ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ – ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Πρόγραμμα Μεταπτυχιακών Σπουδών
«Κατανεμημένα Συστήματα, Ασφάλεια και Αναδυόμενες Τεχνολογίες Πληροφορίας»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	**Ανάπτυξη μιας πλατφόρμας μέτρησης της εξωτερικής επιθετικής επιφάνειας ενός οργανισμού με εργαλεία ανοικτού κώδικα** **Development of a platform for monitoring the external attack surface of an organization with open source tools**
Ονοματεπώνυμο Φοιτητή	**Γεώργιος Παπαδόπουλος**
Πατρώνυμο	**Αναστάσιος**
Αριθμός Μητρώου	**ΜΠΚΣΑ 20010**
Επιβλέπων	**Παναγιώτης Κοτζανικολάου, Αναπληρωτής Καθηγητής**

Ημερομηνία Παράδοσης **Δεκέμβριος 2022**

Τριμελής Εξεταστική Επιτροπή

(υπογραφή)	(υπογραφή)	(υπογραφή)
Παναγιώτης Κοτζανικολάου Αναπληρωτής Καθηγητής	Δημήτριος Αποστόλου Καθηγητής	Μιχαήλ Ψαράκης Αναπληρωτής Καθηγητής

Ευχαριστίες

Η μεταπτυχιακή διατριβή εκπονήθηκε στα πλαίσια του μεταπτυχιακού προγράμματος «Κατανεμημένα Συστήματα, Ασφάλεια και Αναδυόμενες Τεχνολογίες Πληροφορίας» του τμήματος Πληροφορικής του Πανεπιστημίου Πειραιώς.

Από τη θέση αυτή θα ήθελα να ευχαριστήσω τον επιβλέποντα Αναπληρωτή Καθηγητή κ. Κοτζανικολάου Παναγιώτη για την επιλογή του θέματος και για την επίβλεψη του κατά τη διεκπεραίωση της διατριβής μου. Επίσης θα ήθελα να ευχαριστήσω θερμά τον Πλοίαρχο (Μ) Σπυρίδωνα Παπαγεωργίου, Διευθυντή ΓΕΕΘΑ/Ε5 (Διεύθυνση Κυβερνοάμυνας) για τις συμβουλές και υποδείξεις του κατά την εκπόνηση της παρούσας εργασίας.

Ευχαριστίες θα ήθελα επίσης να εκφράσω στο σύνολο των διδασκόντων καθηγητών του τμήματος Πληροφορικής του Πανεπιστημίου Πειραιώς για την πολύτιμη γνώση που μου παρείχαν, αλλά και για το ότι μου άνοιξαν τους ορίζοντες δίνοντάς μου καινούριες ιδέες και οπτικές γωνίες για να κινηθώ μέσα στον κυβερνοχώρο.

Τέλος, θα ήθελα να ευχαριστήσω τους γονείς μου για τη συνεχή ενθάρρυνση και ηθική συμπαράσταση τους καθ’ όλη τη διάρκεια των σπουδών μου.

ΠΕΡΙΕΧΟΜΕΝΑ

ΕΥΡΕΤΗΡΙΟ ΕΙΚΟΝΩΝ

ΠΕΡΙΛΗΨΗ

Στη σύγχρονη εποχή οι οργανισμοί που χρησιμοποιούν τον κυβερνοχώρο για να προσφέρουν τις υπηρεσίες τους αυξάνονται διαρκώς. Επίσης, οι οργανισμοί που ήδη εκμεταλλεύονται τον κυβερνοχώρο, τείνουν να ψηφιοποιούν ακόμα περισσότερες τις υπηρεσίες τους. Αυτή η τάση προς τον ψηφιακό κόσμο, εάν δεν γίνει ορθά και κυρίως με ασφάλεια, εγκυμονεί πολλούς κινδύνους. Έτσι, ενώ οι οργανισμοί και οι πελάτες τους απολαμβάνουν τις ανέσεις που προσφέρουν οι τεχνολογίες πληροφορικής, η ανταλλαγή ευαίσθητων δεδομένων μέσω του διαδικτύου μπορεί να προσελκύσει πολλούς κακόβουλους χρήστες.

Ο σκοπός της διπλωματικής εργασίας είναι να βοηθήσει στην προσπάθεια της διασφάλισης των ευαίσθητων δεδομένων και τη θωράκιση των υπολογιστικών συστημάτων, επεκτείνοντας τις λειτουργίες ενός υπάρχοντος λογισμικού παρακολούθησης εξωτερικής επιθετικής επιφάνειας, του Sn1per. Το λογισμικό Sn1per διανέμεται σε δύο εκδόσεις. Τη δωρεάν έκδοση ανοικτού κώδικα, η οποία εκτελείται σε περιβάλλον γραμμής εντολών και την επί πληρωμή, επαγγελματική έκδοση κλειστού κώδικα, η οποία εκτελείται σε ένα φιλικό γραφικό περιβάλλον (GUI). Για τους σκοπούς της παρούσας διπλωματικής εργασίας, χρησιμοποιήθηκε η έκδοση του ανοικτού κώδικα, καθώς ο πηγαίος της κώδικας είναι δημόσια διαθέσιμος και μπορεί εύκολα να τροποποιηθεί και να επεκταθεί.

Στα πλαίσια της διπλωματικής εργασίας, αναπτύχθηκαν τρεις νέες λειτουργίες στην έκδοση ανοικτού κώδικα του Sn1per. Η πρώτη αφορά την ανάπτυξη ενός νέου γραφικού περιβάλλοντος, εμπνευσμένο από αυτό που χρησιμοποιεί η επαγγελματική έκδοση του Sn1per, κάνοντας έτσι την δωρεάν έκδοση φιλική ακόμα και στον μη εξειδικευμένο χρήστη. Η δεύτερη λειτουργία επεκτείνει τις δυνατότητες της ήδη υπάρχουσας λειτουργίας του Sn1per στο να εντοπίζει εκτεθειμένες διευθύνσεις ηλεκτρονικού ταχυδρομείου. Συγκεκριμένα, το Sn1per αφού δημιουργήσει μια λίστα με όλες τις εκτεθειμένες διευθύνσεις ηλεκτρονικού ταχυδρομείου του προστατευόμενου υπολογιστικού συστήματος, στη συνέχεια επικοινωνεί αυτοματοποιημένα με την ιστοσελίδα haveibeenpwned ώστε να διαπιστώσει πιθανές διαρροές των κωδικών πρόσβασής τους, προκειμένου να γίνουν οι ανάλογες ενέργειες για την ασφάλισή τους. Η τρίτη λειτουργία αφορά τη δυνατότητα υποβολής ερωτημάτων Google dorks, μέσω πολλαπλών διακομιστών μεσολάβησης, στην περίφημη μηχανή αναζήτησης Google. Έτσι, το Sn1per μπορεί να εντοπίσει τυχόν εσφαλμένα δημοσιοποιημένες ευαίσθητες πληροφορίες για τον προστατευόμενο οργανισμό. Τέλος, έγινε μια πιλοτική εφαρμογή της νέας αναβαθμισμένης έκδοσης του Sn1per, με όλες τις παραπάνω λειτουργίες, τα αποτελέσματα της οποίας αναλύονται στο τελευταίο κεφάλαιο της διπλωματικής διατριβής.

ABSTRACT

In the modern era the organizations that use cyberspace to provide their services are constantly increasing. Also, organizations that already take advantage of cyberspace tend to digitize their services even more. This trend towards the digital world, if not done correctly and especially safely, poses many risks. Thus, while organizations and their customers enjoy the conveniences offered by IT technologies, the exchange of sensitive data over the Internet can attract many malicious users.

The purpose of this master thesis is to assist in the effort of securing sensitive data and shielding computing systems by extending the functionality of an existing external attack surface monitoring software, Sn1per. The Sn1per software is distributed in two versions. The free, open-source version, which runs in a command-line environment, and the paid, professional, closed-source version, which runs in a friendly graphical environment (GUI). For the purposes of this thesis, the open-source version was used, as its source code is publicly available and can be easily modified and extended.

In the context of this thesis, three new functions were developed in the open-source version of Sn1per. The first one concerns the development of a new graphical environment, inspired by the one used by the professional version of Sn1per, thus making the free version friendly even to the non-expert user. The second function extends the capabilities of the Sn1per's already existing functionality to detect exposed email addresses. Specifically, Sn1per after creating a list of all exposed e-mail addresses of the protected computing system, then automatically communicates with the haveibeenpwned website to detect possible leaks of their passwords in order to take the appropriate actions to secure them. The third function concerns the ability to submit Google dorks queries, via multiple proxy servers, to the famous Google search engine. This allows Sn1per to identify any incorrectly revealed sensitive information about the protected organization. Finally, a pilot application of the new upgraded version of Sn1per was carried out, with all the above functions, the results of which are discussed in the last chapter of the master thesis.

ΚΕΦΑΛΑΙΟ 1: ΕΙΣΑΓΩΓΗ

1.1 Κυβερνοχώρος - κυβερνοεπιθέσεις

Στη σύγχρονη εποχή, οι Τεχνολογίες Πληροφορικής και Επικοινωνιών (ΤΠΕ) εξελίσσονται με ραγδαίους ρυθμούς και διεισδύουν ολοένα και περισσότερο στην καθημερινότητα των ανθρώπων. Όλο και περισσότερες επιχειρήσεις και οργανισμοί του δημοσίου και ιδιωτικού τομέα υιοθετούν ή επεκτείνουν τις ήδη υπάρχουσες ψηφιακές υπηρεσίες που προσφέρουν στους καταναλωτές και τους πολίτες [1]. Ακόμα και σε απλές συσκευές καθημερινής χρήσης, όπως σε ψυγεία και κλιματιστικά, ενσωματώνονται υπολογιστικές συσκευές με δυνατότητα σύνδεσης σε τοπικά δίκτυα ή και στο διαδίκτυο [2]. Έτσι, ο κυβερνοχώρος, ο παγκόσμιος χώρος διακίνησης δεδομένων και πληροφοριών, επεκτείνει συνεχώς την παρουσία του σχεδόν σε κάθε προϊόν και υπηρεσία. Επακόλουθα, ζωτικής σημασίας λειτουργίες και υποδομές σε διάφορους τομείς, όπως για παράδειγμα υγείας, παιδείας, οικονομίας, ενέργειας, επικοινωνιών, μεταφορών, εθνικής άμυνας, εξαρτώνται σε μεγάλο βαθμό από την ορθή, συνεχή και ασφαλή λειτουργία των υπολογιστικών συστημάτων και της μεταξύ τους διασύνδεσης [3].

Όμως, η σημερινή πολύπλοκη δομή των πληροφοριακών συστημάτων και η ευρεία χρήση των διαδικτυακών εφαρμογών έχει ένα τίμημα. Όσο ταχύτερα εξαπλώνεται ο κυβερνοχώρος, τόσο αυξάνεται και η εξωτερική του επιφάνεια που μπορεί να δεχτεί ψηφιακές επιθέσεις (κυβερνοεπιθέσεις). Έτσι, η πρόληψη, η διαρκής προσαρμογή και η έγκυρη και έγκαιρη αντίδραση σε τέτοιες απειλές, απαιτούν μια εξειδικευμένη τεχνογνωσία και αποτελούν το θεμέλιο για τη θωράκιση της ασφάλειας των υπολογιστικών συστημάτων, την κυβερνοασφάλεια [4].

1.2 Σκοπός και συνεισφορά της διατριβής

Η παρούσα εργασία ασχολείται με τις τεχνικές που χρησιμοποιούνται σήμερα για την ανίχνευση και την προστασία από επιθέσεις που στοχεύουν στην ψηφιακή εξωτερική επιφάνεια των πληροφοριακών συστημάτων. Ο κλάδος της ανάπτυξης λογισμικού παρακολούθησης της εξωτερικής επιφάνειας και αντιμετώπισης ψηφιακών απειλών είναι νεοσύστατος. Τα διαθέσιμα λογισμικά είναι ακόμα λίγα και κυρίως εμπορικά και κλειστού κώδικα, χωρίς δηλαδή τη δυνατότητα τροποποίησης του πηγαίου κώδικα από τους χρήστες. Υπάρχουν και κάποια λογισμικά ανοικτού κώδικα, που όμως ακόμα δεν είναι φιλικά προς τον χρήστη και πολλές φορές απαιτούνται εξειδικευμένες γνώσεις για την εφαρμογή τους. Ένα δημοφιλές λογισμικό παρακολούθησης εξωτερικής επιφάνειας είναι το Sn1per το οποίο διατίθεται σε δύο εκδόσεις, την ελεύθερη έκδοση ανοικτού κώδικα και την επαγγελματική έκδοση κλειστού κώδικα.

Ο σκοπός της παρούσας εργασίας είναι να συμβάλει στην προσπάθεια της διασφάλισης των ευαίσθητων δεδομένων και τη θωράκιση των υπολογιστικών συστημάτων, επεκτείνοντας τις λειτουργίες του Sn1per. Η ανάπτυξη και η ενσωμάτωση των νέων, εύχρηστων, λειτουργιών στο λογισμικό Sn1per πραγματοποιήθηκαν στην ελεύθερη έκδοση του ανοικτού κώδικα. Με σεβασμό στις αρχές του ανοιχτού κώδικα, η εμπλουτισμένη έκδοση που αναπτύχθηκε στην παρούσα εργασία, παραμένει ελεύθερα προσβάσιμη από όλους και ο πηγαίος κώδικας είναι διαθέσιμος στην ιστοσελίδα αποθετηρίων GitHub στη διεύθυνση https://github.com/cgeopapa/sn1per-sc0pe.

1.3 Δομή της διατριβής

Η διπλωματική διατριβή αποτελείται από επτά κεφάλαια. Το πρώτο κεφάλαιο αποτελεί την εισαγωγή που περιγράφονται ο σκοπός και η συνεισφορά της εργασίας, το λογισμικό που χρησιμοποιήθηκε καθώς και η διεύθυνση στην ιστοσελίδα αποθετηρίων GitHub, όπου είναι διαθέσιμο το λογισμικό που αναπτύχθηκε στα πλαίσια της εργασίας.

Στο Κεφάλαιο 2 δίνεται η περιγραφή της στρατηγικής παρακολούθησης της ψηφιακής εξωτερικής επιφάνειας και παρουσιάζονται τα λογισμικά κλειστού και ανοικτού κώδικα που είναι σήμερα διαθέσιμα. Επίσης, εξηγείται ο λόγος για τον οποίο επιλέχθηκε το συγκεκριμένο λογισμικό

για την ανάπτυξη της πλατφόρμας μέτρησης επιθετικής επιφάνειας ενός οργανισμού με εργαλεία ανοικτού κώδικα.

Στο Κεφάλαιο 3 περιγράφεται το λογισμικό παρακολούθησης εξωτερικής επιθετικής επιφάνειας Sn1per που χρησιμοποιήθηκε στην παρούσα διατριβή.

Στα Κεφάλαιο 4 και 5 περιγράφονται οι τρεις νέες λειτουργίες που αναπτύχθηκαν στην έκδοση ανοικτού κώδικα του Sn1per. Συγκεκριμένα, στο Κεφάλαιο 4 περιγράφεται η ανάπτυξη του νέου γραφικού περιβάλλοντος (GUI), εμπνευσμένο από αυτό που χρησιμοποιεί η επαγγελματική έκδοση του Sn1per, έτσι ώστε η δωρεάν έκδοση να είναι φιλική και στον μη εξειδικευμένο χρήστη. Στη συνέχεια, στο Κεφάλαιο 5 περιγράφεται η ανάπτυξη των λειτουργιών για τον εντοπισμό της διαρροής των κωδικών πρόσβασής των εκτεθειμένων διευθύνσεων ηλεκτρονικού ταχυδρομείου του προστατευόμενου υπολογιστικού συστήματος καθώς και για τον εντοπισμό τυχόν εσφαλμένα δημοσιοποιημένων ευαίσθητων πληροφοριών στο διαδίκτυο σχετικά με τον προστατευόμενο οργανισμό.

Στο Κεφάλαιο 6 παρουσιάζεται η πιλοτική εφαρμογή της νέας αναβαθμισμένης έκδοσης του Sn1per, με όλες τις παραπάνω λειτουργίες, θέτοντας ως προστατευόμενο οργανισμό την ιστοσελίδα ενός ενεργού μεγάλου οργανισμού. Για λόγους προστασίας των προσωπικών δεδομένων, σύμφωνα με τον Γενικό Κανονισμό Προσωπικών Δεδομένων (GDPR), ο Οργανισμός αυτός δεν αναφέρεται και έχει εφαρμοστεί η διαδικασία της ανωνυμοποίησης, όπου αυτό ήταν εφικτό.

Τέλος, στο τελευταίο κεφάλαιο, το έβδομο, παρουσιάζονται τα συμπεράσματα που μπορούν να εξαχθούν από την εργασία.

ΚΕΦΑΛΑΙΟ 2: ΚΥΒΕΡΝΟΑΣΦΑΛΕΙΑ

2.1 Η στρατηγική της κυβερνοασφάλειας

Ο όρος κυβερνοασφάλεια αναφέρεται στη διαδικασία προστασίας υπολογιστικών συστημάτων, δικτύων και λογισμικών από ψηφιακές επιθέσεις, τις λεγόμενες κυβερνοεπιθέσεις (cyberattacks). Αυτές οι επιθέσεις συνήθως αποσκοπούν στο να αποκτήσουν πρόσβαση, να μεταβάλλουν ή ακόμα και να καταστρέψουν ευαίσθητα δεδομένα, να αποσπάσουν χρηματικά ποσά από άλλους χρήστες ή γενικότερα να διακόψουν την ομαλή λειτουργία κάποιας επιχειρηματικής διαδικασίας. Η εφαρμογή αποτελεσματικής κυβερνοασφάλειας στις μέρες μας είναι ιδιαιτέρως δύσκολη καθώς υπάρχουν πάρα πολλές συσκευές συνδεδεμένες στο διαδίκτυο και οι επιτιθέμενοι γίνονται όλο και πιο δημιουργικοί με τις επιθετικές τους στρατηγικές [5].

