων και την αποτροπή μελλοντικής εκμετάλλευσης μέσω του ίδιου φορέα απειλής.

Στη διαδικασία έρευνας, το sandboxing είναι μια άλλη κρίσιμη ικανότητα. Το Sandboxing μπορεί να χρησιμοποιηθεί στην περίμετρο, για να βοηθήσει στην παραχώρηση ή άρνηση πρόσβασης, αλλά μπορεί επίσης να χρησιμοποιηθεί αποτελεσματικά μετά το σημείο εισόδου. Sandboxing είναι όταν το αρχείο απομονώνεται σε ένα προσομοιωμένο περιβάλλον και δοκιμάζεται και παρακολουθείται. Το EDR μπορεί να παρέχει sandboxing.

Μέσα σε αυτό το προσομοιωμένο, απομονωμένο περιβάλλον, το EDR θα προσπαθήσει να προσδιορίσει τη φύση του αρχείου χωρίς να διακινδυνεύει ενδεχομένως την ασφάλεια του ευρύτερου περιβάλλοντος. Σε αυτή τη διαδικασία, το EDR μπορεί να κατανοήσει τα χαρακτηριστικά και τη φύση αυτού του κακόβουλου αρχείου, στη συνέχεια να μάθει από αυτό και να προσαρμοστεί για καλύτερη άμυνα έναντι μελλοντικών απειλών.

Elimination

Το πιο προφανές στοιχείο ενός EDR πρέπει να είναι η ικανότητά του να εξαλείφει την απειλή. Ο εντοπισμός, ο περιορισμός και η διερεύνηση μιας απειλής είναι μια καλή αρχή, αλλά αν δεν μπορούμε να την εξαλείψουμε, τότε απλώς συνεχίζουμε να γνωρίζουμε ότι το σύστημά μας έχει

παραβιαστεί. Για να εξαλειφθούν σωστά οι απειλές, το EDR χρειάζεται να απαντήσει σε ερωτήσεις όπως:

- Από πού ξεκίνησε το αρχείο;
- Με ποια διαφορετικά δεδομένα και εφαρμογές αλληλεπίδρασε αυτό το αρχείο;
- Έχει γίνει αναπαραγωγή του αρχείου;

Το να μπορεί να δεί ολόκληρο το χρονοδιάγραμμα ενός αρχείου είναι το κλειδί. Δεν είναι τόσο εύκολο όσο απλά να αφαιρέσουμε το αρχείο που έχουμε παρατηρήσει. Όταν καταργείται το αρχείο, πιθανόν να χρειαστεί να διορθωθούν αυτόματα πολλά μέρη του δικτύου. Για αυτόν τον λόγο, ένα EDR θα πρέπει να παρέχει δεδομένα με δυνατότητα ενέργειας σχετικά με τη διάρκεια ζωής του αρχείου. Εάν ένα εργαλείο EDR έχει αναδρομικές δυνατότητες, αυτά τα δεδομένα με δυνατότητα ενέργειας θα πρέπει να χρησιμοποιηθούν για την αυτόματη αποκατάσταση των συστημάτων στην κατάστασή τους πριν από τη μόλυνση.

6-3-3. Ορισμός Advanced Persistent Threats

Ο όρος Advanced Persistent Threat (APT) χρησιμοποιείται για να περιγράψει μια επίθεση στην οποία ο παράγοντας απειλής καθιερώνει μυστική, μακροπρόθεσμη ανθεκτικότητα στην υπολογιστική υποδομή του θύματος. Ο συνήθης σκοπός είναι η κλοπή δεδομένων ή η διακοπή των υπηρεσιών όταν κρίνεται απαραίτητο από τον παράγοντα της απειλής. Αυτές οι επιθέσεις διαφέρουν από τον τυπικό τρόπο λειτουργίας “hit and run” καθώς μπορεί να εκτείνονται από μήνες έως και χρόνια. Η εκτέλεση μιας επίθεσης APT απαιτεί περισσότερους πόρους από μια τυπική επίθεση διαδικτυακής εφαρμογής και εξαπολύονται από ομάδες υψηλής ειδίκευσης, οι οποίες είναι είτε έθνος-κράτος είτε χρηματοδοτούνται από το κράτος.

Οι επιθέσεις APT διαφέρουν από τις παραδοσιακές απειλές, στο ότι:

- Είναι πολύ πιο σύνθετες.
- Δεν είναι “hit and run” - μόλις διεισδύσει σε ένα δίκτυο, ο δράστης παραμένει για να αποκτήσει όσο το δυνατόν περισσότερες πληροφορίες.
- Εκτελούνται χειροκίνητα (όχι αυτοματοποιημένα) εναντίον ενός συγκεκριμένου στόχου και εξαπολύονται αδιάκριτα εναντίον μιας μεγάλης ομάδας στόχων.
- Συχνά στοχεύουν να διεισδύσουν σε ένα ολόκληρο δίκτυο, και όχι σε ένα συγκεκριμένο τμήμα.