Μια επιτυχημένη στρατηγική κυβερνοασφάλειας αποτελείται από πολλαπλά επίπεδα προστασίας σε όλα τα υπολογιστικά συστήματα, τα δίκτυα, τα λογισμικά και τα δεδομένα που πρέπει να διατηρηθούν ασφαλή. Σε επίπεδο οργανισμών, οι χρήστες, οι διαδικασίες και η τεχνολογία λογισμικού πρέπει να αλληλοσυμπληρώνονται και να βοηθάει ο ένας τον άλλον, ώστε να δημιουργηθεί μια αποτελεσματική άμυνα από εξωτερικές επιθέσεις που προέρχονται από τον κυβερνοχώρο. Οι χρήστες πρέπει να έχουν μια βασική κατανόηση τυπικών αρχών κυβερνοασφάλειας, όπως την επιλογή ισχυρών κωδικών πρόσβασης, να είναι επιφυλακτικοί με τα συνημμένα σε ληφθέντα μηνύματα ηλεκτρονικού ταχυδρομείου και να διατηρούν αντίγραφα ασφαλείας των δεδομένων τους [6]. Οι οργανισμοί πρέπει να διαθέτουν ένα πλαίσιο για το πώς αντιμετωπίζουν τόσο τις απόπειρες όσο και τις επιτυχείς επιθέσεις από τον κυβερνοχώρο. Ένα καλά δομημένο πλαίσιο περιλαμβάνει διαδικασίες για το πώς να εντοπιστούν επιθέσεις, να προστατευτούν τα συστήματα, να ανιχνευτούν και να αντιμετωπιστούν απειλές καθώς και να επιδιορθωθούν ζημιές και απώλειες από επιτυχημένες επιθέσεις [7]. Τέλος, η τεχνολογία λογισμικού είναι απαραίτητη για την παροχή των απαραίτητων εργαλείων ασφάλειας υπολογιστικών συστημάτων καθώς και για την προστασία τους από επιθέσεις από τον κυβερνοχώρο. Οι κυριότερες οντότητες που πρέπει να προστατεύονται είναι οι συσκευές που έχουν τη δυνατότητα σύνδεσης στο διαδίκτυο, όπως οι υπολογιστές, οι έξυπνες συσκευές και οι δρομολογητές, τα δίκτυα και οι υποδομές νέφους. Η πιο συνήθης τεχνολογία που χρησιμοποιείται για την προστασία αυτών των οντοτήτων περιλαμβάνει τείχη προστασίας (firewall), φιλτράρισμα DNS, προστασία από κακόβουλο και ιομορφικό λογισμικό (antivirus) και λύσεις ασφάλειας ηλεκτρονικού ταχυδρομείου [8].

2.2 Εξωτερική Επιφάνεια

Εξωτερική επιφάνεια είναι το σύνολο όλων των πιθανών σημείων, όπου ένας μη εξουσιοδοτημένος χρήστης μπορεί να αποκτήσει πρόσβαση σε ένα υπολογιστικό σύστημα και να υποκλέψει δεδομένα ή και να προβεί σε άλλες κακόβουλες και παράνομες ενέργειες. Όσο πιο μικρή είναι η επιφάνεια αυτή, τόσο πιο εύκολο ένας οργανισμός μπορεί να προστατευτεί [9].

Εξειδικευμένο προσωπικό των οργανισμών πρέπει διαρκώς να παρακολουθεί την εξωτερική επιφάνεια ώστε να αναγνωρίσει και να αποκλείσει πιθανές απειλές όσο το δυνατόν γρηγορότερα. Έτσι, αποκλείοντας τις πιθανές απειλές, μειώνεται η εκτιθέμενη εξωτερική επιφάνεια και ελαττώνεται ο κίνδυνος μιας επιτυχημένης ψηφιακής επίθεσης. Ωστόσο, η διαδικασία αυτή γίνεται όλο και δυσκολότερη, καθώς, γενικά, οι οργανισμοί επεκτείνουν το ψηφιακό τους αποτύπωμα στον κυβερνοχώρο, υιοθετώντας νέες τεχνολογίες και αγνοώντας, αρκετές φορές, τους επικείμενους κινδύνους.

Η εξωτερική επιφάνεια μπορεί να χωριστεί σε δύο κατηγορίες, την ψηφιακή και τη φυσική. Η ψηφιακή αφορά όλο το υλισμικό και λογισμικό που είναι συνδεδεμένο στο τοπικό δίκτυο του οργανισμού καθώς και στο διαδίκτυο. Η φυσική εξωτερική επιφάνεια αφορά όλα τα μέρη όπου ένας επιτιθέμενος μπορεί να αποκτήσει πρόσβαση με φυσική παρουσία, όπως για παράδειγμα

σε χώρους λειτουργίας των υπολογιστικών συστημάτων, σε σκληρούς δίσκους και USB sticks καθώς και σε κινητές συσκευές [10].

Για την παρακολούθηση της ψηφιακής εξωτερικής επιφάνειας υπάρχουν διαθέσιμα λογισμικά με σκοπό την επιβεβαίωση της ορθής και ασφαλούς ρύθμισης των πληροφοριακών συστημάτων ενός δικτύου. Είτε το δίκτυο είναι οικιακό με ελάχιστες συσκευές, είτε είναι δίκτυο με πολλά υποδίκτυα και με διασυνδεδεμένα δεκάδες ή και εκατοντάδες υπολογιστικά συστήματα, τα λογισμικά αυτά εκτελούν πολλαπλούς και εκτενείς ελέγχους ώστε να επιβεβαιώσουν την ασφάλεια στα δίκτυα αυτά.

2.3 Λογισμικά Παρακολούθησης Εξωτερικής Επιφάνειας

Τα λογισμικά παρακολούθησης εξωτερικής επιφάνειας επιτελούν τη διαδικασία επισκόπησης, αξιολόγησης και ασφάλειας ψηφιακών στοιχείων, γνωστή και ως διαχείριση επιφάνειας επιθέσεων (Attack Surface Management, ASM). Πρόκειται για μια συνεχή και εξελίξιμη διαδικασία καθώς διαρκώς ανακύπτουν νέες μέθοδοι εκμετάλλευσης των ευπαθειών από μη εξουσιοδοτημένους χρήστες ή εισβολείς (hackers). Η ορθή εκτέλεση της διαδικασίας ολοκληρώνεται σε δύο βασικά στάδια. Πρώτον, πρέπει να καταγραφούν όλα τα στοιχεία που αποτελούν την ψηφιακή εξωτερική επιφάνεια και τίθενται υπό τον έλεγχο και προστασία του λογισμικού καθώς και να ομαδοποιηθούν ανάλογα με την κρισιμότητα και την ευαισθησία τους (π.χ., προσωπικά δεδομένα, διαπιστευτήρια συνόδου, εμπιστευτικά έγγραφα) προκειμένου να καθοριστεί η προτεραιότητα ελέγχου. Δεύτερον, πρέπει να υπάρχει μια πλήρης και ακριβής απογραφή του εγκατεστημένου λογισμικού. Με την πλήρη τεκμηρίωση των παραπάνω, ένα λογισμικό παρακολούθησης εξωτερικής επιφάνειας μπορεί να αρχίσει να αναζητά για τις τυπικές μεθόδους διεισδύσεις που είναι γνωστό ότι χρησιμοποιούν οι επιτιθέμενοι hackers. Επειδή, νέες μέθοδοι επίθεσης αναπτύσσονται καθημερινά, τα λογισμικά παρακολούθησης εξωτερικής επιφάνειας δεν πρέπει να παραμένουν στάσιμα. Κανένα εξελιγμένο σύστημα δεν παραμένει στάσιμο. Νέα δεδομένα, νέες εταιρικές διαδικασίες, νέα υπολογιστικά συστήματα, νέα λογισμικά και νέες αναβαθμίσεις σε παλαιότερα συστήματα μπορούν να δημιουργήσουν νέες ευπάθειες και απειλές. Ένα εξελιγμένο λογισμικό παρακολούθησης εξωτερικής επιφάνειας θα πρέπει να διατηρείται ενήμερο για την κατάσταση των υπολογιστικών συστημάτων και των δεδομένων του οργανισμού που προστατεύει [11].

Η παρακολούθηση της ψηφιακής εξωτερικής επιφάνειας είναι ένας αναπτυσσόμενος τομέας στις ΤΠΕ. Σήμερα, η κύρια πρακτική είναι να πραγματοποιείται αυτή η διαδικασία από λογισμικά σάρωσης ευπαθειών (vulnerability assessment). Ωστόσο, χρησιμοποιούνται και λογισμικά ανίχνευσης εισβολών (Intrusion Detection Systems, IDSs), τα οποία μπορούν να εντοπίσουν απώλειες δεδομένων από μεμονωμένα υπολογιστικά συστήματα. Επιπλέον, χρησιμοποιούνται και λογισμικά πρόληψης απώλειας δεδομένων (Data Loss Prevention DPL), κυρίως μέσω της διατήρησης αντιγράφων ασφαλείας, προκειμένου να είναι εύκολη η ανάκτησή τους σε περίπτωση απώλειας.

Τα λογισμικά παρακολούθησης εξωτερικής επιφάνειας είναι είτε κλειστού κώδικα ή ανοικτού κώδικα. Στα λογισμικά κλειστού κώδικα, τον πηγαίο τους κώδικα τον διαχειρίζεται μια οντότητα, είτε αυτή είναι μια εταιρία ή μια ομάδα προγραμματιστών. Δεν μπορεί κανείς να αποκτήσει πρόσβαση στον κώδικα αυτό, ώστε να επεκτείνει ή να προσθέσει νέες λειτουργίες. Εκτός από τα λογισμικά κλειστού κώδικα υπάρχουν τα λογισμικά ανοιχτού κώδικα, τα οποία αποτελούν το επίκεντρο ενδιαφέροντος της παρούσας διπλωματικής εργασίας.

2.3.1 Λογισμικά Κλειστού Κώδικα

Παρακάτω περιγράφονται σύντομα τα πιο ευρέως χρησιμοποιούμενα λογισμικά παρακολούθησης εξωτερικής επιφάνειας κλειστού κώδικα.

Rapid7 InsightVM

Το insightVM είναι ένα λογισμικό διαχείρισης ευπαθειών, που έχει αναπτυχθεί από την εταιρεία Rapid7. Το λογισμικό αυτό έχει τη δυνατότητα σάρωσης όλων των υπολογιστικών συστημάτων που βρίσκονται συνδεδεμένα στο ίδιο δίκτυο καθώς και πιθανά εκτεθειμένα σημεία στο διαδίκτυο. Το εν λόγω εργαλείο είναι βασισμένο σε τεχνολογίες cloud, επιτρέποντας στους χειριστές του να σαρώσουν οποιοδήποτε δίκτυο.

Ένα σημαντικό χαρακτηριστικό που προσφέρει το insightVM είναι το λεγόμενο Project Sonar. Το χαρακτηριστικό αυτό προσφέρει άμεση ενημέρωση στον χειριστή του σχετικά με πιθανές παραβιάσεις δεδομένων που έχουν διαρρεύσει στο διαδίκτυο. Επιπλέον, συλλέγει πληροφορίες σχετικά με κυβερνοεπιθέσεις που εξαπολύθηκαν κατά άλλων οργανισμών, προκειμένου να ενημερώσει τους χειριστές του για πιθανές, παρόμοιες ευπάθειες που μπορεί να υπάρχουν στα δικά τους πληροφοριακά συστήματα [12].

Digital Shadows SearchLight

Το searchlight είναι ένα εργαλείο που έχει αναπτυχθεί από την εταιρία Digital Shadows. Πρόκειται για ένα λογισμικό που παρακολουθεί την εξωτερική επιφάνεια ενός οργανισμού ενημερώνοντας τους χειριστές του για τα ευρήματά του.

Ένα από τα σημαντικότερα χαρακτηριστικά του είναι ότι παραμένει διαρκώς ενημερωμένο σχετικά με νέες πιθανές επιθετικές στρατηγικές. Ο τρόπος που το πετυχαίνει αυτό είναι παρακολουθώντας διάφορα διαδικτυακούς τόπους δημόσιας συζήτησης (forums) ακόμα και σε δικτυακούς τόπους του Dark Web, όπου οι χρήστες τους ανταλλάσσουν σχετικές πληροφορίες. Επιπλέον, παρακολουθεί για πιθανές αναφορές του οργανισμού που προστατεύει, προκειμένου να προειδοποιήσει για πιθανή επικείμενη κυβερνοεπίθεση. Το searchlight, λοιπόν, είναι και ένα web crawler, δηλαδή ένα λογισμικό που περιηγείται σε ιστοσελίδες και αναλύει τις πληροφορίες που βρίσκει σε αυτές. Ο χειριστής του θα πρέπει να το διαμορφώσει κατάλληλα, γνωστοποιώντας του την ονομασία του οργανισμού και τον εξοπλισμό που διαθέτει ώστε να μπορεί να τα προστατέψει και να τα αναγνωρίζει κατά την περιήγησή του σε διαδικτυακά forums. Ένα σημαντικό μειονέκτημά του, λοιπόν, είναι ότι δεν αναγνωρίζει αυτόματα τον εξοπλισμό του οργανισμού, αλλά βασίζεται στον χειριστή για την σωστή ενημέρωσή του [13].

Bugcrowd

Το bugcrowd είναι ένα ιδιαίτερο λογισμικό. Χρησιμοποιεί μια διαφορετική προσέγγιση στην παρακολούθηση της εξωτερικής επιφάνειας. Είναι δημιουργημένο από μια παγκόσμια κοινότητα καλόβουλων hackers.

Το bugcrowd παρακολουθεί τον εξοπλισμό του οργανισμού που προστατεύει, παραμένοντας διαρκώς ενημερωμένο για πιθανές μεταβολές σε αυτόν. Η διαφορετική προσέγγισή του ως προς την ασφάλεια είναι ότι προσκαλεί κάποιους από τους κορυφαίους hackers της κοινότητας να προσπαθήσουν να παραβιάσουν τα συστήματα του οργανισμού που προστατεύει. Πρόκειται για μια κανονική διαδικασία δοκιμής διείσδυσης (penetration test) καθώς το bugcrowd ανταμείβει αυτόν που θα τα καταφέρει. Φυσικά, η διαδικασία αυτή δεν γίνεται με σκοπό να ζημιωθεί ο προστατευόμενος οργανισμός αλλά να ανακαλυφθούν πιθανές ευπάθειες μέσω ενός ανθρωποκεντρικού ελέγχου [14].

Immuni Web Discovery

Το ImmuniWeb Discovery περιλαμβάνει πολλές υπηρεσίες ασφαλείας που μπορούν να επιτελέσουν τον ρόλο παρακολούθησης εξωτερικής επιφάνειας.

Το σύστημα αυτό μπορεί να ανιχνεύσει όλο τον τεχνολογικό εξοπλισμό ενός οργανισμού. Μπορεί να εντοπίσει το λογισμικό που χρησιμοποιείται και να ενημερώσει τον χειριστή για πιθανά περιστατικά ασφαλείας που αφορούν το λογισμικό αυτό. Επιπλέον, όπως και το Searchlight που περιγράφηκε παραπάνω, αναζητά στο Dark Web για εύρεση πληροφοριών που μπορεί να αφορούν τον οργανισμό που προστατεύει. Το ImmuniWeb Discovery ανιχνεύει εσωτερικές

αδυναμίες και εξωτερικές απειλές, παρέχοντας συμβουλές για την επιδιόρθωση και προστασία των ευρημάτων του [15].

2.3.2 Λογισμικά Ανοικτού Κώδικα

Με τον όρο λογισμικό ανοιχτού κώδικα (open source) εννοείται το λογισμικό του οποίου ο πηγαίος κώδικας είναι δημόσια προσβάσιμος. Συγκεκριμένα, οποιοσδήποτε μπορεί να έχει πρόσβαση, να τροποποιήσει και να διανέμει τον κώδικα, όπως αυτός κρίνει κατάλληλα. Τέτοιου είδους λογισμικό αναπτύσσεται με έναν αποκεντρωμένο και συνεργατικό τρόπο, βασιζόμενο στην ανασκόπηση και παραγωγή της κοινότητας που συνήθως δημιουργείται. Τα λογισμικά ανοικτού κώδικα είναι φθηνότερα (τις περισσότερες φορές δωρεάν), πιο ευέλικτα και έχουν, συνήθως, μεγαλύτερη διάρκεια ζωής από τα περισσότερα αντίστοιχα λογισμικά κλειστού κώδικα. Πολλές πακέτα εφαρμογών ανοικτού κώδικα φιλοξενούνται στον ιστότοπο GitHub (https://github.com/), όπου πολλές νέες κοινότητες δραστηριοποιούνται. Ενδεικτικά, αναφέρονται μερικά πολύ γνωστά λογισμικά ανοικτού κώδικα, όπως το λειτουργικό σύστημα Linux, το σύστημα αυτοματισμού Kubernetes και το σύστημα απομακρυσμένου ελέγχου υπολογιστών Ansible.