Σε πολλές τέτοιες επιθέσεις, οι φορείς απειλών χρησιμοποιούν κακόβουλο λογισμικό χωρίς αρχείο (fileless malware), έναν συγκεκριμένο τύπο κακόβουλου λογισμικού που δεν αφήνει κανένα κακόβουλο αποτύπωμα στο σύστημα αρχείων του θύματος καθώς λειτουργούν στη μνήμη. Η βασική ιδέα πίσω από αυτό είναι ότι το θύμα θα παρασυρθεί στο άνοιγμα ενός καλοήθους δυαδικού και αυτό το δυαδικό θα χρησιμοποιηθεί για την εκτέλεση ενός συνόλου κακόβουλων εργασιών. Στην πραγματικότητα, υπάρχουν πολλά δυαδικά αρχεία και σενάρια προεγκατεστημένα στα Windows ή μεταγενέστερα ληφθέντα από το λειτουργικό σύστημα και είναι είτε ψηφιακά υπογεγραμμένα είτε στη λίστα επιτρεπόμενων από το λειτουργικό σύστημα και επιτρέπουν την εκτέλεση ενός συνόλου εκμεταλλεύσιμων λειτουργιών. Δεδομένου ότι είναι ψηφιακά υπογεγραμμένα από τη Microsoft, ο User Account Control (UAC) τους επιτρέπει να εκτελούν ένα σύνολο εργασιών χωρίς να ειδοποιούν τον χρήστη.

Οι πιο συνηθισμένες επιθέσεις, όπως η remote file inclusion (RFI), η SQL injection και η cross-site scripting (XSS), χρησιμοποιούνται συχνά από τους επιτιθέμενους για τη δημιουργία ερείσματος σε ένα στοχευμένο δίκτυο. Στη συνέχεια, τα Trojans και τα backdoor shells χρησιμοποιούνται συχνά για να επεκτείνουν αυτό το έδαφος και να δημιουργήσουν μια μόνιμη παρουσία εντός της στοχευμένης περιμέτρου.

6-3-4. Εξέλιξη μιας APT

Μια επιτυχημένη επίθεση APT μπορεί να χωριστεί σε τρία στάδια: 1) διείσδυση δικτύου (network infiltration), 2) επέκταση της παρουσίας του εισβολέα (expansion) και 3) εξαγωγή συγκεντρωμένων δεδομένων (extraction) — όλα αυτά χωρίς να εντοπιστούν.

Infiltration

Στις επιχειρήσεις συνήθως εκτελείται διείσδυση μέσω της παραβίασης ενός από τους τρεις τομείς επίθεσης: web assets, πόρους δικτύου ή εξουσιοδοτημένους χρήστες.

Αυτό επιτυγχάνεται είτε μέσω κακόβουλων μεταφορτώσεων (π.χ. RFI, SQL injection) είτε μέσω επιθέσεων κοινωνικής μηχανικής (π.χ. spear phishing)—απειλές που αντιμετωπίζουν οι μεγάλοι οργανισμοί σε τακτική βάση.

Επιπλέον, οι εισβολείς μπορούν να εκτελέσουν ταυτόχρονα μια επίθεση DDoS εναντίον του στόχου τους. Αυτό χρησιμεύει και ως προπέτασμα καπνού για να αποσπά την προσοχή του προσωπικού του δικτύου και ως μέσο αποδυνάμωσης μιας περιμέτρου ασφαλείας, καθιστώντας ευκολότερη την παραβίασή της.

Μόλις επιτευχθεί η αρχική πρόσβαση, οι εισβολείς εγκαθιστούν γρήγορα ένα backdoor shell—κακόβουλο λογισμικό που παρέχει πρόσβαση στο δίκτυο και επιτρέπει απομακρυσμένες, μυστικές λειτουργίες. Τα backdoors μπορούν επίσης να έρθουν με τη μορφή Trojans που καλύπτονται ως νόμιμα κομμάτια λογισμικού.

Expansion

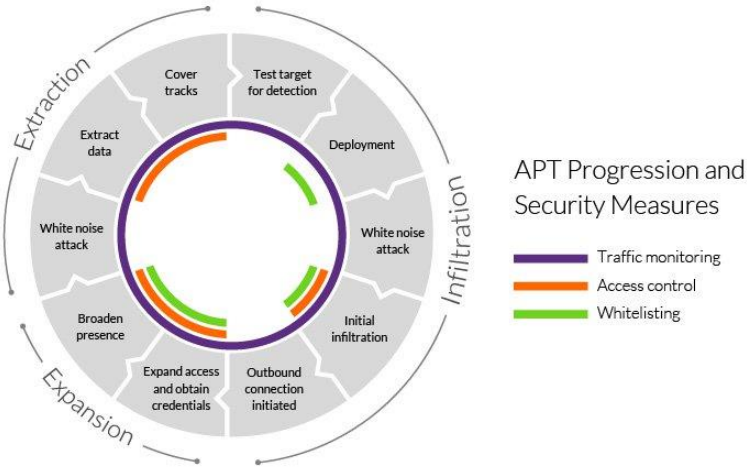
Αφού εδραιωθεί παρουσία στο δίκτυο, οι εισβολείς κινούνται για να διευρύνουν την παρουσία τους μέσα στο δίκτυο. Αυτό περιλαμβάνει την κίνηση ανοδικά στην ιεραρχία ενός οργανισμού, εκμεταλλευόμενοι το προσωπικό με πρόσβαση στα πιο ευαίσθητα δεδομένα. Με αυτόν τον τρόπο, είναι σε θέση να συλλέγουν κρίσιμες επιχειρηματικές πληροφορίες, συμπεριλαμβανομένων πληροφοριών για τη γραμμή προϊόντων, δεδομένων εργαζομένων και οικονομικών αρχείων.

Ανάλογα με τον τελικό στόχο επίθεσης, τα συσσωρευμένα δεδομένα μπορούν να πωληθούν σε μια ανταγωνιστική επιχείρηση, να τροποποιηθούν για να υπονομεύσουν τη σειρά προϊόντων μιας εταιρείας ή να χρησιμοποιηθούν για την εξάλειψη ενός ολόκληρου οργανισμού. Εάν το κίνητρο είναι το σαμποτάζ, αυτή η φάση χρησιμοποιείται για να αποκτήσει διακριτικά τον έλεγχο πολλαπλών κρίσιμων λειτουργιών και να τις χειριστεί σε μια συγκεκριμένη σειρά για να προκαλέσει τη μέγιστη ζημιά. Για παράδειγμα, οι εισβολείς θα μπορούσαν να διαγράψουν ολόκληρες βάσεις δεδομένων μέσα σε μια εταιρεία και στη συνέχεια να διακόψουν τις επικοινωνίες δικτύου προκειμένου να παρατείνουν τη διαδικασία ανάκτησης.