Η τελική έκδοση του λογισμικού κυκλοφορεί με άδεια χρήσης ανοικτού κώδικα. Η άδεια αυτή, σύμφωνα με τον επίσημα οργανισμό ανοιχτού κώδικα (https://opensource.org/), θα πρέπει να συμμορφώνεται με κάποια συγκεκριμένα κριτήρια. Ειδικότερα, δεν θα πρέπει να υπάρχει κανένας περιορισμός, απαίτηση για αμοιβή ή άλλα δικαιώματα σχετικά με τη διανομή του λογισμικού. Η διανομή του λογισμικού θα πρέπει να συνοδεύεται από τον πηγαίο κώδικα καθώς και την εκτελέσιμη μορφή του. Θα πρέπει επίσης να επιτρέπονται τροποποιήσεις στο λογισμικό. Επιπλέον επιτρέπεται στους δημιουργούς να απαιτούν οι τροποποιημένες εκδόσεις να έχουν διαφορετικό όνομα ή αριθμό κυκλοφορίας από το αρχικό λογισμικό. Η άδεια θα πρέπει να είναι γενική και να μην έχει εξαιρέσεις, περιορισμούς ή να αποκλείει συγκεκριμένα άτομα ή κοινωνικές ομάδες, έργα, τεχνολογίες, ή τομείς έρευνας. Υπάρχουν πολλές διαφορετικές άδειες χρήσης οι οποίες είναι σύμφωνες με τα παραπάνω, δημιουργημένες από πανεπιστήμια, όπως το Ινστιτούτο Τεχνολογίας της Μασαχουσέτης (MIT), μη κερδοσκοπικούς οργανισμούς, όπως τον οργανισμό Apache, ακόμα και κυβερνητικούς οργανισμούς, όπως την Ευρωπαϊκή Ένωση. Φυσικά, η μη συμμόρφωση με τους όρους της άδειας ενός τέτοιου λογισμικού μπορεί να επιφέρει νομικές κυρώσεις.

Είναι σημαντικό να τονιστεί ότι οι μέθοδοι ανάπτυξης λογισμικών ανοικτού κώδικα, τους προσδίδουν κάποιες πολύ σημαντικές ιδιότητες, τις οποίες πολλές φορές τα λογισμικά κλειστού κώδικα στερούνται. Καθώς ο πηγαίος κώδικας είναι ελεύθερα προσβάσιμος και η κοινότητα ανοικτού κώδικα είναι πολύ ενεργή, ο κώδικας ελέγχεται και βελτιώνεται συνεχώς από ομότιμους προγραμματιστές. Με τον τρόπο αυτό αναπτύσσεται διαρκώς ένα αξιόπιστο λογισμικό, το οποίο τις περισσότερες φορές έχει μεγαλύτερη βιωσιμότητα από το αντίστοιχο κλειστού κώδικα. Αυτό συμβαίνει διότι τα λογισμικά κλειστού κώδικα βασίζονται σε μια εταιρεία ή προγραμματιστή για να τα διατηρεί ενεργά και χωρίς δυσλειτουργίες και αστοχίες, ενώ ο ανοιχτός κώδικας συνήθως επιβιώνει περισσότερο από τους αρχικούς δημιουργούς του, επειδή ενημερώνεται συνεχώς μέσω ενεργών κοινοτήτων. Τέλος, στη σύγχρονη εποχή, όπου η ανησυχία για την ιδιωτικότητα και τα προσωπικά δεδομένα διαρκώς αυξάνεται, ο ανοιχτός κώδικας επιτρέπει στον οποιοδήποτε να ελέγξει και να επιβεβαιώσει επακριβώς τα δεδομένα που χρησιμοποιεί κάποιο λογισμικό.

Η ιδέα του ανοιχτού κώδικα έχει πλέον εξελιχθεί σε κίνημα και τρόπο εργασίας που ξεπερνάει τα όρια της ανάπτυξης λογισμικού. Το κίνημα ανοιχτού κώδικα χρησιμοποιεί τις αξίες του μοντέλου της αποκεντρωμένης ανάπτυξης λογισμικού ώστε να βρίσκει νέους τρόπους για να λύνει προβλήματα σε κοινότητες και σε βιομηχανίες εκτός του κλάδου της Πληροφορικής [16].

Παρακάτω αναλύονται λογισμικά ανοικτού κώδικα, καταλήγοντας στο λογισμικό που έχει επιλεγεί για την εκπόνηση της διπλωματικής εργασίας.

OWASP Zed Attack Proxy

Το Zed Attack Proxy είναι ένα λογισμικό ανοιχτού κώδικα το οποίο πραγματοποιεί αναζήτηση σε μια ιστοσελίδα για τις δέκα πιο σημαντικές ευπάθειες του OWASP (Open Web Application Security Project). Ο OWASP είναι ένας μη κερδοσκοπικός οργανισμός με στόχο να βελτιώσει την ασφάλεια λογισμικού. Μέσω λογισμικών ανοιχτού κώδικα που αναπτύσσει με τη βοήθεια της κοινότητας που έχει δημιουργήσει και με κορυφαία εκπαιδευτικά συνέδρια, ο OWASP αποτελεί μια σημαντική πηγή γνώσεων για πολλούς προγραμματιστές και εταιρείες τεχνολογίας πληροφορικής.

Το Zed Attack Proxy μπορεί να μετατραπεί σε λογισμικό παρακολούθησης εξωτερικής επιφάνειας με τη χρήση μιας πρόσθετης εφαρμογής, που τιτλοφορείται ως Attack Surface Detector. Η πρόσθετη εφαρμογή εντοπίζει τα υπολογιστικά συστήματα και τις δικτυακές εφαρμογές που εκτελούνται σε αυτά και αποθηκεύει τον πηγαίο κώδικά τους. Στη συνέχεια, αναλύει και προσπαθεί να εντοπίσει ευπάθειες στους πηγαίους κώδικες και τη δομή τους. Παράλληλα, παρακολουθεί τις εφαρμογές αυτές για τυχόν αλλαγές που μπορεί να επιφέρουν νέες ευπάθειες [17].

Sn1per

Το Sn1per είναι ένα λογισμικό, κατάλληλο για να αυτοματοποιεί την παρακολούθηση της ψηφιακής εξωτερικής επιφάνειας ενός οργανισμού. Μπορεί να εκτελεστεί μόνο σε περιβάλλον λειτουργικού συστήματος Linux και διανέμεται σε δύο εκδόσεις. Την ελεύθερη έκδοση ανοικτού κώδικα και την επαγγελματική έκδοση, κλειστού κώδικα, η οποία απαιτεί την αγορά ετήσιας συνδρομής.

Το Sn1per είναι το λογισμικό που επιλέχθηκε για την εκπόνηση της παρούσας διπλωματικές εργασίας και περιγράφεται αναλυτικά στο επόμενο κεφάλαιο. Στο πλαίσιο της εργασίας έχει χρησιμοποιηθεί η έκδοση ανοικτού κώδικα, που εκτελείται σε περιβάλλον γραμμής εντολών. Έχοντας, λοιπόν, τη δυνατότητα τροποποίησης του ανοικτού κώδικα, αναπτύχθηκαν και ενσωματώθηκαν νέες λειτουργίες, επεκτείνοντας έτσι τις δυνατότητες του λογισμικού Sn1per. Ο λόγος που επιλέχθηκε το συγκεκριμένο λογισμικό είναι διότι ανάμεσα στις ελάχιστες επιλογές λογισμικών ανοιχτού κώδικα, θεωρείται ως το καλύτερο.

ΚΕΦΑΛΑΙΟ 3: ΤΟ ΛΟΓΙΣΜΙΚΟ ΠΑΡΑΚΟΛΟΥΘΗΣΗΣ ΕΞΩΤΕΡΙΚΗΣ ΕΠΙΦΑΝΕΙΑΣ SN1PER

Στο κεφάλαιο αυτό περιγράφεται αναλυτικά η λειτουργία του λογισμικού Sn1per. Όπως αναφέρθηκε προηγουμένως, το Sn1per είναι διαθέσιμο σε δύο εκδόσεις, την επί πληρωμή έκδοση κλειστού κώδικα και την ελεύθερη έκδοση ανοικτού κώδικα. Μεταξύ των δύο εκδόσεων δεν υπάρχουν λειτουργικές διαφορές, αφού και οι δύο χρησιμοποιούν το ίδιο λογισμικό με τις ίδιες δυνατότητες, που είναι αυτό του ανοικτού κώδικα. Ωστόσο, η έκδοση κλειστού κώδικα εφαρμόζεται μέσω ενός φιλικού γραφικού περιβάλλοντος διαχείρισης των εντολών του Sn1per, που φιλοξενείται σε έναν φυλλομετρητή. Ουσιαστικά, λοιπόν, η έκδοση κλειστού κώδικα πλεονεκτεί στη διευκόλυνση που προσφέρει στη χρήση του, χωρίς όμως να διαθέτει επιπλέον λειτουργίες από την αντίστοιχη έκδοση ανοιχτού κώδικα.

3.1 Αναλυτική Περιγραφή του Λογισμικού Sn1per

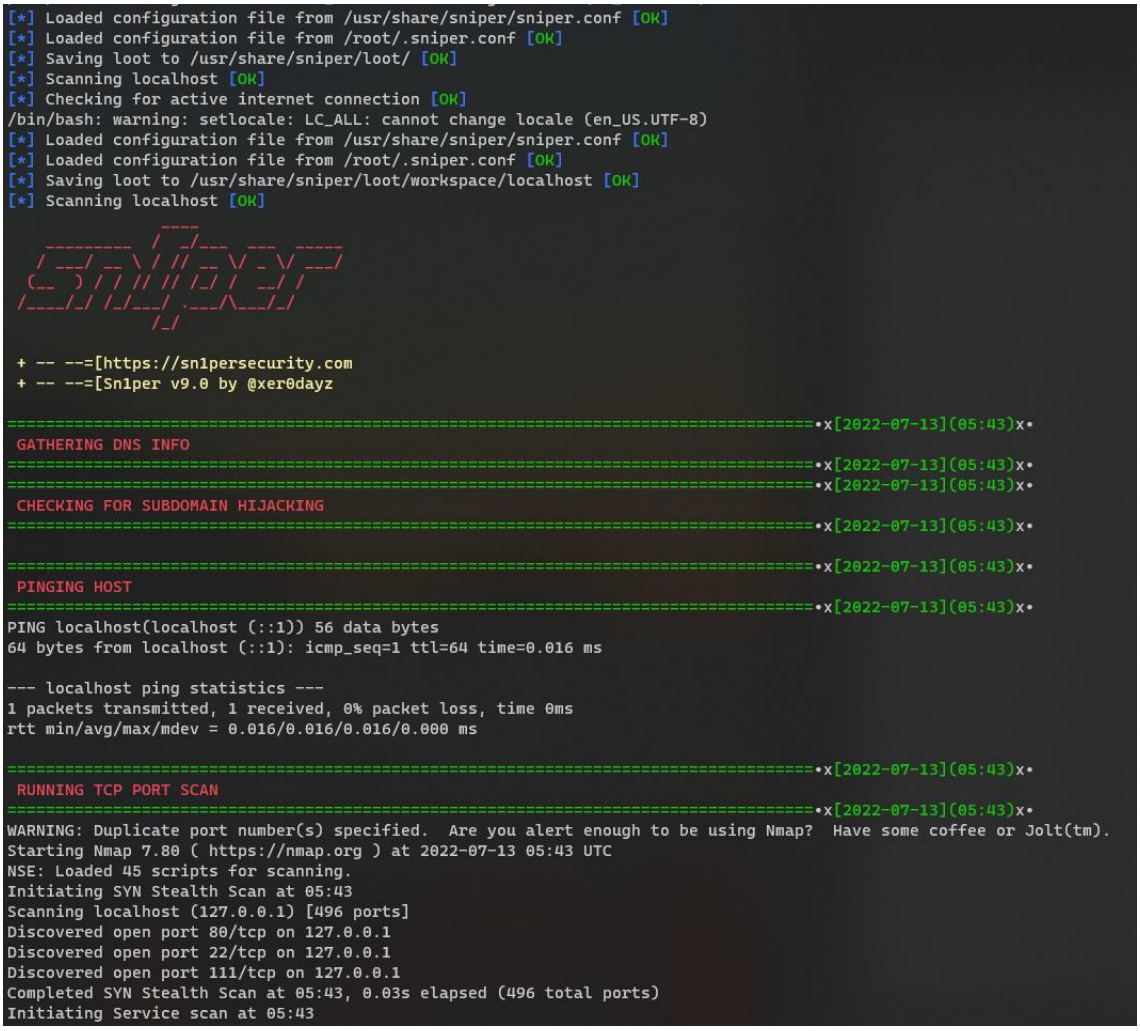
Το Sn1per ενορχηστρώνει και συνδυάζει πολλά άλλα εργαλεία ανοικτού κώδικα κατά την εκτέλεση μιας διαδικασίας σάρωσης του προστατευόμενου υπολογιστικού συστήματος. Με την ολοκλήρωση της σάρωσης, τα ευρήματα κάθε εργαλείου που έχει ρυθμιστεί να εκτελεστεί, αρχειοθετούνται σε ξεχωριστά αρχεία. Στον πυρήνα του, το Sn1per, είναι μια εκτενής ακολουθία εντολών του λειτουργικού συστήματος Linux, που προκειμένου να αυτοματοποιηθεί η εκάστοτε διαδικασία σάρωσης, η οποία ακολουθία εισάγεται με γραμμική ροή σε αρχεία scripts στο κέλυφος bash. Η σύνθεση των εντολών της ακολουθίας αυτής ρυθμίζει τη σειρά και τον τρόπο εκτέλεσης των επιπρόσθετων εργαλείων καθώς και την ανακατεύθυνση της εξόδου των αποτελεσμάτων τους σε αρχεία. Υπάρχουν ξεχωριστά αρχεία ροής εντολών (scripts) για την εγκατάσταση, την απεγκατάσταση, την εκτέλεση καθώς και τους εκάστοτε διαφορετικούς τύπους λειτουργίας, οι οποίοι θα αναλυθούν στο υποκεφάλαιο 3.3. Η δομή αυτή κάνει το Sn1per ιδιαίτερα ευέλικτο και εύκολο στην επέκταση των λειτουργιών του, καθώς για την ενσωμάτωση κάποιου επιπρόσθετου εργαλείου, αρκεί η προσθήκη μιας μόνο εντολής στο κατάλληλο αρχείο ροής εντολών.

Το Sn1per είναι ένα αρκετά απλό λογισμικό στη χρήση του. Για την εγκατάσταση του αρκεί η εκτέλεση του αντίστοιχου αρχείου εντολών, μέσω του οποίου θα εγκατασταθούν όλα τα εργαλεία τα οποία θα χρησιμοποιήσει το Sn1per για την εκτέλεση μιας διαδικασίας σάρωσης του προστατευόμενου υπολογιστικού συστήματος. Αντίστοιχα, για την απεγκατάσταση του, ο χρήστης αρκεί να εκτελέσει το αντίστοιχο αρχείο εντολών, ώστε να αφαιρεθούν τα εγκαταστημένα εργαλεία καθώς και όλα τα σχετικά βοηθητικά αρχεία που είχε δημιουργήσει το Sn1per κατά την εγκατάσταση και την εφαρμογή του.

Για την εκτέλεση μιας διαδικασίας σάρωσης αρκεί η σύνταξη μιας εντολής στη γραμμή εντολών με τις κατάλληλες παραμέτρους. Το Sn1per μπορεί να δεχτεί διάφορες παραμέτρους μέσω της γραμμής εντολών, καθώς και να παραμετροποιηθεί εκτενέστερα μέσω ειδικών αρχείων διαμόρφωσης, το οποία θα αναλυθούν στο υποκεφάλαιο 3.4. Η μόνη υποχρεωτική παράμετρος είναι η ηλεκτρονική διεύθυνση του προστατευόμενου υπολογιστικού συστήματος (-t). Οι υπόλοιπες παράμετροι είναι προαιρετικές και υποστηρίζουν την προσαρμογή της σάρωσης του προστατευόμενου υπολογιστικού συστήματος σύμφωνα με τις ανάγκες του χρήστη. Αυτές είναι η επιλογή της κατάστασης λειτουργίας (-m), που θα αναλυθεί στον υποκεφάλαιο 3.3, η επιλογή της θύρας δικτύου (-p), η επιλογή χώρου εργασίας (workspace, –w), που θα αναλυθεί στο τέλος του υποκεφαλαίου 3.2. Τέλος, δίνονται τέσσερις ακόμα ειδικές παράμετροι οι οποίες εκτελούν κάποιους επιπλέον ελέγχους. Αυτές είναι:

a) ο έλεγχος όλων των θυρών του προστατευόμενου υπολογιστικού συστήματος και όχι μόνο των πιο συνηθισμένων (full port scan, -fp),
b) ο εξαντλητικός έλεγχος για εύρεση διαπιστευτηρίων συνόδου (brute force, -b),
c) ο έλεγχος πληροφοριών ανοικτής πηγής (open source intelligence, OSINT, -o)
d) ο απλός έλεγχος για ενεργά συστήματα του προστατευόμενου υπολογιστικού συστήματος (recon, -re).

Μετά τη σύνθεση της επιθυμητής εντολής, το Sn1per εκκινεί τη διαδικασία σάρωσης του προστατευόμενου υπολογιστικού συστήματος. Κατά την εκτέλεση της σάρωσης εμφανίζονται στη γραμμή εντολών τα αποτελέσματα από τα διάφορα εργαλεία που εκτελεί το Sn1per. Ο τρόπος εμφάνισης είναι μέσω ενοτήτων, με την καθεμία να αποτελείται από έναν τίτλο, που υποδηλώνει την ενέργεια η οποία θα εκτελεστεί και από τα ευρήματα μετά την εκτέλεση της ενέργειας αυτής. Στην Εικόνα 3.1, φαίνεται ένα παράδειγμα έναρξης μιας διαδικασίας σάρωσης η οποία εκτελείται στο ίδιο υπολογιστικό σύστημα που έχει εγκατασταθεί το Sn1per (localhost). Στο παράδειγμα φαίνονται οι ενότητες όπου παρουσιάζονται τα αποτελέσματα από τέσσερεις ενέργειες σάρωσης. Αυτές οι ενέργειες είναι οι “gathering dns info”, “checking for subdomain hijacking”, “pinging host” και “running tcp port scan”. Στις δύο πρώτες ενότητες δεν καταγράφηκε κάποιο αποτέλεσμα από την εφαρμογή των εργαλείων του Sn1per, σε αντίθεση με τις επόμενες δύο, όπου εμφανίζεται κάποιο περιεχόμενο. Ειδικότερα, στην πρώτη ενότητα εμφανίζεται το αποτέλεσμα από την προσπάθεια για ανάκτηση πληροφοριών που αφορούν το προστατευόμενο υπολογιστικό σύστημα μέσω κάποιου προκαθορισμένου διακομιστή Domain Name System (DNS). Επειδή όμως στο συγκεκριμένο απλό παράδειγμα, το προστατευόμενο σύστημα είναι το ίδιο με αυτό που τρέχει το Sn1per, ο DNS διακομιστής δεν επιστρέφει κάποια απάντηση, οπότε και η ενέργεια “gathering dns info” δεν παράγει κάποιο αποτέλεσμα. Η δεύτερη ενότητα καταγράφει το αποτέλεσμα από την ανίχνευση της ύπαρξης τυχόν προσβάσιμων υποπεριοχών (subdomains), που όμως όπως φαίνεται στο συγκεκριμένο παράδειγμα, δεν υπάρχουν στο προστατευόμενο σύστημα. Στην τρίτη ενότητα εμφανίζεται το αποτέλεσμα από την προσπάθεια για επικοινωνία με το υπολογιστικό σύστημα ώστε να ελεγχθεί αν είναι ενεργό. Η τέταρτη και τελευταία ενέργεια που εκτελέστηκε στο συγκεκριμένο παράδειγμα, προσπαθεί να ανιχνεύσει τις όποιες ανοιχτές θύρες δικτύου του υπολογιστικού συστήματος. Καθώς το εν λόγω σύστημα διαθέτει μερικές ανοικτές θύρες δικτύου, το Sn1per τις εντοπίζει επιτυχώς και εμφανίζει τα ευρήματά του στην ενότητα αυτή. Οι τέσσερις αυτές ενέργειες που απεικονίζονται στις ενότητες που φαίνονται στην Εικόνα 3.1 είναι μόνο μερικές από τις δεκάδες ενέργειες που εκτελεί το Sn1per σε κάθε διαδικασία σάρωσης.



Εικόνα 3.1: Παράδειγμα εκτέλεσης σάρωσης του υπολογιστικού συστήματος πάνω στο οποίο εκτελείται το Sn1per (localhost).

3.2 Αποθήκευση Εξόδου

Κατά την ολοκλήρωση μιας σάρωσης, το Sn1per αποθηκεύει την έξοδο των αποτελεσμάτων κάθε εκτελέσιμου εργαλείου σε ξεχωριστά αρχεία. Η πλειοψηφία των αρχείων αυτών είναι αρχεία απλού κειμένου (plain text) και πολύ λιγότερα είναι σε γλώσσα σήμανσης XML (Extensible Mark-up Language). Η ονομασία κάθε αρχείου εξόδου είναι πολύ σημαντική. Αποτελείται από το θέμα του περιεχομένου του και την ηλεκτρονική διεύθυνση διαδικτυακού πρωτοκόλλου (Internet Protocol, IP, address) που το αφορά. Σε περίπτωση που η ηλεκτρονική διεύθυνση IP απουσιάζει, τότε πρόκειται για αρχείο με περιεχόμενα γενικής φύσεως. Έτσι για παράδειγμα, όταν ένα αρχείο περιέχει τα αποτελέσματα μιας σάρωσης εύρεσης ενεργών συστημάτων, τότε στην ονομασία του δεν περιλαμβάνεται κάποια ηλεκτρονική διεύθυνση IP. Αντίθετα, στην ονομασία ενός αρχείου που αφορά την ανίχνευση ανοικτών θυρών δικτύου ενός συστήματος, περιλαμβάνεται η ηλεκτρονική διεύθυνση IP του συστήματος που ελέγχθηκε. Τα αρχεία με τα αποτελέσματα της διαδικασίας σάρωσης, κατηγοριοποιούνται σε φακέλους που αντικατοπτρίζουν τα περιεχόμενά των περιεχόμενων αρχείων. Οι κατηγορίες των φακέλων και τα αρχεία που περιέχουν είναι τα εξής:

- **credentials**: περιέχει αρχεία όπου αποθηκεύονται τυχόν διαπιστευτήρια συνόδου δικτυακών υπηρεσιών που μπορεί να ανιχνεύτηκαν κατά τη διαδικασία σάρωσης.

- **domains:** περιέχει αρχεία με λίστες από όλες τις περιοχές (domains) που σαρώθηκαν από τη διαδικασία σάρωσης.
- **ips:** περιέχει αρχεία με λίστες από όλες τις ηλεκτρονικές διευθύνσεις IP των ενεργών υπολογιστικών συσκευών που ανιχνεύτηκαν στο προστατευόμενο υπολογιστικό σύστημα.
- **nmap:** περιέχει αρχεία με πληροφορίες παρόμοιες με τον φάκελο ips. Συγκεκριμένα, εδώ βρίσκονται τα αποτελέσματα της εκτέλεσης του εργαλείου nmap (ένα από τα εργαλεία που χρησιμοποιεί το Sn1per και ένα από τα διασημότερα εργαλεία ανίχνευσης δικτύου). Το Sn1per εκτελεί το nmap επαναλαμβανόμενα με διαφορετικές παραμέτρους με σκοπό την ανάκτηση διαφορετικών πληροφοριών για το προστατευόμενο υπολογιστικό σύστημα. Για παράδειγμα, να ανιχνευτούν οι ενεργές υπολογιστικές συσκευές, οι δικτυακές υπηρεσίες που εκτελούνται καθώς και οι τυχόν ανοιχτές θύρες δικτύου που βρέθηκαν σε κάθε μία από αυτές.
- **notes:** περιέχει αρχεία με πληροφορίες που αφορούν τον χειριστή. Το Sn1per τα αγνοεί, επιτρέποντας στον χειριστή να αποθηκεύει εκεί τυχόν σημειώσεις του.
- **osint:** περιέχει αρχεία με όλες τις πληροφορίες ανοικτής πηγής (open source intelligence OSINT). Αναλόγως τον τύπο της σάρωσης που έχει εκτελεστεί αυτός ο φάκελος μπορεί να είναι άδειος. Στη γενικότερη περίπτωση περιέχει αρχεία με λίστες από εκτεθειμένα email, τηλέφωνα και άλλες προσωπικές πληροφορίες που ανιχνεύτηκαν.
- **output:** περιέχει αρχεία με την έξοδο κάθε σάρωσης που έχει εκτελεστεί. Η ονομασία κάθε αρχείου περιέχει την ημερομηνία και την ώρα που εκτελέστηκε η αντίστοιχη σάρωση, επιτρέποντας έτσι τη διατήρηση ενός ιστορικού σαρώσεων.
- **reports:** όπως και στον φάκελο output, σε αυτόν περιέχονται αρχεία με την έξοδο από κάθε σάρωση που έχει εκτελεστεί, αλλά σε μορφή html για την προβολή σε φυλλομετρητή.
- **scans:** περιέχει αρχεία με τις εντολές και τις παραμέτρους των σαρώσεων που έχουν εκτελεστεί.
- **screenshots:** περιέχει αρχεία με στιγμιότυπα οθόνης από ιστοσελίδες που μπορεί να φιλοξενούνται στο προστατευόμενο υπολογιστικό σύστημα.
- **vulnerabilities:** περιέχει αρχεία αναφορών ευπαθειών που μπορεί να ανιχνεύτηκαν στο προστατευόμενο υπολογιστικό σύστημα. Οι αναφορές είναι χωρισμένες σε αρχεία ανάλογα με την κρισιμότητά τους καθώς και συνολικά όλες οι ευπάθειες σε ένα συγκεντρωτικό αρχείο.
- **web:** περιέχει αρχεία με πληροφορίες σχετικά με δικτυακές υπηρεσίες που υποστηρίζουν κάποια ιστοσελίδα. Εδώ βρίσκονται πληροφορίες σχετικά με την έκδοση του διακομιστή και τον τύπο του φιλοξενούμενου συστήματος, πιθανά σημεία για επίθεση μέσω δέσμης ενεργειών από άλλη τοποθεσία (Cross Site Scripting, XSS) καθώς και υποφάκελος με όλα τα αρχεία της γλώσσας Javascript που εντοπίστηκαν.

Όλη αυτή η δομή αρχείων με τη σειρά της αποθηκεύεται μέσα σε έναν κεντρικό φάκελο. Για να μπορεί να γίνει διαχωρισμός μεταξύ σαρώσεων με διαφορετικές παραμέτρους και προστατευόμενα υπολογιστικά συστήματα το Sn1per έχει, επιπλέον, την παράμετρο “χώρος εργασίας” (workspace). Η παράμετρος αυτή είναι προαιρετική και δέχεται ως όρισμα ένα όνομα. Το όνομα αυτό θα πάρει ο κεντρικός φάκελος μέσα στον οποίο βρίσκονται τα αποτελέσματα της σάρωσης. Σε περίπτωση που δεν έχει οριστεί η παράμετρος αυτή τότε τα αποτελέσματα της σάρωσης αποθηκεύονται σε μια προκαθορισμένη θέση (/etc/share/sniper/loot), με το μειονέκτημα ότι μπορεί εύκολα να μπερδευτούν τα αποτελέσματα διαφορετικών σαρώσεων μεταξύ τους. Αντιθέτως εάν έχει οριστεί η παράμετρος αυτή τότε τα αποτελέσματα της σάρωσης θα αποθηκευτούν στη θέση /etc/share/sniper/loot/workspace/ ${workspace_name}.

3.3 Τύποι λειτουργίας

Έχει ήδη αναφερθεί ότι κάποιες κατηγορίες φακέλων μπορεί να παραμείνουν κενές ενώ κάποιες άλλες μπορεί να περιέχουν ελάχιστες πληροφορίες. Αυτό συμβαίνει διότι το Sn1per μπορεί να εκτελεστεί με διαφορετικούς τύπους λειτουργίας. Αναλόγως του επιλεγμένου τύπου λειτουργίας τα αποτελέσματα που θα αποθηκευτούν στη δομή που αναλύθηκε προηγουμένως, ποικίλουν.

Παρακάτω θα αναλυθούν οι διάφοροι τύποι λειτουργίας του λογισμικού Sn1per και τα αποτελέσματα που παράγουν.

- **normal:** εκτελεί μια βασική σάρωση του προστατευόμενου υπολογιστικού συστήματος και των θυρών δικτύου του.
- **stealth:** όπως και η κατάσταση λειτουργίας **normal**, με τη διαφορά ότι προσπαθεί να δημιουργήσει όσο το δυνατό λιγότερο δικτυακό «θόρυβο» ώστε να μην εντοπιστεί και μπλοκαριστεί από πιθανά τείχη προστασίας (firewall) που προστατεύουν το υπολογιστικό σύστημα.
- **flyover:** εκτελεί μια γρήγορη σάρωση σε πολλαπλά υπολογιστικά συστήματα με σκοπό τον εντοπισμό των ενεργών.
- **discover:** εντοπίζει όλα τα υπολογιστικά συστήματα ενός δικτύου και εκτελεί μια γρήγορη σάρωση στο καθένα, εντοπίζοντας ενεργά συστήματα και ανοιχτές θύρες δικτύου.
- **port:** σαρώνει μόνο μία συγκεκριμένη θύρα δικτύου μίας υπολογιστικής συσκευής.
- **fullportonly:** εκτελεί μια λεπτομερή σάρωση όλων των θυρών δικτύου του προστατευόμενου υπολογιστικού συστήματος.
- **web:** σαρώνει τις ιστοσελίδες του προστατευόμενου υπολογιστικού συστήματος με περισσότερη λεπτομέρεια και αποθηκεύει πιο αναλυτικές αναφορές στον φάκελο web.
- **webporthttp:** εκτελεί την ίδια σάρωση όπως και το web, σαρώνοντας μόνο τις υπηρεσίες που λειτουργούν με το πρωτόκολλο HTTP του προστατευόμενου υπολογιστικού συστήματος.
- **webporthttps:** εκτελεί την ίδια σάρωση όπως και το web, σαρώνοντας μόνο τις υπηρεσίες που λειτουργούν με το πρωτόκολλο HTTPS του προστατευόμενου υπολογιστικού συστήματος.
- **vulnscan:** καταγράφει μόνο τις ευπάθειες του προστατευόμενου υπολογιστικού συστήματος.

3.4 Αρχεία διαμόρφωσης

Μια ακόμα δυνατότητα του Sn1per είναι η εκτενής παραμετροποίησή του μέσω των ειδικών αρχείων διαμόρφωσης (configuration files). Όπως περιγράφηκε στο υποκεφάλαιο 3.1, παρότι δίνεται η δυνατότητα παραμετροποίησης του τρόπου εκτέλεσης της σάρωσης του προστατευόμενου υπολογιστικού συστήματος, μέσω της γραμμής εντολών, είναι όμως αυτή περιορισμένη. Ωστόσο, μέσω των αρχείων διαμόρφωσης δίνεται μια καλύτερη ευελιξία στον χειριστή στο να ορίσει ποια εργαλεία θα εκτελεστούν κατά τη σάρωση και με ποιες παραμέτρους. Φυσικά, για την ορθή παραμετροποίηση απαιτείται μια τυπική οικειότητα με τα εργαλεία αυτά. Επιπλέον, προσφέρονται κάποια προκαθορισμένα αρχεία διαμόρφωσης, παραμετροποιημένα κατάλληλα για διάφορα πιθανά σενάρια σαρώσεων, όπως για παράδειγμα μία γρήγορη σάρωση με περιορισμένα αποτελέσματα ή κάποια σάρωση η οποία να δημιουργεί την ελάχιστη δυνατή κίνηση στο δίκτυο ώστε να μην εντοπίζεται το Sn1per από πιθανά τείχη προστασίας. Το κατάλληλο αρχείο διαμόρφωσης σε συνδυασμό με την κατάλληλη κατάσταση λειτουργίας μπορεί να επιφέρει πολύ χρήσιμα αποτελέσματα.

Στην Εικόνα 3.2 φαίνεται ένα απόσπασμα από το προκαθορισμένο αρχείο διαμόρφωσης, δηλαδή το αρχείο “default”, που χρησιμοποιείται σε περίπτωση που ο χειριστής δεν έχει ορίσει κάποια συγκεκριμένη διαμόρφωση. Τα αρχεία διαμόρφωσης στην ουσία είναι μια λίστα από

μεταβλητές τις οποίες διαβάζει το Sn1per κατά την εκτέλεση κάποιας σάρωσης και ανάλογα με τις τιμές τους εκτελεί τα αντίστοιχα εργαλεία με τις αντίστοιχες παραμέτρους. Οι γραμμές που ξεκινούν με το σύμβολο της δίεσης (#) είναι σχόλια και αγνοούνται από το Sn1per. Στο απόσπασμα που παρουσιάζεται στην Εικόνα 3.2, φαίνονται οι παράμετροι για τέσσερα διαφορετικά εργαλεία, το burp, openvas, nessus και metasploit. Στην περίπτωση των openvas και nessus φαίνονται οι ομώνυμες μεταβλητές να έχουν την τιμή 0. Αυτό υποδεικνύει ότι κατά την σάρωση δεν θα χρησιμοποιηθούν τα δύο αυτά εργαλεία. Τα τέσσερα εργαλεία του αποσπάσματος έχουν τον ίδιο τρόπο λειτουργίας, χρησιμοποιούν, δηλαδή, κάποιον διακομιστή και μέσω αυτού γίνεται η επικοινωνία με το ίδιο το εργαλείο. Έτσι και στα τέσσερα εργαλεία υπάρχουν μεταβλητές που αφορούν την ηλεκτρονική διεύθυνση IP του διακομιστή (127.0.0.1, δηλαδή το ίδιο το υπολογιστικό σύστημα στο οποίο εκτελείται το Sn1per) και την αντίστοιχη θύρα δικτύου την οποία χρησιμοποιεί το καθένα. Επιπλέον, στην περίπτωση των openvas και nessus υπάρχουν μεταβλητές με διαπιστευτήρια συνόδου (credentials) τα οποία είναι απαραίτητα για την επιτυχή επικοινωνία του Sn1per με τον διακομιστή του εκάστοτε εργαλείου.

```
# BURP 2.0 SCANNER CONFIG
BURP_HOST="127.0.0.1"
BURP_PORT="1338"

# OPENVAS CONFIG
OPENVAS="0"
OPENVAS_HOST="127.0.0.1"
OPENVAS_PORT="9390"
OPENVAS_USERNAME="admin"
OPENVAS_PASSWORD=""
OPENVAS_RUNAS_USER="kali"

# NESSUS CONFIG
NESSUS="0"
NESSUS_HOST="127.0.0.1:8834"
NESSUS_USERNAME="admin"
NESSUS_PASSWORD=""
NESSUS_POLICY_ID="c3cbcd46-329f-a9ed-1077-554f8c2af33d0d44f09d736969bf"

# METASPLOIT SCANNER CONFIG
METASPLOIT_IMPORT="0"
MSF_LHOST="127.0.0.1"
MSF_LPORT="4444"
```