Extraction

Ενώ ένα συμβάν APT βρίσκεται σε εξέλιξη, οι κλεμμένες πληροφορίες συνήθως αποθηκεύονται σε μια ασφαλή τοποθεσία μέσα στο δίκτυο που δέχεται επίθεση. Μόλις συλλεχθούν αρκετά δεδομένα, οι επιτιθέμενοι πρέπει να τα εξαγάγουν χωρίς να εντοπιστούν.

Συνήθως, οι τακτικές λευκού θορύβου χρησιμοποιούνται για να αποσπάσουν την προσοχή της ομάδας ασφαλείας σας, ώστε οι πληροφορίες να μπορούν να μετακινηθούν. Αυτό μπορεί να λάβει τη μορφή επίθεσης DDoS, που δεσμεύει ξανά το προσωπικό του δικτύου ή/και αποδυναμώνει την άμυνα του ιστότοπου για να διευκολύνει την εξαγωγή.



Εικόνα 47. Φάσεις μιας APT επίθεσης και μέτρα ασφαλείας.

6-3-5. Μέτρα ασφαλείας για APT επιθέσεις.

Η σωστή ανίχνευση και προστασία APT απαιτεί μια πολύπλευρη προσέγγιση από την πλευρά των διαχειριστών δικτύου, των παρόχων ασφάλειας και των μεμονωμένων χρηστών. Τα παρακάτω μέτρα υλοποιούνται ως επί το πλείστον από τα συστήματα EDR που είδαμε παραπάνω.

Traffic Monitoring

Η παρακολούθηση της εισερχόμενης και εξερχόμενης κίνησης του δικτύου θεωρείται η βέλτιστη πρακτική για την αποτροπή της εγκατάστασης backdoor και τον αποκλεισμό της εξαγωγής κλεμμένων δεδομένων. Η επιθεώρηση της κυκλοφορίας εντός της περιμέτρου του δικτύου μπορεί επίσης να βοηθήσει στην ειδοποίηση του προσωπικού ασφαλείας για οποιαδήποτε ασυνήθιστη συμπεριφορά που μπορεί να υποδηλώνει κακόβουλη δραστηριότητα.

Ένα τείχος προστασίας εφαρμογών ιστού (Web Application Firewall) που έχει αναπτυχθεί στην άκρη του δικτύου φιλτράρει την κίνηση στους διακομιστές εφαρμογών Ιστού, προστατεύοντας έτσι μια από τις πιο ευάλωτες επιφάνειες επίθεσης. Μεταξύ άλλων λειτουργιών, ένα WAF μπορεί να βοηθήσει στην εξάλειψη επιθέσεων επιπέδου εφαρμογής, όπως επιθέσεις RFI και SQL injection, που χρησιμοποιούνται συνήθως κατά τη φάση διείσδυσης APT.

Οι υπηρεσίες παρακολούθησης εσωτερικής κυκλοφορίας, όπως τα τείχη προστασίας δικτύου (network firewalls), είναι η άλλη πλευρά αυτής της εξίσωσης. Μπορούν να παρέχουν μια αναλυτική προβολή που δείχνει πώς αλληλεπιδρούν οι χρήστες στο δίκτυό σας, ενώ βοηθούν στον εντοπισμό εσωτερικών ανωμαλιών στην κυκλοφορία (π.χ. ακανόνιστες συνδέσεις ή ασυνήθιστα μεγάλες μεταφορές δεδομένων). Το τελευταίο θα μπορούσε να σηματοδοτήσει μια επίθεση APT που λαμβάνει χώρα. Μπορούμε επίσης να παρακολουθούμε την πρόσβαση σε κοινόχρηστα αρχεία ή honeypot του συστήματος.

Τέλος, οι υπηρεσίες παρακολούθησης της εισερχόμενης κυκλοφορίας θα μπορούσαν να είναι χρήσιμες για τον εντοπισμό και την αφαίρεση των backdoor shells. Αυτά μπορούν να αναγνωριστούν με την υποκλοπή απομακρυσμένων αιτημάτων από τους χειριστές.

Application and domain Whitelisting

Η δημιουργία μιας λίστας επιτρεπόμενων εφαρμογών και υπηρεσιών (whitelist) είναι ένας τρόπος ελέγχου των domain από τους οποίους είναι προσβάσιμο το δίκτυό μας, καθώς και των εφαρμογών που μπορούν να εγκατασταθούν από τους χρήστες. Αυτή είναι μια άλλη χρήσιμη μέθοδος για τη μείωση του ποσοστού επιτυχίας των επιθέσεων APT ελαχιστοποιώντας τις διαθέσιμες επιφάνειες επίθεσης. Ωστόσο, αυτό το μέτρο ασφαλείας απέχει πολύ από το να είναι αλάνθαστο, καθώς ακόμη και οι πιο αξιόπιστοι domain μπορούν να τεθούν σε κίνδυνο. Είναι επίσης γνωστό ότι τα κακόβουλα αρχεία συνήθως φτάνουν υπό το πρόσχημα του νόμιμου λογισμικού. Επιπλέον, παλαιότερες εκδόσεις προϊόντων λογισμικού είναι επιρρεπείς σε παραβίαση και εκμετάλλευση.

Για αποτελεσματική δημιουργία whitelist, θα πρέπει να εφαρμόζονται αυστηρές πολιτικές ενημέρωσης για να διασφαλίζεται ότι οι χρήστες εκτελούν πάντα την πιο πρόσφατη έκδοση οποιασδήποτε εφαρμογής που εμφανίζεται στη λίστα.

Access control

Για τους επιτιθέμενους, οι υπάλληλοι μιας εταιρίας αντιπροσωπεύουν συνήθως το μεγαλύτερο και πιο ευάλωτο σημείο στην περίμετρο ασφαλείας. Τις περισσότερες φορές, αυτός είναι ο λόγος για τον οποίο οι χρήστες του δικτύου θεωρούνται από τους εισβολείς ως μια εύκολη πύλη για να διεισδύσουν σε ένα δίκτυο, διευρύνοντας παράλληλα την ισχύ τους εντός της περιμέτρου ασφαλείας.
Οι πιθανοί στόχοι εμπίπτουν σε μία από τις ακόλουθες τρεις κατηγορίες:

- Απρόσεκτοι χρήστες που αγνοούν τις πολιτικές ασφάλειας δικτύου και παραχωρούν εν αγνοία τους πρόσβαση σε πιθανές απειλές.
- Κακόβουλοι πληροφοριοδότες που σκόπιμα κάνουν κατάχρηση των διαπιστευτηρίων χρήστη τους για να παραχωρήσουν πρόσβαση στον δράστη.
- Παραβιασμένοι χρήστες των οποίων τα δικαιώματα πρόσβασης στο δίκτυο παραβιάζονται και χρησιμοποιούνται από εισβολείς.

Η ανάπτυξη αποτελεσματικών ελέγχων απαιτεί μια ολοκληρωμένη ανασκόπηση όλων σε έναν οργανισμό, ιδιαίτερα των πληροφοριών στις οποίες έχουν πρόσβαση. Για παράδειγμα, η ταξινόμηση

δεδομένων με βάση την ανάγκη γνώσης βοηθά να αποκλείσει την ικανότητα ενός εισβολέα να κλέβει τα διαπιστευτήρια σύνδεσης από ένα μέλος του προσωπικού χαμηλού επιπέδου, χρησιμοποιώντας τα για πρόσβαση σε ευαίσθητο υλικό.

Τα βασικά σημεία πρόσβασης δικτύου θα πρέπει να ασφαλίζονται με έλεγχο ταυτότητας δύο παραγόντων (2FA). Απαιτεί από τους χρήστες να χρησιμοποιούν μια δεύτερη μορφή επαλήθευσης κατά την πρόσβαση σε ευαίσθητες περιοχές (συνήθως έναν κωδικό πρόσβασης που αποστέλλεται στην κινητή συσκευή του χρήστη). Αυτό αποτρέπει μη εξουσιοδοτημένους φορείς που είναι μεταμφιεσμένοι ως νόμιμοι χρήστες από το να μετακινούνται στο δίκτυο.

Πρόσθετα μέτρα

Εκτός από τα παραπάνω, αυτά είναι τα μέτρα βέλτιστης πρακτικής που πρέπει να λαμβάνονται κατά την ασφάλιση ενός δικτύου:

- Επιδιόρθωση λογισμικού δικτύου και ευπάθειας του λειτουργικού συστήματος όσο το δυνατόν γρηγορότερα.
- Κρυπτογράφηση απομακρυσμένων συνδέσεων για να αποτρέψει τους εισβολείς από το να τις εκμεταλλευτούν για να διεισδύσουν στο δίκτυο.
- Φιλτράρισμα εισερχόμενων μηνυμάτων ηλεκτρονικού ταχυδρομείου για την αποτροπή επιθέσεων ανεπιθύμητης αλληλογραφίας και phishing που στοχεύουν το δίκτυο.
- Άμεση καταγραφή συμβάντων ασφαλείας για τη βελτίωση των whitelists και άλλων πολιτικών ασφαλείας.

6-3-6. Cyber Kill Chain

Το Cyber kill chain είναι ένα μοντέλο που επιτρέπει στους αναλυτές ασφάλειας να αποδομήσουν μια κυβερνοεπίθεση, παρά την πολυπλοκότητά της, σε αμοιβαία μη αποκλειστικές φάσεις. Το γεγονός ότι κάθε φάση είναι απομονωμένη από τις άλλες επιτρέπει σε κάποιον να αναλύσει κάθε μέρος της επίθεσης ξεχωριστά και να δημιουργήσει μεθόδους μετριασμού και κανόνες ανίχνευσης που μπορούν να διευκολύνουν αμυντικούς μηχανισμούς για την υπό εξέταση επίθεση ή παρόμοιους. Επιπλέον, οι μπλε ομάδες πρέπει να αντιμετωπίσουν μικρότερα προβλήματα, ένα κάθε φορά που είναι πολύ πιο αποδοτικό ως προς τους πόρους από το να αντιμετωπίσουν ένα μεγάλο πρόβλημα στο σύνολό του. Στο μοντέλο του cyber kill chain, θεωρούμε ότι ένας παράγοντας απειλής προσπαθεί να διεισδύσει σε ένα δίκτυο υπολογιστών σε ένα σύνολο διαδοχικών, σταδιακών και προοδευτικών βημάτων. Έτσι, εάν οποιοδήποτε στάδιο της επίθεσης αποτραπεί, τότε η επίθεση δεν θα είναι επιτυχής. Επομένως, τα μικρά βήματα που αναφέραμε παραπάνω είναι ζωτικής σημασίας για την αντιμετώπιση μιας επίθεσης στον κυβερνοχώρο και όσο πιο νωρίς καταφέρει να αποτρέψει μια επίθεση, τόσο μικρότερο αντίκτυπο θα έχει. Αν και το μοντέλο είναι μάλλον ευέλικτο, έχει υποβληθεί σε ορισμένες ενημερώσεις για να ταιριάζει σε πιο στοχευμένες περιπτώσεις χρήσης, π.χ. Internal Cyber Kill Chain για την αντιμετώπιση προβλημάτων με εσωτερικούς κακόβουλους παράγοντες, όπως δυσαρεστημένους ή δύσπιστους υπαλλήλους. Αρχικά αναπτύχθηκε από τη Lockheed Martin το 2011.