Εικόνα 3.2: Παράδειγμα αποσπάσματος αρχείου διαμόρφωσης

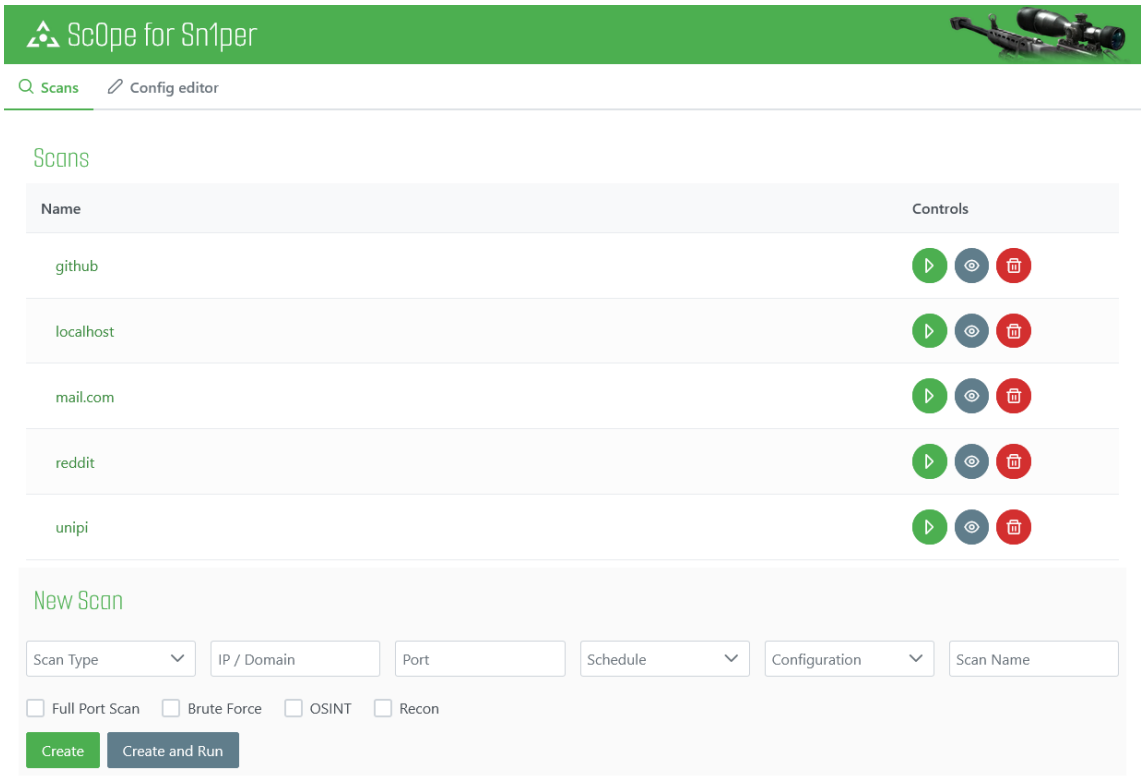
ΚΕΦΑΛΑΙΟ 4: ΓΡΑΦΙΚΟ ΠΕΡΙΒΑΛΛΟΝ

Το γραφικό περιβάλλον ενός λογισμικού είναι ένα σύνολο από γραφικά στοιχεία που έχει σχεδιαστεί για να αναπαραστήσει με όμορφο, εύχρηστο και λειτουργικό τρόπο το περιβάλλον εργασίας με το οποίο αλληλοεπιδρά ο χρήστης με την υπολογιστική συσκευή. Η ελεύθερη έκδοση, ανοικτού κώδικα του Sn1per, μπορεί να εκτελεστεί μόνο σε περιβάλλον γραμμής εντολών, όπως περιγράφηκε στο προηγούμενο κεφάλαιο. Από τον χειριστή του απαιτείται καλή γνώση των εντολών και των παραμέτρων τους καθώς και των κανόνων σύνταξης των εντολών αυτών, ώστε να εκτελεστεί η κατάλληλη διαδικασία σάρωσης. Η απαίτηση αυτή κάνει το περιβάλλον γραμμής εντολών αφιλόξενο για τους περισσότερους χρήστες. Όμως, μέσω ενός γραφικού περιβάλλοντος, ο χειριστής του Sn1per δεν χρειάζεται να γνωρίζει τις εντολές και τις παραμέτρους τους, καθώς η χρήση του λογισμικού γίνεται μέσω διαισθητικών γραφικών στοιχείων, όπως κουμπιών, αναπτυσσόμενων λιστών και αναδυόμενων καρτελών φυλλομετρητή.

Η επαγγελματική, κλειστού κώδικα, έκδοση του Sn1per προσφέρεται με τη δυνατότητα πλοήγησης μέσω γραφικού περιβάλλοντος στον φυλλομετρητή, αποκρύπτοντας από τον χρήστη το αφιλόξενο περιβάλλον γραμμής εντολών. Στην παρούσα εργασία αναπτύχθηκε στην ελεύθερη έκδοση του Sn1per, ανοικτού κώδικα, ένα φιλικό και εύχρηστο γραφικό περιβάλλον, εμπνευσμένο από αυτό που παρέχει η έκδοση κλειστού κώδικα, προκειμένου να διευκολύνει τον οποιαδήποτε χρήστη στην εφαρμογή του.

4.1 Κεντρική Σελίδα του Γραφικού Περιβάλλοντος

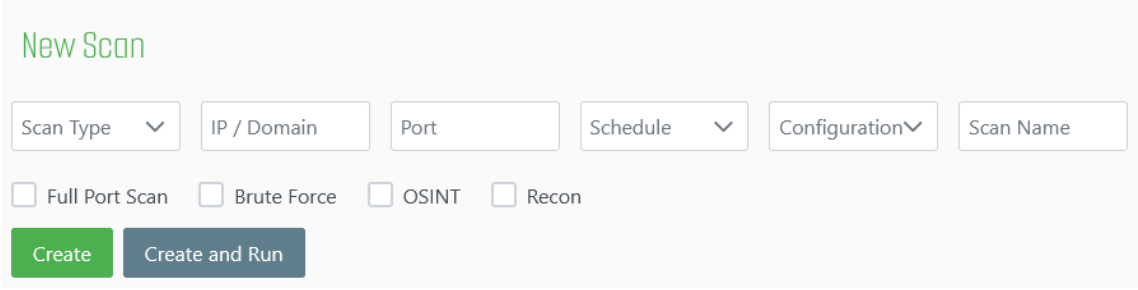
Η κεντρική σελίδα του γραφικού περιβάλλοντος παρουσιάζεται στην Εικόνα 4.1.



Εικόνα 4.1: Παράδειγμα κεντρικής σελίδας του γραφικού περιβάλλοντος

Στο κάτω μέρος της κεντρικής σελίδας βρίσκεται μια φόρμα προς συμπλήρωση, όπως φαίνεται στην Εικόνα 4.2. Με τη φόρμα αυτή ο χειριστής μπορεί εύκολα να δημιουργήσει μια εντολή

εκτέλεσης σάρωσης του υπολογιστικού συστήματος που επιθυμεί να προστατέψει μέσω του Sn1per.



Εικόνα 4.2: Γραφικό περιβάλλον δημιουργίας νέας εντολής σάρωσης.

Στην Εικόνα 4.2 διακρίνεται το πεδίο “Scan Name” (φιλική ονομασία), η οποία χρησιμοποιείται για τη δημιουργία ενός χώρου εργασίας (workspace) για την αποθήκευση των αποτελεσμάτων της σάρωσης, όπως έχει περιγραφεί στο υποκεφάλαιο 3.2, καθώς και το πεδίο της ηλεκτρονικής διεύθυνσης του διαδικτυακού πρωτοκόλλου IP ή του ονόματος περιοχής (IP/domain) του προστατευόμενου υπολογιστικού συστήματος. Τα δύο αυτά πεδία είναι υποχρεωτικά και αν δεν συμπληρωθούν δεν δύναται να συνεχίσει η διαδικασία. Τα υπόλοιπα πεδία είναι προαιρετικά. Αυτά είναι:

1) ο τύπος λειτουργίας της διαδικασίας σάρωσης (scan type), δίνοντας τη δυνατότητα επιλογής, μέσω αναπτυσσόμενης λίστας, για έναν από τους δέκα τύπους που αναφέρθηκαν στο υποκεφάλαιο 3.2,
2) η θύρα δικτύου (port) του προστατευόμενου υπολογιστικού συστήματος που επιθυμεί ο χειριστής να ελεγχθεί,
3) ο προγραμματισμός επανάληψης (schedule) της διαδικασίας σάρωσης, επιλέγοντας αν θα είναι καθημερινά, εβδομαδιαία ή μηνιαία μέσω αναπτυσσόμενης λίστας και
4) το αρχείο διαμόρφωσης (configuration) που θα χρησιμοποιηθεί κατά τη διαδικασία σάρωσης, σύμφωνα με τα αναφερόμενα στο υποκεφάλαιο 3.4.

Τέλος, δίνονται επιπλέον τέσσερις δυνατότητες επιλογής που αφορούν την εκτέλεση επιπρόσθετων ελέγχων. Αυτές είναι:

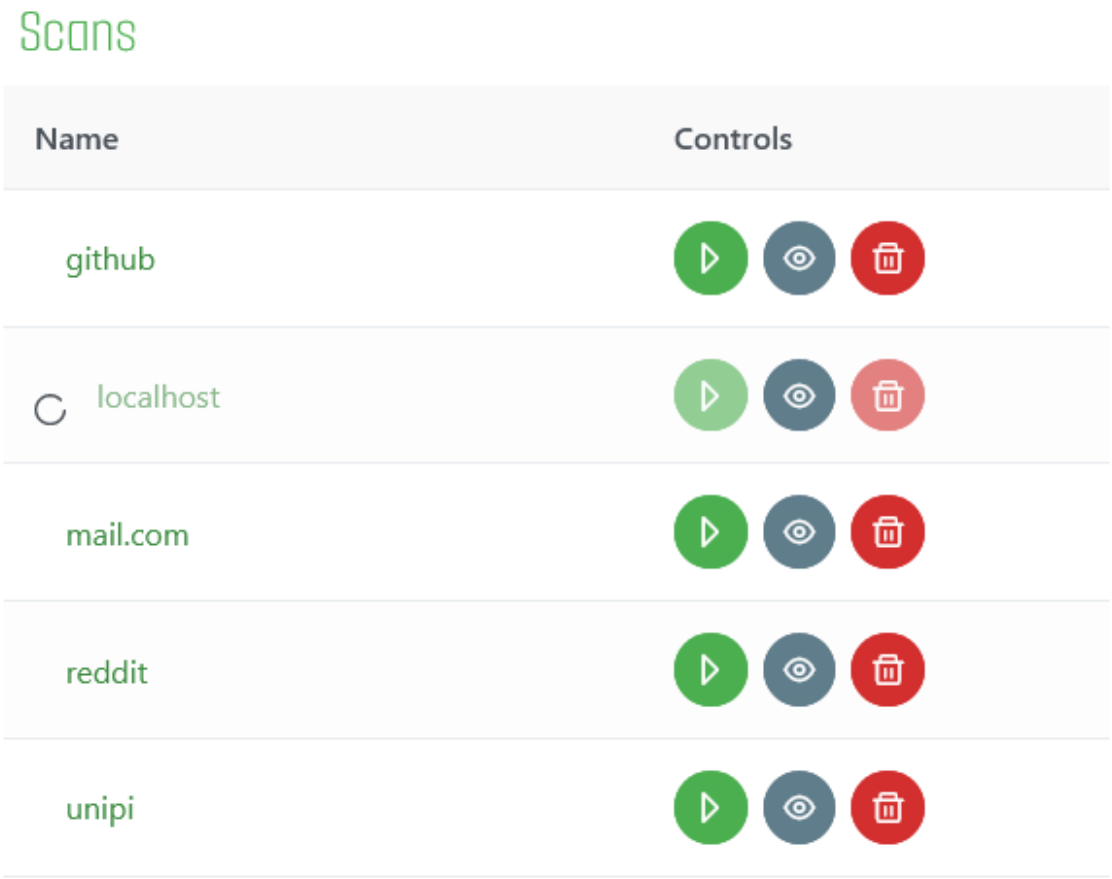
1) ο έλεγχος όλων των θυρών του προστατευόμενου υπολογιστικού συστήματος και όχι μόνο των πιο συνηθισμένων (full port scan),
2) ο εξαντλητικός έλεγχος για εύρεση διαπιστευτηρίων συνόδου (brute force),
3) ο έλεγχος πληροφοριών ανοικτής πηγής (open source intelligence, OSINT) και
4) ο απλός έλεγχος για ενεργά συστήματα του προστατευόμενου υπολογιστικού συστήματος (recon).

Καθένα από τα πεδία αυτά αντιστοιχεί σε μια ξεχωριστή παράμετρο της εντολής εκτέλεσης του Sn1per, όπως έχουν περιγραφεί στο Κεφάλαιο 3.

Αφού συμπληρωθούν κατάλληλα όλα τα απαιτούμενα πεδία της φόρμας, στη συνέχεια ο χειριστής έχει τη δυνατότητα να επιλέξει ανάμεσα σε δύο κουμπιά επιλογής, το “Create” (Δημιουργία) ή το “Create and Run” (Δημιουργία και Εκτέλεση). Στην περίπτωση της επιλογής του κουμπιού “Create”, το γραφικό περιβάλλον θα δημιουργήσει έναν φάκελο με τη φιλική ονομασία που όρισε ο χειριστής στο πεδίο Scan Name, ο οποίος θα παίξει τον ρόλο του φακέλου του χώρου εργασίας (workspace) που θα χρησιμοποιήσει το Sn1per για να αποθηκεύει τα αποτελέσματα της διαδικασίας σάρωσης. Στον φάκελο αυτό αποθηκεύεται και η ίδια η εντολή με τις παραμέτρους που δημιουργήθηκε μέσω της φόρμας, για μελλοντική εκτέλεση. Στην

περίπτωση που ο χειριστής έχει επιλέξει το κουμπί “Create and Run”, τότε αμέσως μετά την προαναφερόμενη διαδικασία αποθήκευσης, ξεκινάει η εκτέλεση της διαδικασίας σάρωσης.

Πάνω από τη φόρμα δημιουργίας μιας νέας εντολής, στην κεντρική σελίδα βρίσκεται η λίστα με όλες τις προηγούμενες εντολές που έχουν ήδη δημιουργηθεί και αποθηκευτεί, όπως αυτή φαίνεται καλύτερα στην Εικόνα 4.3.



Εικόνα 4.3: Λίστα αποθηκευμένων εντολών στο γραφικό περιβάλλον

Η λίστα των αποθηκευμένων εντολών χωρίζεται σε δύο μέρη. Αριστερά, φαίνονται οι φιλικές ονομασίες που έχουν δοθεί στις αποθηκευμένες εντολές και δεξιά βρίσκονται τα κουμπιά ελέγχου κάθε εντολής. Η κάθε εντολή μπορεί να βρίσκεται σε δύο καταστάσεις, είτε ενεργή είτε ανενεργή.

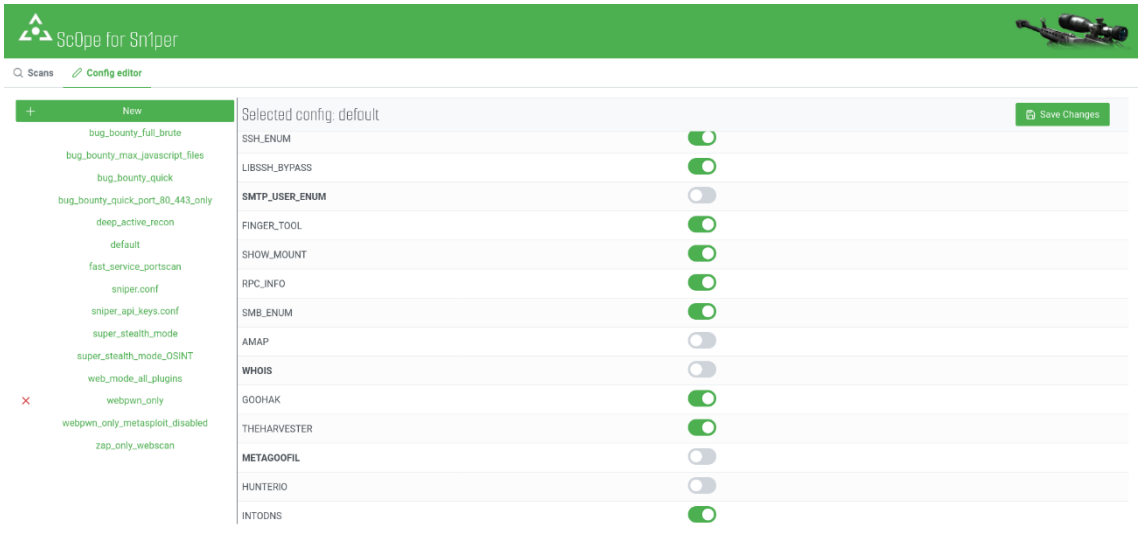
Όταν μια εντολή είναι ενεργή σημαίνει ότι εκείνη τη στιγμή εκτελείται η διαδικασία σάρωσης του προστατευόμενου υπολογιστικού συστήματος που ορίστηκε κατά τη δημιουργία της. Το Sn1per έχει αρχίσει, δηλαδή, να σαρώνει το προστατευόμενο σύστημα και να παράγει αποτελέσματα. Όταν μια εντολή βρίσκεται σε αυτή την κατάσταση εμφανίζεται αριστερά της φιλικής της ονομασίας ένα περιστρεφόμενο κυκλικό σύμβολο. Όπως είχε αναλυθεί στο Κεφάλαιο 3, το Sn1per κατά την εκτέλεσή του εμφανίζει τα ευρήματά του στο περιβάλλον γραμμής εντολών. Οι πληροφορίες αυτές είναι χρήσιμες για τον χειριστή καθώς ενημερώνεται για τον συγκεκριμένο έλεγχο που εκτελείται κάθε στιγμή της σάρωσης. Για να μην χαθεί η πληροφορία αυτή, στο γραφικό περιβάλλον έχει δημιουργηθεί ένα ειδικό περιβάλλον προβολής της γραμμής εντολών και των πληροφοριών που εμφανίζονται σε αυτή. Πατώντας το μεσαίο γκρι κουμπί (μάτι), ανοίγει

σε μια νέα καρτέλα του φυλλομετρητή το περιβάλλον αυτό που θα αναλυθεί με λεπτομέρεια στο υποκεφάλαιο 4.4.1.