6-3-7. Φάσεις του Cyber Kill Chain

Το αρχικό μοντέλο του Cyber Kill Chain της Lockheed Martin περιείχε επτά διαδοχικά βήματα:

Phase 1: Reconnaissance

Κατά τη φάση της Αναγνώρισης, ένας κακόβουλος παράγοντας προσδιορίζει έναν στόχο και διερευνά τρωτά σημεία και αδυναμίες που μπορούν να εκμεταλλευτούν μέσα στο δίκτυο. Ως μέρος αυτής της διαδικασίας, ο εισβολέας μπορεί να συλλέξει διαπιστευτήρια σύνδεσης ή να συγκεντρώσει άλλες πληροφορίες, όπως διευθύνσεις email, αναγνωριστικά χρηστών, φυσικές τοποθεσίες, εφαρμογές λογισμικού και λεπτομέρειες λειτουργικού συστήματος, τα οποία μπορεί να είναι χρήσιμα σε επιθέσεις phishing ή πλαστογράφησης. Σε γενικές γραμμές, όσο περισσότερες πληροφορίες μπορεί να συλλέξει ο επιτιθέμενος κατά τη φάση της αναγνώρισης, τόσο πιο εξελιγμένη και πειστική θα είναι η επίθεση και, ως εκ τούτου, τόσο μεγαλύτερη είναι η πιθανότητα επιτυχίας.

Phase 2: Weaponization

Κατά τη φάση Weaponization, ο εισβολέας δημιουργεί ένα διάνυσμα επίθεσης, όπως κακόβουλο λογισμικό απομακρυσμένης πρόσβασης, ransomware, ιό ή worm που μπορεί να εκμεταλλευτεί

μια γνωστή ευπάθεια. Κατά τη διάρκεια αυτής της φάσης, ο εισβολέας μπορεί επίσης να δημιουργήσει backdoors έτσι ώστε να μπορεί να συνεχίσει να έχει πρόσβαση στο σύστημα εάν το αρχικό σημείο εισόδου του εντοπιστεί και κλείσει από τους διαχειριστές του δικτύου.

Phase 3: Delivery

Σε αυτή την φάση, ο εισβολέας εξαπολύει την επίθεση. Τα συγκεκριμένα βήματα που θα γίνουν θα εξαρτηθούν από το είδος της επίθεσης που σκοπεύουν να πραγματοποιήσουν. Για παράδειγμα, ο εισβολέας μπορεί να στείλει συνημμένα email ή έναν κακόβουλο σύνδεσμο για να ενθαρρύνει τη δραστηριότητα των χρηστών για την προώθηση του σχεδίου. Αυτή η δραστηριότητα μπορεί να συνδυαστεί με τεχνικές κοινωνικής μηχανικής για την αύξηση της αποτελεσματικότητας.

Phase 4: Exploitation

Στη φάση της Εκμετάλλευσης, ο κακόβουλος κώδικας εκτελείται μέσα στο σύστημα του θύματος.

Phase 5: Installation

Αμέσως μετά τη φάση της Εκμετάλλευσης, το κακόβουλο λογισμικό ή άλλος φορέας επίθεσης θα εγκατασταθεί στο σύστημα του θύματος. Αυτό είναι ένα σημείο καμπής στον κύκλο ζωής της επίθεσης, καθώς ο παράγοντας απειλής έχει εισέλθει στο σύστημα και μπορεί πλέον να αναλάβει τον έλεγχο.

Phase 6: Command and Control

Στο Command & Control, ο εισβολέας μπορεί να χρησιμοποιήσει το κακόβουλο λογισμικό για να αναλάβει τον απομακρυσμένο έλεγχο μιας συσκευής ή μιας οντότητας εντός του δικτύου προορισμού. Σε αυτό το στάδιο, ο εισβολέας μπορεί επίσης να εργαστεί για να μετακινηθεί πλευρικά σε όλο το δίκτυο, επεκτείνοντας την πρόσβασή του και δημιουργώντας περισσότερα σημεία εισόδου για το μέλλον.

Phase 7: Actions on Objectives

Σε αυτό το στάδιο, ο εισβολέας λαμβάνει μέτρα για να επιτύχει τους στόχους του, οι οποίοι μπορεί να περιλαμβάνουν κλοπή δεδομένων, καταστροφή, κρυπτογράφηση ή εξαγωγή.

Με την πάροδο του χρόνου, πολλοί ειδικοί στην ασφάλεια πληροφοριών έχουν επεκτείνει την αλυσίδα θανάτωσης για να συμπεριλάβει ένα όγδοο βήμα: τη δημιουργία εσόδων.

Phase 8: Monetization

Σε αυτή τη φάση, ο κυβερνοεγκληματίας επικεντρώνεται στην απόκτηση εισοδήματος από την επίθεση, είτε μέσω κάποιας μορφής λύτρων που θα πληρώσει το θύμα είτε πουλώντας ευαίσθητες πληροφορίες, όπως προσωπικά δεδομένα ή εμπορικά μυστικά.

Σε γενικές γραμμές, όσο νωρίτερα μπορεί ο οργανισμός να σταματήσει την απειλή εντός του κύκλου ζωής της επίθεσης στον κυβερνοχώρο, τόσο λιγότερο κίνδυνο θα αναλάβει ο οργανισμός. Οι επιθέσεις που φτάνουν στη φάση “Command and Control” απαιτούν συνήθως πολύ πιο προηγμένες προσπάθειες αποκατάστασης, συμπεριλαμβανομένων των σαρώσεων σε βάθος του δικτύου και των endpoints για τον προσδιορισμό της κλίμακας και του βάθους της επίθεσης. Ως εκ τούτου, οι οργανισμοί θα πρέπει να λαμβάνουν μέτρα για τον εντοπισμό και την εξουδετέρωση των απειλών όσο το δυνατόν νωρίτερα στον κύκλο ζωής, προκειμένου να ελαχιστοποιηθεί τόσο ο κίνδυνος επίθεσης όσο και το κόστος επίλυσης ενός συμβάντος.

7. Συμπεράσματα - Επίλογος

7-1. Εισαγωγή

Η σπουδαιότητα της ασφάλειας υπολογιστών είναι ίδια με τη σπουδαιότητα της ιδιωτικής ζωής. Από τη στιγμή που σχεδόν όλα τα προσωπικά δεδομένα κάθε σύγχρονου ανθρώπου από τον αριθμό ταυτότητας και του φορολογικού μητρώου έως τις προτιμήσεις του για τη μουσική, τους φίλους και τα προϊόντα που αγοράζει βρίσκονται σε ηλεκτρονική μορφή και μάλιστα στο «σύννεφο» του διαδικτύου, η παραπάνω πρόταση μοιάζει αυταπόδεικτη.

Από το πρώτο worm του δικτύου μέχρι σήμερα οι τεχνικές που χρησιμοποιούνται για την παράνομη πρόσβαση σε ηλεκτρονικούς υπολογιστές έχουν εξελιχθεί και το ίδιο έχουν κάνει και τα κίνητρα για την κατάρριψη των σχετικών ασφαλιστικών δικλείδων. Παράλληλα, η ανάγκη της αγοράς για ολοένα και περισσότερες δυνατότητες, ολοένα και περισσότερα χαρακτηριστικά και η απαίτηση για χαμηλότερες τιμές κάνουν την υπόθεση των ασφαλών προϊόντων να είναι μακριά από τη πραγματικότητα. Τα Υπολογιστικά Συστήματα δεν ξεφεύγουν από αυτή την κατάσταση και η πραγματικότητα κινείται στους ρυθμούς των patches και των διορθώσεων μετά την ανακάλυψη των αδυναμιών που απειλούν τα υπολογιστικά συστήματα που βασίζονται πάνω τους. Τα παραπάνω έρχονται να προστεθούν στην ούτως ή άλλως δύσκολη υπόθεση της εξέτασης ενός προγράμματος για αδυναμίες και στην αδυναμία που υπάρχει ως τώρα για αυτόματη εξέτασή τους με αποτελεσματικό τρόπο. Επιπλέον, διάφορες λύσεις που έχουν προταθεί αν και λύνουν πολλά προβλήματα, δεν μπορούν να υιοθετηθούν λόγω της ανάγκης συμβατότητας με προηγούμενα συστήματα που δεν μπορούν να τις υποστηρίξουν.

Όπως έχει παρουσιαστεί στα παραπάνω κεφάλαια αυτής της εργασίας ένα σύστημα είναι συνεχώς εκτεθειμένο απέναντι σε πολλές απειλές που κρύβονται στο δίκτυο στο οποίο συμμετέχει. Κακόβουλοι χρήστες προσπαθούν συνεχώς να εντοπίσουν και να εκμεταλλευτούν αδυναμίες σε συστήματα με σκοπό να αποκτήσουν έλεγχο σε αυτό. Οι αδυναμίες αυτές μπορεί να αποτελούν είτε μέρος κακών ρυθμίσεων των συστημάτων από του διαχειριστές τους είτε αδυναμίες κάποιων εφαρμογών που τρέχουν στο σύστημα ή ακόμη και αδυναμίες στα πρωτόκολλα που χρησιμοποιούνται.

Όλα τα παραπάνω καθημερινά εξελίσσονται, αλλάζουν και παρουσιάζουν νέα χαρακτηριστικά. Γι' αυτό το λόγο δεν μπορεί ποτέ κάποιος διαχειριστής να εφησυχάσει αλλά πρέπει συνεχώς να βρίσκεται σε εγρήγορση και διαρκή ενημέρωση ώστε να είναι έτοιμος να αντιμετωπίσει τις προκλήσεις των κινδύνων που δέχονται τα συστήματα.