Όταν μια εντολή είναι ανενεργή, τότε δίνονται περισσότερες δυνατότητες στον χειριστή. Όπως και στην προηγούμενη κατάσταση, δύναται να προβληθεί η έξοδος της γραμμής εντολών της τελευταίας σάρωσης. Επιπλέον, δίνεται η δυνατότητα μετάβασης της σάρωσης σε ενεργή κατάσταση με το πάτημα του αριστερού πράσινου κουμπιού (τρίγωνο). Το δεξιό κόκκινο κουμπί (κάδος απορριμμάτων) διαγράφει την εντολή και τον αντίστοιχο χώρο εργασίας (workspace) καθώς και τα αποτελέσματά των σαρώσεων. Τέλος, πατώντας πάνω στη φιλική ονομασία της εντολής εμφανίζεται η σελίδα προβολής αποτελεσμάτων, που θα αναλυθεί με λεπτομέρεια στο υποκεφάλαιο 4.3.

4.2 Σελίδα επεξεργασίας των αρχείων διαμόρφωσης

Σε αυτή τη σελίδα ο χρήστης μπορεί να δημιουργήσει νέα, να επεξεργαστεί ήδη υπάρχοντα και να διαγράψει αρχεία διαμόρφωσης (configuration files), όπως αυτά έχουν περιγραφεί στο υποκεφάλαιο 3.4. Στην Εικόνα 4.4 φαίνεται ένα απόσπασμα οθόνης από τη σελίδα αυτή.



Εικόνα 4.4: Οθόνη επεξεργασίας αρχείων διαμόρφωσης

Στα αριστερά της οθόνης φαίνεται μια λίστα με όλα τα διαθέσιμα αρχεία διαμόρφωσης και αυτά που έχουν δημιουργηθεί από τον χειριστή καθώς και τα έτοιμα, προκαθορισμένα, αρχεία που προσφέρει το ίδιο το Sn1per. Στο πάνω μέρος της λίστας αυτής υπάρχει ένα κουμπί “New” (Νέο) με το οποίο μπορεί να δημιουργηθεί ένα νέο αρχείο. Όταν αυτό πατηθεί, θα ζητηθεί από τον χειριστή να ονομάσει το νέο αρχείο, μέσω ενός αναδυόμενου παραθύρου και δημιουργείται ένα νέο αρχείο με περιεχόμενα ίδια με αυτά του αρχείου διαμόρφωσης “default”. Το αρχείο “default” προτείνεται από τους δημιουργούς του Sn1per για διαδικασίες σάρωσης γενικής φύσεως. Επιπρόσθετα, μέσω της λίστας αυτής είναι δυνατή η διαγραφή κάποιου αρχείου διαμόρφωσης. Συγκεκριμένα, περνώντας τον κέρσορα πάνω από κάποιο όνομα αρχείου στη λίστα, τότε στα αριστερά του εμφανίζεται το κουμπί διαγραφής με ένα κόκκινο σύμβολο “X”. Στο παράδειγμα της Εικόνας 4.4, το κουμπί διαγραφής “X” έχει εμφανιστεί δίπλα στο αρχείο διαμόρφωσης “webpwn_only”.

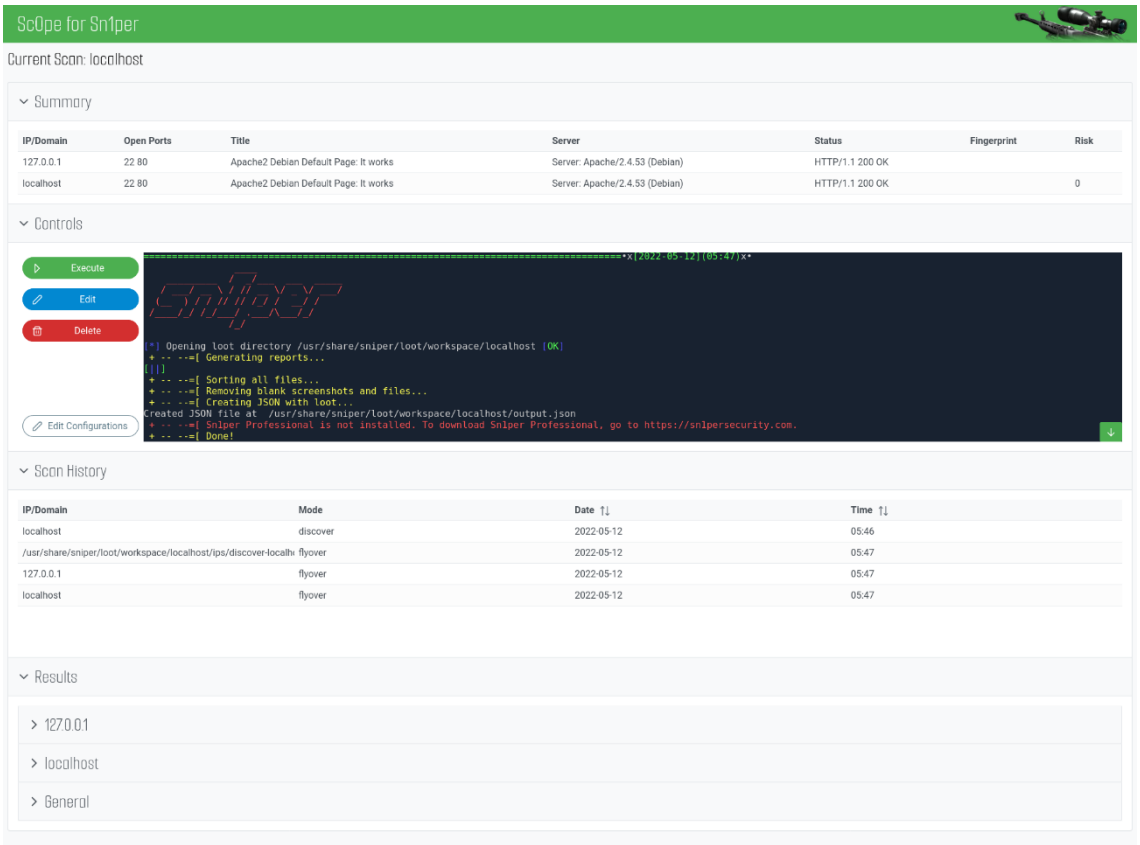
Επιλέγοντας κάποιο αρχείο διαμόρφωσης από τη λίστα, στο κεντρικό μέρος της οθόνης εμφανίζεται ένας πίνακας με όλες τις ρυθμίσεις του αρχείου, τις οποίες μπορεί να τροποποιήσει ο χειριστής. Εάν πραγματοποιηθεί κάποια τροποποίηση, τότε στο πάνω δεξιά μέρος της οθόνης

εμφανίζεται ένα πράσινο κουμπί αποθήκευσης “Save changes” (αποθήκευση αλλαγών), προκειμένου να αποθηκευτεί το αρχείο διαμόρφωσης στη νέα μορφή του. Οι τροποποιήσεις επισημαίνονται με έντονα (bold) γράμματα.

4.3 Σελίδα προβολής αποτελεσμάτων

Τα αποτελέσματα από μία διαδικασία σάρωσης παρουσιάζονται στη σελίδα προβολής αποτελεσμάτων. Η σελίδα αποτελείται από τέσσερα αναπτυσσόμενα τμήματα, “Summary” (Σύνοψη), “Controls” (Έλεγχος), “Scan History” (Ιστορικό διαδικασιών σάρωσης) και “Results” (Αποτελέσματα).

Στην Εικόνα 4.5 φαίνεται η σελίδα προβολής αποτελεσμάτων από μία δοκιμαστική εφαρμογή του Sn1per. Για πρακτικούς λόγους, επιλέχθηκε η σάρωση να βασιστεί στον τύπο λειτουργίας “discover”, ο οποίος εκτελεί πολύ βασικές ενέργειες σάρωσης. Ωστόσο, υπενθυμίζεται ότι το Sn1per, μέσω των εργαλείων που χρησιμοποιεί, έχει τη δυνατότητα να αντλεί έναν μεγάλο όγκο πληροφοριών για τα υπολογιστικά συστήματα που καλείται να προστατεύσει. Στο συγκεκριμένο παράδειγμα, το προστατευόμενο υπολογιστικό σύστημα ήταν το ίδιο με αυτό που είναι εγκαταστημένο το Sn1per, οπότε έτσι περιορίζεται ο όγκος των πληροφοριών που αντλούνται.



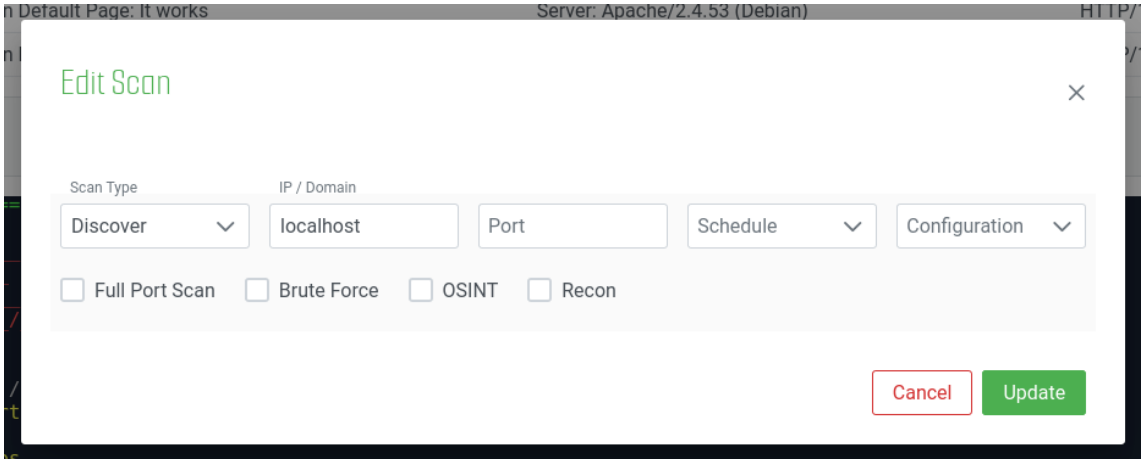
Εικόνα 4.5 Σελίδα προβολής αποτελεσμάτων μετά από τη σάρωση του ίδιου του υπολογιστικού συστήματος στο οποίο είναι εγκαταστημένο το Sn1per

Το πρώτο τμήμα, “Summary”, παρουσιάζει μια σύνοψη των πληροφοριών που μπόρεσαν να αποσπαστούν από το υπολογιστικό σύστημα που σαρώθηκε. Συγκεκριμένα, παρουσιάζεται ένας πίνακας με όλες τις ενεργές υπολογιστικές συσκευές που εντοπίστηκαν στο σύστημα και για

την καθεμία παρουσιάζονται ορισμένες πληροφορίες. Οι πληροφορίες αυτές είναι η ηλεκτρονική διεύθυνση IP της συσκευής, οι ανοιχτές θύρες δικτύου που βρέθηκαν, ο τίτλος και η κατάσταση της ιστοσελίδας που φιλοξενείται στη συσκευή, εάν υπάρχει, η έκδοση του διακομιστή, η έκδοση του λειτουργικού συστήματος και το συνολικό ρίσκο στο οποίο είναι εκτεθειμένη η υπολογιστική συσκευή σύμφωνα με τις ευπάθειες που έχουν εντοπιστεί. Ο πίνακας αυτός περιέχει όσες πληροφορίες μπόρεσαν να ανακτηθούν μετά την ολοκλήρωση της διαδικασίας σάρωσης, οπότε είναι πιθανόν να υπάρχουν κάποια κενά κελιά.

Το δεύτερο τμήμα, “Controls”, περιέχει κουμπιά ελέγχου για την αποθηκευμένη εντολή σάρωσης, παρόμοια με αυτά που βρίσκονται και στη λίστα αποθηκευμένων εντολών της κεντρικής σελίδας, καθώς και την έξοδο των αποτελεσμάτων της διαδικασίας σάρωσης, όπως αυτή παρουσιάζεται στη γραμμή εντολών. Τα κουμπιά ελέγχου απαρτίζονται από το κουμπί εκκίνησης, “Execute” (Εκτέλεση) και το κουμπί διαγραφής “Delete” (Διαγραφή) της αποθηκευμένης εντολής που εκτελούν την ίδια λειτουργία με αυτά που υπάρχουν και στη λίστα αποθηκευμένων εντολών. Επίσης, υπάρχει και ένα κουμπί “Edit” (Επεξεργασία) που επιτρέπει την τροποποίηση των παραμέτρων της αποθηκευμένης εντολής σάρωσης. Όταν πατηθεί το κουμπί “Edit”, τότε η επεξεργασία των παραμέτρων της εντολής γίνεται μέσω ενός αναδυόμενου παράθυρου, το οποίο παρέχει την ίδια φόρμα που βρίσκεται στην κεντρική σελίδα για τη δημιουργία νέας εντολής σάρωσης. Η μόνη διαφορά είναι ότι σε αυτή τη φόρμα δεν δίνεται η δυνατότητα επεξεργασίας της φιλικής ονομασίας. Η φόρμα αυτή παρουσιάζεται στην Εικόνα 4.6.

Τέλος, σε αυτό το τμήμα υπάρχει και ένα κουμπί “Edit Configurations” (Επεξεργασία διαμορφώσεων) για την τροποποίηση των αρχείων διαμόρφωσης, το πάτημα του οποίου μεταφέρει τον χειριστή στη σελίδα επεξεργασίας αρχείων διαμόρφωσης, όπως έχει περιγραφεί στο υποκεφάλαιο 4.2.

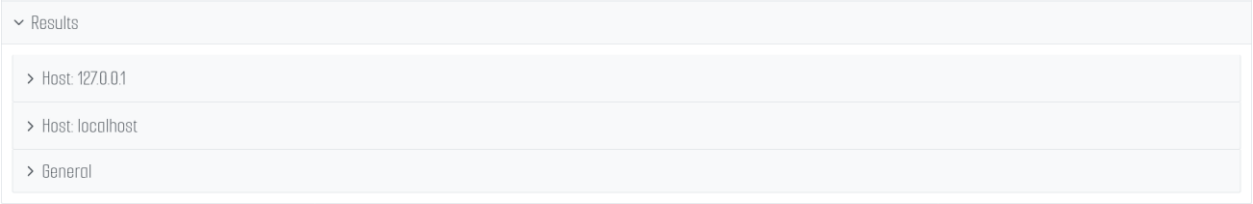


Εικόνα 4.6: Φόρμα επεξεργασίας αποθηκευμένης εντολής σάρωσης

Το τρίτο τμήμα, “Scan History”, περιέχει έναν πίνακα με το ιστορικό προηγούμενων εκτελέσεων της επιλεγμένης εντολής σάρωσης. Ο πίνακας αυτός παρέχει πληροφορίες σχετικά με τις ηλεκτρονικές διευθύνσεις που έχουν σαρωθεί, τον τύπο της σάρωσης, την ημερομηνία και την ώρα εκτέλεσης.

Το τέταρτο και τελευταίο τμήμα, “Results”, παρουσιάζει τα αποτελέσματα της διαδικασίας σάρωσης. Όπως έχει περιγραφεί στο υποκεφάλαιο 3.2, το Sn1per αποθηκεύει τα αποτελέσματα της σάρωσης σε διαφορετικά αρχεία, κατηγοριοποιημένα σε φακέλους, ανάλογα με το περιεχόμενό τους. Το γραφικό περιβάλλον αναπτύχθηκε υιοθετώντας αυτήν τη δομή αποθήκευσης, ώστε να διευκολύνει την περιήγηση στους φακέλους και στα περιεχόμενα αρχεία τους. Αρχικά εκτελεί μια κατηγοριοποίηση με βάση την ηλεκτρονική διεύθυνση IP. Όπως αναφέρθηκε στο υποκεφάλαιο 3.2, η ονοματολογία των αρχείων γίνεται με βάση την πληροφορία

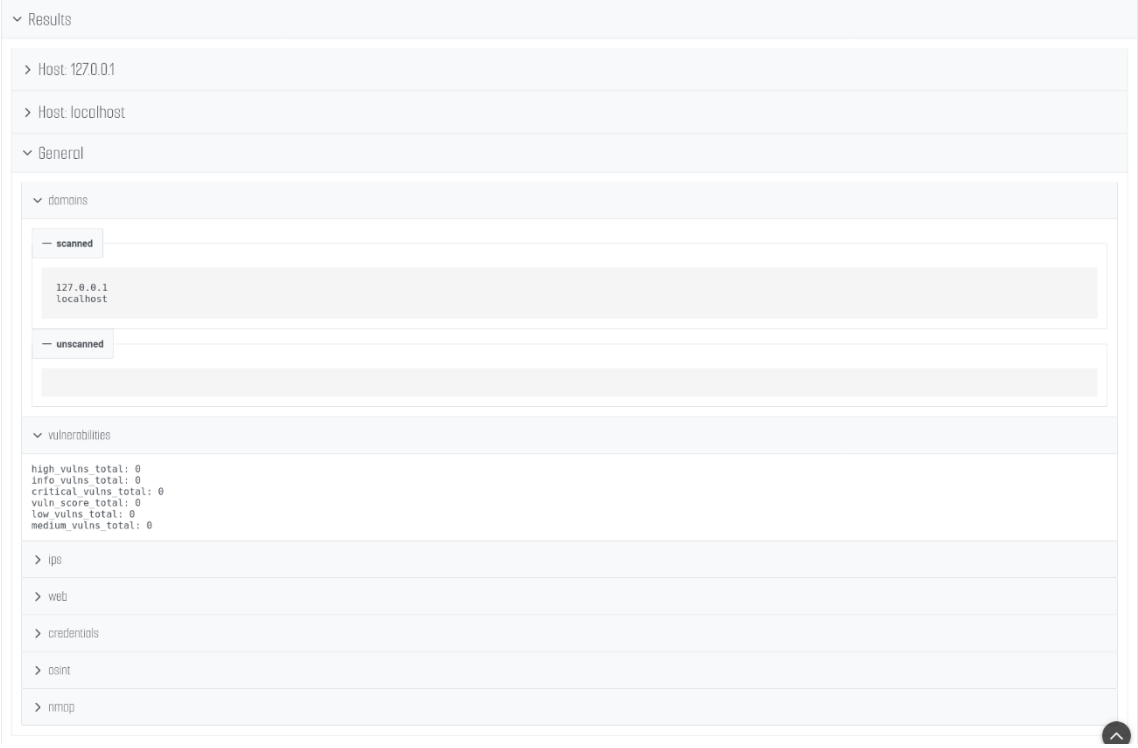
που φέρουν και την ηλεκτρονική διεύθυνση της υπολογιστικής συσκευής την οποία αφορούν. Αναγνωρίζοντας από την ονομασία του κάθε αρχείου, το γραφικό περιβάλλον κατηγοριοποιεί τα αποτελέσματα σύμφωνα με την ηλεκτρονική διεύθυνση IP. Δημιουργεί τόσα αναπτυσσόμενα τμήματα, όσα και οι διαφορετικές ηλεκτρονικές διευθύνσεις. Επιπλέον, δημιουργεί και ένα ακόμα αναπτυσσόμενο τμήμα “General” (Γενικό τμήμα) στο οποίο βρίσκονται γενικές πληροφορίες σχετικά με τη διαδικασία σάρωσης. Στην Εικόνα 4.7 φαίνεται συνοπτικά η κατηγοριοποίηση που περιγράφηκε.



Εικόνα 4.7: Παράδειγμα κατηγοριοποίησης των αποτελεσμάτων της διαδικασίας σάρωσης.

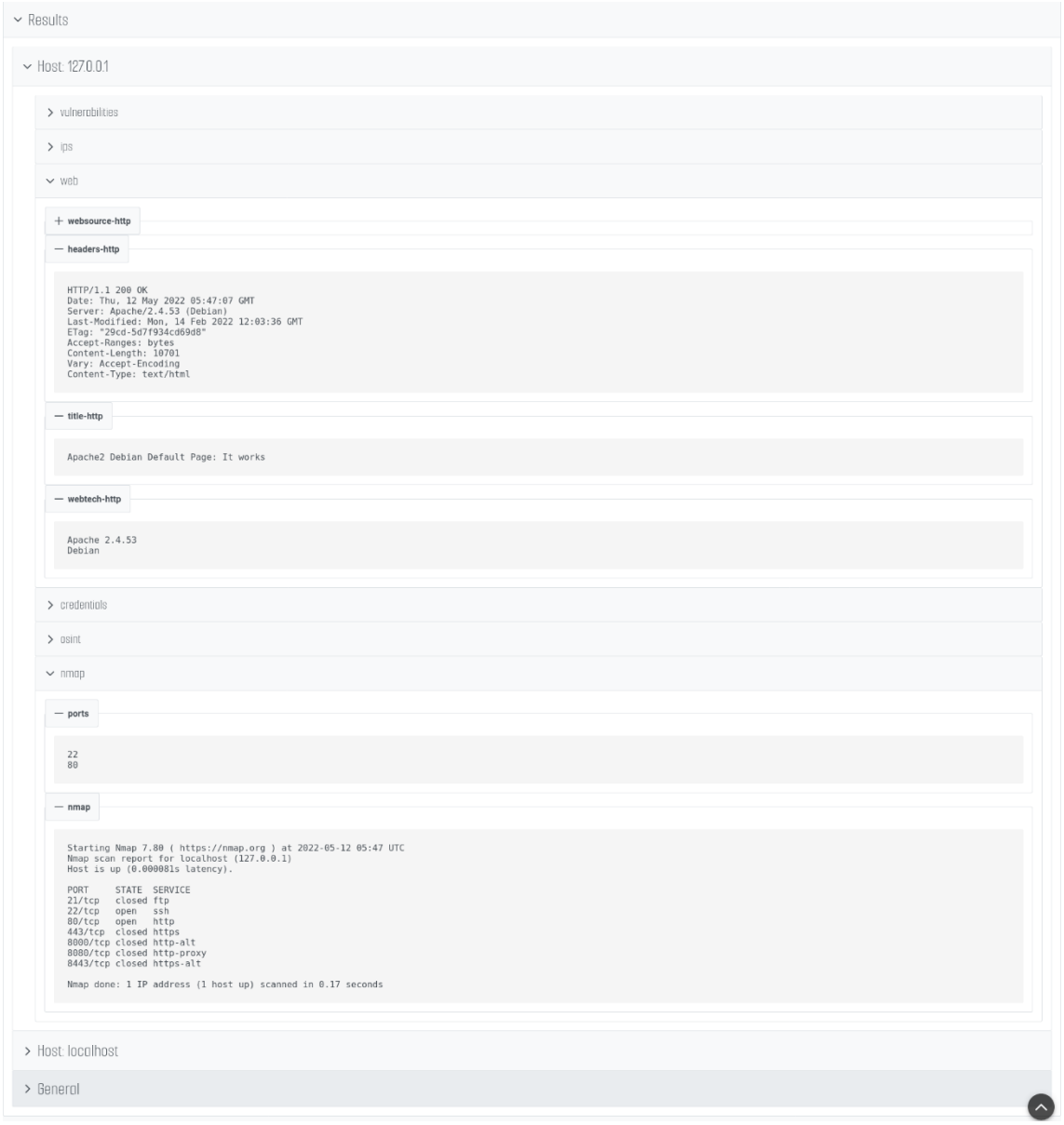
Στα περιεχόμενα κάθε κατηγορίας βρίσκονται δώδεκα αναπτυσσόμενα τμήματα όσοι και οι φάκελοι κατηγορίας των αποτελεσμάτων που δημιουργεί το Sn1per, όπως έχουν περιγραφεί στο υποκεφάλαιο 3.1. Μέσα στην κάθε κατηγορία βρίσκονται τόσα αναπτυσσόμενα πεδία όσα και τα αρχεία που περιέχονται στον κάθε φάκελο κατηγορία. Πιθανόν, κάποιες κατηγορίες μπορεί να είναι κενές. Για παράδειγμα, είναι πιθανόν η κατηγορία credentials να είναι κενή, καθώς μπορεί να μην ήταν εφικτή η ανάκτηση διαπιστευτηρίων συνόδου των δικτυακών υπηρεσιών που εντοπίστηκαν να εκτελούνται στο προστατευόμενο υπολογιστικό σύστημα.

Συνεχίζοντας την περιγραφή της σελίδας προβολής αποτελεσμάτων της δοκιμαστικής σάρωσης, επεκτείνοντας (βλ. Εικόνα 4.7) το αναπτυσσόμενο τμήμα “General” εμφανίζονται οι γενικές πληροφορίες που αποθηκεύτηκαν για τη δοκιμαστική διαδικασία σάρωσης. Στην Εικόνα 4.8 φαίνονται οι πληροφορίες που περιέχονται στον κάθε φάκελο κατηγορίας. Για παράδειγμα, στον φάκελο κατηγορία “domains” βρίσκονται δύο πεδία, το ένα με όνομα “scanned” και το άλλο “unscanned”. Αυτά τα δύο πεδία αφορούν τα δυο αρχεία που βρίσκονται στη θέση domains/scanned.txt και domains/unscanned.txt, αντίστοιχα. Οι πληροφορίες που περιέχονται σε αυτά τα δύο αρχεία αφορούν, αντίστοιχα, τις ηλεκτρονικές διευθύνσεις IP για τις οποίες ολοκληρώθηκε επιτυχώς η διαδικασία σάρωσης και αυτές που για κάποιον λόγο η διαδικασία διακόπηκε. Στην προκειμένη περίπτωση έχουν σαρωθεί οι στόχοι localhost και 127.0.0.1, δηλαδή το ίδιο το υπολογιστικό σύστημα στο οποίο είναι εγκαταστημένο το Sn1per.



Εικόνα 4.8: Παράδειγμα σελίδας παρουσίασης γενικών πληροφοριών από τη δοκιμαστική σάρωση.

Συνεχίζοντας στο ίδιο παράδειγμα, επεκτείνοντας τα περιεχόμενα μιας από τις ενεργές ηλεκτρονικές διευθύνσεις που εντοπίστηκαν, φαίνονται οι σχετικές πληροφορίες που αντλήθηκαν και την αφορούν. Στην Εικόνα 4.9 φαίνονται οι πληροφορίες που έχουν αντληθεί σχετικά με το υπολογιστικό σύστημα της ηλεκτρονικής διεύθυνσης 127.0.0.1, δηλαδή το ίδιο υπολογιστικό σύστημα στο οποίο είναι εγκαταστημένο το Sn1per. Στον φάκελο κατηγορία “web” βρίσκεται ένα πεδίο με όνομα “webtech-http”, το οποίο αφορά το αντίστοιχο αρχείο που είναι αποθηκευμένο στη θέση web/webtech-http.txt. Η πληροφορία που περιέχει το αρχείο αυτό είναι η έκδοση του διακομιστή που χρησιμοποιεί το σύστημα της ηλεκτρονικής διεύθυνσης 127.0.0.1 και το λειτουργικό σύστημα που χρησιμοποιεί. Στο συγκεκριμένο παράδειγμα πρόκειται για έναν apache διακομιστή στην έκδοση 2.4.53 και για ένα λειτουργικό σύστημα Linux βασισμένο στο Debian. Μια ακόμα χρήσιμη πληροφορία που φαίνεται στο παράδειγμα της Εικόνας 4.9, είναι τα αποτελέσματα από τη διενέργεια της σάρωσης με το εργαλείο nmap. Τα αποτελέσματα φαίνονται στο πεδίο ports του φακέλου κατηγορίας nmap. Φαίνεται ότι το εν λόγω υπολογιστικό σύστημα έχει ανοιχτές τις θύρες δικτύου 22 και 80. Είναι γνωστό ότι οι συγκεκριμένες θύρες εξυπηρετούν κατά κανόνα τις συνδέσεις των πρωτοκόλλων ssh και http, αντίστοιχα. Έτσι αντλείται η πληροφορία ότι το σύστημα εκτελεί τις αντίστοιχες υπηρεσίες. Επιπλέον, ολόκληρη η έξοδος των αποτελεσμάτων από την εφαρμογή του εργαλείου nmap υπάρχει στο πεδίο “nmap” του φακέλου κατηγορίας “nmap”, στην οποία αναφέρεται ποιες δικτυακές υπηρεσίες εκτελούνται σε κάθε θύρα δικτύου που ανιχνεύτηκε ότι είναι ανοικτή και δέχεται δικτυακές συνδέσεις.



Εικόνα 4.9: Παράδειγμα σελίδας παρουσίασης πληροφοριών που αφορούν το σύστημα της ηλεκτρονικής διεύθυνσης 127.0.0.1.

4.4 Τεχνολογίες

Σε αυτό το υποκεφάλαιο περιγράφονται οι τεχνολογίες που χρησιμοποιήθηκαν για την ανάπτυξη του γραφικού περιβάλλοντος. Επιπλέον, γίνεται αναφορά σε συγκεκριμένα σημεία του πηγαίου κώδικα τα οποία έχουν προγραμματιστική αξία.

Η Javascript, που συχνά αναφέρεται και ως JS, είναι μια ελαφριά, δυναμική, αντικειμενοστραφής γλώσσα προγραμματισμού η οποία χρησιμοποιεί διερμηνέα (just-in-time JIT). Αυτό σημαίνει ότι τα προγράμματα που γράφονται σε αυτή τη γλώσσα δεν μεταγλωττίζονται ολόκληρα πριν εκτελεστούν από τον υπολογιστή αλλά διερμηνεύονται εντολή προς εντολή την στιγμή που εκτελούνται. Είναι κυρίως γνωστή ως η γλώσσα για ιστοσελίδες. Κατά κανόνα η

Javascript εκτελείται στον φυλλομετρητή του χρήστη και έτσι μπορεί να προγραμματίσει πως θα πρέπει να συμπεριφερθεί το περιεχόμενο της ιστοσελίδας σε περίπτωση εμφάνισης κάποιου συμβάντος, για παράδειγμα ο χρήστης πάτησε κάποιο κουμπί. Το συντακτικό της γλώσσας Javascript είναι εμπνευσμένο από τη γλώσσα Java, αλλά, σε αντίθεση με τη δημοφιλή εσφαλμένη αντίληψη, δεν υπάρχει κάποια σχέση μεταξύ των δύο γλωσσών, πέρα από το παρόμοιο συντακτικό και ονομασία [18].

Για τη διευκόλυνση ανάπτυξης δικτυακών εφαρμογών έχουν αναπτυχθεί πολλά πακέτα βιβλιοθηκών (frameworks) βασισμένα στη γλώσσα Javascript. Τα δημοφιλέστερα από αυτά είναι τα Angular, ReactJS και VueJS. Όλα αυτά τα frameworks αντιμετωπίζουν το ίδιο πρόβλημα με διαφορετικές προσεγγίσεις.

Για την εκπόνηση της εργασίας αυτής επιλέχθηκε το πακέτο βιβλιοθηκών Angular παρότι, σύμφωνα με τη δημοσκόπηση της περίφημης ιστοσελίδας stack overflow, το δημοφιλέστερο πακέτο βιβλιοθηκών το 2021 ήταν το ReactJS, και μάλιστα με σημαντική διαφορά από το Angular που βρέθηκε στην τέταρτη θέση [19]. Όμως, το Angular χρησιμοποιεί, μεταξύ άλλων, το πακέτο βιβλιοθηκών Typescript της γλώσσας Javascript, του οποίου το βασικότερο χαρακτηριστικό είναι ότι προσθέτει επιπλέον συντακτικούς κανόνες με αποτέλεσμα πιο ευανάγνωστο και λιγότερο επιρρεπή σε λάθη κώδικα [20]. Επιπλέον, χρησιμοποιεί ένα σύνθετο σύστημα που αποτελείται από επιμέρους στοιχεία (components). Κάθε στοιχείο αποτελείται από τμήματα κώδικα που μπορούν να παραμετροποιηθούν και να επαναχρησιμοποιηθούν, ελαχιστοποιώντας έτσι την πολυπλοκότητα του κώδικα διευκολύνοντας τη διαδικασία ανάπτυξης εφαρμογών και αυξάνοντας την επίδοση και την ταχύτητα της τελικής εφαρμογής. Ένα ακόμα χαρακτηριστικό του Angular είναι ότι πρόκειται για το πιο ολοκληρωμένο πακέτο βιβλιοθηκών, καθώς προσφέρει πολύ περισσότερες ενσωματωμένες βιβλιοθήκες, σε σχέση με τους ανταγωνιστές του, που βοηθούν στην ευκολότερη και γρηγορότερη ανάπτυξη δικτυακών εφαρμογών. Τέλος, είναι σημαντικό να σημειωθεί ότι το Angular έχει αναπτυχθεί και συντηρείται από τη γνωστή εταιρεία Google και πρόκειται για ένα πακέτο βιβλιοθηκών ανοιχτού κώδικα [21]. Επομένως, για τους παραπάνω λόγους το Angular επιλέχθηκε ως το πακέτο βιβλιοθηκών στο οποίο βασίστηκε η ανάπτυξη του γραφικού περιβάλλοντος του λογισμικού Sn1per.

Ο χειριστής του Sn1per αλληλοεπιδρά με το γραφικό περιβάλλον. Για να μπορεί το γραφικό περιβάλλον, όμως, να αλληλοεπιδρά με το Sn1per απαιτείται μια ενδιάμεση οντότητα η οποία θα μεταφράζει τις ενέργειες που εκτελεί ο χειριστής στο γραφικό περιβάλλον, στις αντίστοιχες εντολές του Sn1per. Για τον λόγο αυτό έχει αναπτυχθεί ένας ενδιάμεσος διακομιστής REST με τον οποίο αλληλοεπιδρά το γραφικό περιβάλλον μέσω του πρωτοκόλλου επικοινωνίας http. Ο διακομιστής με τη σειρά του, αλληλοεπιδρά με το λογισμικό Sn1per και εκτελεί τις ανάλογες ενέργειες. Για να διευκολυνθεί η ανάπτυξη του διακομιστή, καθώς το γραφικό περιβάλλον αναπτύχθηκε με τη χρήση πακέτου βιβλιοθηκών της γλώσσας Javascript, επιλέχθηκε να χρησιμοποιηθεί το αντίστοιχο πακέτο βιβλιοθηκών της ίδιας γλώσσας με το οποίο μπορεί να αναπτυχθεί ένας διακομιστής REST.

Η Javascript συνήθως εκτελείται στον φυλλομετρητή του χρήστη, αλλά μπορεί να εκτελεστεί και σε περιβάλλον διακομιστή με τη χρήση του πακέτου βιβλιοθηκών ανοικτού κώδικα, NodeJS. Μέσω του NodeJS, δίνεται η δυνατότητα εκτέλεσης της γλώσσας Javascript σε έναν διακομιστή ή ακόμα και σε έναν προσωπικό υπολογιστή. Πολλές βιβλιοθήκες έχουν γραφεί πάνω στο NodeJS, όπως είναι η ExpressJS που επιτρέπει τη δημιουργία και τον προγραμματισμό ενός διακομιστή REST. Η ExpressJS είναι η βιβλιοθήκη που επιλέχθηκε για τη δημιουργία του διακομιστή καθώς είναι η δημοφιλέστερη επιλογή [22].