Λόγω αυτών των ιδιαιτεροτήτων της βιομηχανίας της πληροφορικής η κατάσταση στο τομέα της ασφάλειας υπολογιστών αναμένεται να μείνει στάσιμη για αρκετό καιρό ακόμα. Ναι μεν, διορθώνονται καθημερινά πολλά προβλήματα αλλά συνεχώς ανακαλύπτονται καινούργια και μάλιστα σε μια περίοδο που έχουμε τις νέου είδους συσκευές που είναι μικρές και φορητές να καταλαμβάνουν όλο και περισσότερο τις προτιμήσεις των καταναλωτών και να υιοθετούν όλο και περισσότερες λειτουργίες. Η δυνατότητα συναλλαγών μέσω διαδικτυακών υπηρεσιών διευρύνεται και προσελκύει όλο και περισσότερο το ενδιαφέρον τόσο εγκληματιών που σκοπό έχουν την απόκτηση των περιουσιακών στοιχείων των χρηστών όσο και διαφόρων εταιρειών που με σκοπό τη στοχευμένη διαφήμιση θα προσπαθήσουν να βρουν τις προτιμήσεις των ανθρώπων με οποιοδήποτε τρόπο. Επιπλέον, η σχετική νομοθεσία στις διάφορες χώρες του κόσμου σπάνια αντιμετωπίζει το πρόβλημα στις σωστές του διαστάσεις. Η Γερμανία, για παράδειγμα, το 2007 ψηφίζοντας ένα νόμο με σκοπό να αποτρέψει τα ηλεκτρονικά εγκλήματα έβγαλε εκτός νόμου όλα τα προγράμματα που μπορούν να χρησιμοποιηθούν για να «σπάσουν» κωδικούς καθώς και αυτά που μπορούν να ελέγχουν για αδυναμίες σε κάποιο υπολογιστικό σύστημα. Με αυτό τον τρόπο απέτρεψε τους ανθρώπους από το να μπορούν να δοκιμάζουν με νόμιμο τρόπο αν οι υπολογιστές τους μπορούν να προσφέρουν υπηρεσίες με ασφάλεια αλλά απέτρεψε επίσης την δημιουργία καινούργιων προγραμμάτων που προσφέρουν ασφάλεια γιατί πολλές φορές χρησιμοποιούν τις ίδιες μεθόδους με τα κακόβουλα προγράμματα.

7-2. Προτεινόμενα μέτρα

Είδαμε κατά τη διάρκεια της εργασίας ότι στο Metasploit Framework και κατ΄επέκταση και στο Armitage υπάρχει πληθώρα επιλογών σε όλα τα βήματα μιας επίθεσης τα οποία μπορούν να βοηθήσουν στην απόκτηση πρόσβασης σε κάποιο σύστημα ή δίκτυο και αργότερα να δώσουν τη δυνατότητα και τα δικαιώματα σε κάποιον να πράξει όπως θελήσει. Οι δοκιμές έγιναν πάνω σε συστήματα των οποίων η τεχνολογία ήταν ουσιαστικά ξεπερασμένη. Όσο περνάει ο χρόνος και παλιώνει κάποια τεχνολογία, όλο και

περισσότεροι τρόποι βρίσκονται και εφευρίσκονται για να τη διαβάλλει κάποιος. Για αυτό το λόγο υπάρχουν οι ενημερώσεις των λογισμικών, των προγραμμάτων και γενικά των τεχνολογιών που χρησιμοποιούνται, ώστε να καλύπτονται αυτά τα κενά ασφαλείας και να μην δίνουν πρόσβαση στους εισβολείς.

Η καλύτερη λύση για έναν διαχειριστή, είτε δικτύων, είτε ιστοσελίδων, είτε γενικά συστημάτων, είναι να έχει πάντοτε τα πάντα ενημερωμένα και πάντα να ψάχνει εάν έχει βρεθεί κάποιο κενό ασφαλείας ή λάθος στην ενημέρωση πριν την κάνει. Όσο νεότερη η έκδοση των συστημάτων του, τόσο λιγότερες πιθανότητες υπάρχουν για να εισβάλλει κάποιος. Το ίδιο ισχύει και για τις πολιτικές ασφαλείας που ακολουθεί. Θα πρέπει πάντα να προσέχει ότι όλα είναι σωστά οργανωμένα, διότι ακόμα και το μικρότερο κενό σε ένα σύστημα, είναι αρκετά μεγάλο για έναν αποφασισμένο εισβολέα.

7-3. Μελλοντική έρευνα

Η έρευνα στο χώρο της ασφάλειας των υπολογιστών έχει ευρύ φάσμα. Ουσιαστικά, ο ερευνητής στο τομέα μπορεί να ασχοληθεί με οποιοδήποτε κομμάτι αποτελεί ένα υπολογιστή, από το hardware μέχρι την υλοποίηση των πρωτοκόλλων για τη δικτυακή επικοινωνία. Τα Υπολογιστικά Συστήματα όμως είναι ο κοινός παρονομαστής σε όλα αφού είναι ο μόνος τρόπος που ο χρήστης επικοινωνεί με το υλικό του υπολογιστή και είναι αυτά που τελικά πρέπει να παρέχουν μια στέρεα βάση για να χτιστούν όλες οι άλλες εφαρμογές. Αν αυτά χειρίζονται με λανθασμένο τρόπο τα δεδομένα και παρέχουν ευκαιρίες σε κακόβουλους χρήστες να παρανομούν με θύματα τους απλούς χρήστες ο τρόπος που γράφονται τα υπόλοιπα προγράμματα δεν έχει καμία επίπτωση. Ισχύει δηλαδή, ότι η ασφάλεια είναι μια αλυσίδα τόσο ισχυρή όσο και ο πιο αδύναμος κρίκος της και δεν μπορεί να επιτραπεί ο κρίκος αυτός να είναι το κύριο συστατικό ενός υπολογιστή.

Το Metasploit Framework είναι ένα πολυ-λειτουργικό framework και πλούσιο σε δυνατότητες και επιλογές που ανανεώνεται συνέχεια αλλά και αυξάνεται, καθώς συνέχεια δημιουργούνται νέες τεχνολογίες, προγράμματα και λογισμικά, τα οποία δοκιμάζονται από τη μεγάλη κοινότητα του, η οποία αμέσως μετά προσθέτει τους νέους τρόπους επιθέσεων και δοκιμών πίσω στο πλαίσιο. Με αυτόν τον τρόπο, το πλαίσιο δεν παλιώνει ποτέ, αλλά αντιθέτως παραμένει πάντοτε ενημερωμένο και γεμάτο εργαλεία για κάθε χρήση.

Δυστυχώς, καμία λύση δεν μπορεί να προσφέρει πλήρη ασφάλεια σε έναν οργανισμό. Παρά τις σημαντικές προόδους στην ασφάλεια στον κυβερνοχώρο, ένας οργανισμός πρέπει να αναπτύξει ένα ευρύ φάσμα εργαλείων για να παραμείνει ασφαλής και να μην εξαρτάται αποκλειστικά από μία λύση. Επιπλέον, απαιτείται μη αυτόματη αξιολόγηση των αρχείων καταγραφής ασφαλείας και μια ολιστική επισκόπηση των γεγονότων για την αποτροπή επιθέσεων στον κυβερνοχώρο, ειδικά των APT. Λόγω της φύσης του τελευταίου, είναι σημαντικό να τονιστεί ο ανθρώπινος παράγοντας, ο οποίος σε πολλές περιπτώσεις είναι ο πιο αδύναμος κρίκος στην αλυσίδα ασφαλείας και συνήθως χρησιμοποιείται για την απόκτηση αρχικής πρόσβασης σε έναν οργανισμό. Οι οργανισμοί πρέπει να επενδύσουν περισσότερο στις Blue Teams, ώστε να μην εξαρτώνται από τα αποτελέσματα ενός μόνο εργαλείου και να μάθουν να ανταποκρίνονται πέρα από ένα περιορισμένο σύνολο συγκεκριμένων απειλών. Αυτό θα ενισχύσει την ικανότητά τους και θα ανεβάσει αρκετά τον πήχη ώστε να αποτρέψει πολλούς παράγοντες απειλής από το να διεισδύσουν στα συστήματά τους. Τέλος, η εισαγωγή της μηχανικής μάθησης και της τεχνητής νοημοσύνης στην ασφάλεια αναμένεται να βελτιώσει την ισορροπία υπέρ των Blue Teams μετριάζοντας τις επιθέσεις στον κυβερνοχώρο καθώς έχουν ήδη γίνει σημαντικά βήματα από τους ερευνητές. Οι προηγμένοι αλγόριθμοι αναγνώρισης προτύπων και συσχέτισης βρίσκουν το δρόμο τους σε λύσεις ασφαλείας και οι EDR, ειδικότερα, εντοπίζουν ή ακόμη και αποτρέπουν πολλές επιθέσεις στον κυβερνοχώρο στα αρχικά τους στάδια, μειώνοντας τον πιθανό αντίκτυπό τους.

Βιβλιογραφία

[1] Georgia Weidman. “Penetration Testing. A Hands-On Introduction to Hacking”. 2014.

[2] Patrick Engebretson. “The Basics of Hacking and Penetration Testing: Ethical Hacking and Penetration Testing Made Easy (Syngress Basics Series)”. 2013.

[3] Tuyikeze T., and D. Pottas. "An information security policy development life cycle." Proceedings of the South African Information Security Multi-Conference (SAISMC), Port Elizabeth, South Africa. 2011.

[4] Karantzas George, and Constantinos Patsakis. "An empirical assessment of endpoint detection and response systems against advanced persistent threats attack vectors." Journal of Cybersecurity and Privacy 1.3. 2021.

[5] Hutchins, E.M., Cloppert, M.J., Amin, R.M. “Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains. Lead. Issues” Inf. Warf. Secur. Res. 2011.

[6] Peter Kim. “The Hacker Playbook 2. Practical Guide to Penetration Testing”. 2015.

[7] Chuvakin, A. “Named: Endpoint Threat Detection & Response.” https://blogs.gartner.com/anton-chuvakin/2013/07/26/named-endpoint-threat-detection-response/

[8] PCI, Security Standards Council: “Information Supplement: Penetration Testing Guidance”. 2015. https://listings.pcisecuritystandards.org/documents/Penetration_Testing_Guidance_March_2015.pdf

[9] Penetration Testing Flowchart, https://www.pngitem.com/middle/hobxbom_penetration-testing-flow-chart-hd-png-download/

[10] Armitage Update : Graphical cyber attack management tool for Metasploit, https://thehackernews.com/2012/02/armitage-update-graphical-cyber-attack.html

[11] Metasploit, https://www.metasploit.com/

[12] Metasploit Architecture https://www.offensive-security.com/metasploit-unleashed/metasploit-architecture/

[13] Armitage, https://www.offensive-security.com/metasploit-unleashed/armitage/

[14] Kali Linux: Professional Penetration-Testing Distro, http://docs.kali.org/

[15] National Vulnerability database, https://nvd.nist.gov

[16] Metasploitable 2 Exploitability Guide, https://docs.rapid7.com/metasploit/metasploitable-2-exploitability-guide/

[17] Virustotal https://www.virustotal.com/gui/home/upload

[18] Shellter, https://www.shellterproject.com/

[19] Η Κοινωνική Μηχανική στην κυβερνοασφάλεια,
https://www.eset.com/gr/social-engineering-business/

[20] What Is Endpoint Detection and Response (EDR)?,
https://www.trellix.com/en-us/security-awareness/endpoint/what-is-endpoint-detection-and-response.html

[21] What Is Endpoint Detection and Response (EDR)?,
https://www.cisco.com/c/en/us/products/security/endpoint-security/what-is-endpoint-detection-response-edr-medr.html

[22] Advanced persistent threat (APT),
https://www.imperva.com/learn/application-security/apt-advanced-persistent-threat/

[23] The Cyber Kill Chain,
https://www.lockheedmartin.com/en-us/capabilities/cyber/cyber-kill-chain.html