4.4.1 Ανάπτυξη δικτυακού περιβάλλοντος γραμμής εντολών πραγματικού χρόνου

Ένα σημαντικό στάδιο στην ανάπτυξη του γραφικού περιβάλλοντος είναι η δημιουργία ενός δικτυακού περιβάλλοντος γραμμής εντολών. Όπως έχει περιγραφεί στο υποκεφάλαιο 3.1, το Sn1per εκτελείται σε περιβάλλον γραμμής εντολών και πολλές χρήσιμες πληροφορίες, για την πορεία εκτέλεσης της τρέχουσας διαδικασίας σάρωσης, εμφανίζονται μόνο στο περιβάλλον

γραμμής εντολών και δεν αποθηκεύονται. Έτσι, μετά την ολοκλήρωση της σάρωσης είναι πιθανόν κάποια σημαντικά αποτελέσματα να διαφύγουν της προσοχής του χειριστή.

Για να μην χαθεί η πληροφορία αυτή, στην παρούσα εργασία αναπτύχθηκε το δικτυακό περιβάλλον γραμμής εντολών. Για να λειτουργήσει επιτυχώς ένα τέτοιο περιβάλλον είναι απαραίτητη η επικοινωνία σε πραγματικό χρόνο μεταξύ διακομιστή και γραφικού περιβάλλοντος. Με τον συμβατικό τρόπο επικοινωνίας, το γραφικό περιβάλλον πρέπει κάθε φορά να αποστέλνει στον διακομιστή νέο αίτημα για ενημέρωση και να περιμένει να λάβει την αντίστοιχη απάντηση. Για να διευκολυνθεί η μεταξύ τους επικοινωνία και να μειωθεί στο ελάχιστο η δικτυακή κίνηση, χρησιμοποιήθηκε η τεχνολογία WebSocket. Το WebSocket είναι μια εξειδικευμένη τεχνολογία που επιτρέπει την αμφίδρομη επικοινωνία μεταξύ διακομιστή και γραφικού περιβάλλοντος και διατηρώντας μόνιμα ανοικτή μία σύνδεση επιτρέπει τον διακομιστή να ενημερώνει σε πραγματικό χρόνο το γραφικό περιβάλλον, μόλις εμφανιστούν νέα στοιχεία στο περιβάλλον γραμμής εντολών της διαδικασίας σάρωσης [23]. Για την άμεση ενημέρωση της εμφάνισης των νέων στοιχείων χρησιμοποιήθηκε μία συνήθης πρακτική του λειτουργικού συστήματος Linux, η ανακατεύθυνση σε αρχείο. Συγκεκριμένα, για να υπάρχει πρόσβαση στην έξοδο όλων των αποτελεσμάτων της διαδικασίας σάρωσης που εμφανίζονται στο περιβάλλον γραμμής εντολών ακόμα και μετά την ολοκλήρωση της εκτέλεσής της ή και τερματισμού της σάρωσης, γίνεται ανακατεύθυνση της εξόδου προς το αρχείο scan.out. Το αρχείο αυτό αποθηκεύεται στον ίδιο φάκελο με αυτόν που αποθηκεύονται τα αποτελέσματα της διαδικασίας σάρωσης. Όταν ο διακομιστής λάβει ένα αίτημα για την παρακολούθηση της εξόδου των αποτελεσμάτων μίας σάρωσης, τότε ξεκινάει μια νέα επικοινωνία WebSocket και χρησιμοποιώντας τη βιβλιοθήκη ShellJS της Javascript, για να μπορεί να εκτελέσει εντολές του λειτουργικού συστήματος Linux, εκτελεί την εντολή tail. Συγκεκριμένα, η εντολή συντάσσεται με τις παραμέτρους -n 1000 και -f. Η εντολή tail έχει προκαθοριστεί να επιστρέφει τις τελευταίες πέντε γραμμές του δοθέντος αρχείου. Με την παράμετρο -n 1000, η εντολή θα επιστρέψει τις τελευταίες χίλιες γραμμές και με την παράμετρο -f, η εντολή θα παραμείνει ενεργή και όταν μία νέα γραμμή θα προστίθεται στο αρχείο θα ενημερώνεται και η έξοδός της. Τέλος, γίνεται χρήση μιας ειδικής διεργασίας που προσφέρει η βιβλιοθήκη ShellJS, η οποία καλείται όταν η εκτελούμενη εντολή έχει κάποια αλλαγή στην έξοδό της. Με τον τρόπο αυτό όταν εντοπιστεί κάποια αλλαγή στην έξοδο της ειδικά παραμετροποιημένης εντολής tail, τότε στέλνεται αντίστοιχη ενημέρωση στο γραφικό περιβάλλον μέσω της σύνδεσης WebSocket που έχει ήδη ενεργοποιηθεί.

Επίσης, επιλύθηκε ένα θέμα ασυμφωνίας που αφορούσε τα χρώματα του κειμένου της εξόδου της διαδικασίας σάρωσης (βλ. Εικόνα 3.1) με αυτά που τελικά μεταφέρονται στο δικτυακό περιβάλλον. Αυτό είναι αναμενόμενο καθώς η μορφοποίηση του κειμένου που παράγεται κατά την εφαρμογή του Sn1per είναι τέτοια ώστε να παίρνει τα κατάλληλα χρώματα στο περιβάλλον γραμμής εντολών. Για το δικτυακό περιβάλλον θα έπρεπε αυτή η μορφοποίηση να μεταφραστεί στην αντίστοιχη που χρησιμοποιεί η γλώσσα σήμανσης HTML. Για τον λόγο αυτό χρησιμοποιήθηκε η βιβλιοθήκη ansi-to-html της Javascript.

Κατά την ανάπτυξη του γραφικού περιβάλλοντος, λήφθηκε ιδιαίτερη μέριμνα για τον σχεδιασμό ενός ασφαλούς τρόπου με τον οποίο ο διακομιστής παρέχει στο γραφικό περιβάλλον πρόσβαση στην έξοδο όλων των αποτελεσμάτων της διαδικασίας σάρωσης. Εφόσον αυτή η άδεια πρόσβασης γίνεται μέσω της φιλικής ονομασίας, ο διακομιστής πριν αποστείλει το οτιδήποτε στο γραφικό περιβάλλον, επαληθεύει πρώτα ότι το αίτημα αφορά κάποιο αρχείο με προκαθορισμένο όνομα. Σε περίπτωση που δεν είναι ένα από τα αποδεκτά ονόματα, τότε ο διακομιστής επιστρέφει απλώς τη λέξη “No”.

Με τη σύνθεση των τεχνικών αυτών δημιουργήθηκε το δικτυακό περιβάλλον γραμμής εντολών του γραφικού περιβάλλοντος. Καθώς πρόκειται για ένα περιβάλλον γραμμής εντολών που προορίζεται μόνο για την προβολή της εξόδου των αποτελεσμάτων κάποιας διαδικασίας σάρωσης, θεωρήθηκε πιο ασφαλές να μην υπάρχει η δυνατότητα αλληλεπίδρασης με τον χειριστή. Στην Εικόνα 4.10 φαίνεται ο πηγαίος κώδικας του ενδιάμεσου διακομιστή REST.

```
//
// ~~~ WEBSOCKET ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
//
const wss = new ws.Server({ server });
wss.on('connection', sock => {
  sock.on('message', function(msg) {
    const s = msg.toString();
    if(s.startsWith("pls ")) {
      const scan = msg.toString().substring(4);
      const scans = fs.readdirSync(workspacesFolder, {withFileTypes: true}).filter(dir => dir.isDirectory()).map(dirent => dirent.name);
      if(scans.includes(scan))
      {
        const path = `${workspacesFolder}${scan}`;
        const e = shell.exec(`tail -n1000 -f ${path}/scan.out`, {async: true, silent: true});
        e.stdout.on('data', function(data) {
          sock.send(convert.toHtml(data));
        })
      }
      else {
        sock.send("No");
      }
    }
    else if(s === "scans") {
      wsCons.push(sock);
      const scansFile = fs.readFileSync(scanStatusPath)
      scanStatus = JSON.parse(scansFile);
      wsCons.forEach((s) => s.send(JSON.stringify(scanStatus)))
    }
  })
})
```

Εικόνα 4.10: Ο πηγαίος κώδικας του ενδιάμεσου διακομιστή REST.

4.4.2 Συγκέντρωση αποτελεσμάτων της διαδικασίας σάρωσης

Το Sn1per αποθηκεύει τα αποτελέσματα της κάθε διαδικασίας σάρωσης σε ξεχωριστά αρχεία κατηγοριοποιημένα σε φακέλους κατηγορίας, όπως έχει περιγραφεί στο υποκεφάλαιο 3.2. Για την καλύτερη παρουσίαση των αποτελεσμάτων της σάρωσης, ετοιμάστηκε μια διεργασία για τη συγκέντρωση όλων των αποτελεσμάτων σε ένα αρχείο. Η διεργασία αυτή εκτελείται μετά την ολοκλήρωση κάθε διαδικασίας σάρωσης.

Το αρχείο που δημιουργείται χρησιμοποιεί τη μορφοποίηση JSON (JavaScript Object Notation). Η μορφοποίηση JSON είναι μια τυποποιημένη μορφή αναπαράστασης δεδομένων που ακολουθεί τους συντακτικούς κανόνες της γλώσσας προγραμματισμού JavaScript. Αυτή η μορφή συνήθως χρησιμοποιείται για τη μεταφορά δεδομένων μέσω δικτύου [24].

Για τη διεργασία δημιουργίας του αρχείου JSON ετοιμάστηκε ένα πρόγραμμα σε γλώσσα Python. Η Python είναι μια διερμηνευμένη, αντικειμενοστραφής, υψηλού επιπέδου γλώσσα προγραμματισμού με δυναμική σημασιολογία. Οι ενσωματωμένες δομές δεδομένων υψηλού επιπέδου, σε συνδυασμό με τη δυναμική τυποποίηση και τη δυναμική δέσμευση μνήμης, την καθιστούν πολύ ελκυστική για την ταχεία ανάπτυξη εφαρμογών [25].

Παρακάτω εξηγείται ο τρόπος λειτουργίας και η ανάλυση του κώδικα της διεργασίας δημιουργίας του αρχείου JSON. Ο κώδικας φαίνεται στις Εικόνες 4.11, 4.12 και 4.13. Καθώς γίνεται διαχωρισμός μεταξύ των πληροφοριών γενικού περιεχομένου και αυτών που αντλήθηκαν από το προστατευόμενο υπολογιστικό σύστημα, το πρώτο στάδιο του κώδικα είναι η συγκέντρωση όλων των διαφορετικών ηλεκτρονικών διευθύνσεων IP. Ανατρέχοντας στα περιεχόμενα του αρχείου στη θέση domains/targets-all-sorted.txt, ανακτώνται όλες οι ηλεκτρονικές διευθύνσεις που ανιχνεύθηκαν στο υπολογιστικό σύστημα που σαρώθηκε. Η διαδικασία αυτή φαίνεται στις γραμμές 17 με 20 του κώδικα της Εικόνας 4.11. Στις γραμμές 22 έως 24, προετοιμάζεται μια λίστα αντικειμένων, για να εισαχθούν τα αποτελέσματα που αφορούν την κάθε μια ηλεκτρονική διεύθυνση.

```
PATH = "/usr/share/sniper/loot/workspace"
workspace = sys.argv[1]

loot = os.path.join(PATH, workspace)
directories = os.listdir(loot)
dirs = list(filter(lambda d: d not in ["output", "screenshots", "scans", "notes", "output.json", "reports", "scan.out", "scan.sh", "domains"], directories))
json_obj = {
    "domains": {}
}

f = open(path.join(loot, "domains", "targets-all-sorted.txt"))
r = f.read().strip().split("\n")
json_obj["domains"]["scanned"] = r
f.close()

domains = r
for d in domains:
    json_obj[d] = {}

f = open(path.join(loot, "domains", "targets-all-unscanned.txt"))
r = f.read().strip().split("\n")
json_obj["domains"]["unscanned"] = r
f.close()

for ip in domains:
    json_obj[ip] = {}
    for dir in dirs:
        json_obj[ip][dir] = {}
```

Εικόνα 4.11: Μέρος του κώδικα για τη διεργασία συγκέντρωσης των αποτελεσμάτων της σάρωσης σε αρχείο JSON

Αφού ολοκληρωθεί το πρώτο αυτό στάδιο της προετοιμασίας, το πρόγραμμα συνεχίζει στη διεργασία συγκέντρωσης των αποτελεσμάτων. Προσπελαύνει ένα προς ένα όλα τα περιεχόμενα αρχεία αποτελεσμάτων του κάθε φακέλου κατηγορίας. Το κάθε αρχείο αποτελεσμάτων περνάει από δύο ελέγχους, ώστε να αποφανθεί εάν θα αποθηκευτούν τα περιεχόμενά του στο τελικό παραγόμενο αρχείο JSON. Ο πρώτος έλεγχος είναι η κατάληξη του αρχείου αποτελεσμάτων να μην είναι .xml, .html, .old, .diff, .pdf ή το κενό. Αυτό γίνεται διότι τα αρχεία αποτελεσμάτων με κατάληξη .old και .diff έχουν παλαιότερες πληροφορίες από προηγούμενες σαρώσεις και τα αρχεία αποτελεσμάτων με κατάληξη .pdf είναι αρχεία κειμένου τα οποία δεν γίνεται να προβληθούν στον φυλλομετρητή. Τα αρχεία αποτελεσμάτων με κατάληξη .xml και .html περιέχουν αποτελέσματα μορφοποιημένα στις γλώσσες σήμανσης XML και HTML αντίστοιχα. Όλα τα εργαλεία που χρησιμοποιεί το Sn1per κατά τη διαδικασία σάρωσης αποθηκεύουν τα ευρήματά τους σε αρχεία απλού κειμένου (.txt). Ορισμένα εργαλεία του Sn1per αποθηκεύουν, επιπλέον, τα αποτελέσματά τους και μορφοποιημένα σε κάποια από τις δύο προαναφερθείσες γλώσσες σήμανσης. Προκειμένου όλα τα αποτελέσματα να ακολουθούν μια κοινή μορφοποίηση προβολής, γίνεται χρήση μόνο των αρχείων αποτελεσμάτων απλού κειμένου.

Ο δεύτερος έλεγχος αφορά την ιδιαιτερότητα του Sn1per να αποθηκεύει τα αποτελέσματα μιας διαδικασίας σάρωσης σε δύο τύπους αρχείων. Δηλαδή, στα αρχεία στο όνομα των οποίων περιέχεται η λέξη “sorted” και σε αυτά όπου περιέχεται η λέξη “unsorted”. Στα αρχεία με το “sorted” αποθηκεύονται μόνο τα αποτελέσματα της τελευταίας διαδικασίας σάρωσης, ενώ στα αρχεία με το “unsorted” περιέχονται τα αποτελέσματα που παράχθηκαν από όλες τις διαδικασίες σάρωσης που έχουν εκτελεστεί κατά καιρούς. Επειδή, λοιπόν ο στόχος του προγράμματος της διεργασίας συγκέντρωσης αποτελεσμάτων είναι η ενημέρωση του γραφικού περιβάλλοντος με τα αποτελέσματα της εκάστοτε διαδικασίας σάρωσης, τα αρχεία που περιέχουν στο όνομα τους τη λέξη “unsorted” θα πρέπει να αγνοηθούν.

Το πρόγραμμα διαβάζει τα περιεχόμενα καθενός από τα αρχεία αποτελεσμάτων που έχει περάσει τους δύο παραπάνω ελέγχους και τα αποθηκεύει στη λίστα αντικειμένων που έχει προετοιμαστεί. Τα περιεχόμενα των αρχείων αποτελεσμάτων τα οποία αφορούν κάποια υπολογιστική συσκευή, αποθηκεύονται με τέτοιο τρόπο ώστε να συσχετιστούν με την ηλεκτρονική διεύθυνση της συσκευής που αφορούν. Σε διαφορετική περίπτωση τα περιεχόμενα αποθηκεύονται χωρίς τη συσχέτιση με κάποια ηλεκτρονική διεύθυνση. Επιπλέον, για κάθε αρχείο γίνεται ο αντίστοιχος συσχετισμός με τον φάκελο κατηγορίας στον οποίο ανήκει. Με τον συσχετισμό αυτό μπορεί εύκολα τα γραφικό περιβάλλον να παρουσιάζει τα αποτελέσματα μιας διαδικασίας σάρωσης κατηγοριοποιημένα, όπως έχει περιγραφεί στο υποκεφάλαιο 4.3.

```
for dir in dirs:
    json_obj[dir] = {}
    for (dirpath, d, files) in os.walk(os.path.join(loot, dir)):
        for file in files:
            name = os.path.splitext(file)
            if not any(word in name[0] for word in ["unsorted"]) and (name[1] not in [".xml", ".html", ".old", ".diff", ".pdf", ""]):
                f = open(os.path.join(dirpath, file), "r", encoding="unicode_escape")
                found = False
                if name[1] == ".json":
                    f = open(os.path.join(dirpath, file), "r")
                    contents = json.loads(f.read())
                else:
                    contents = f.readlines()
                    contents = [c.strip() for c in contents]
                for ip in domains:
                    if ip in [s.strip() for s in re.split("-|_", name[0])]:
                        n = name[0].replace("-"+ip, "")
                        if len(contents) == 1:
                            json_obj[ip][dir][n] = contents[0]
                        else:
                            json_obj[ip][dir][n] = contents
                        f.close()
                        found = True
                        break
                if not found:
                    if ip in contents:
                        json_obj[ip][dir][name[0]] = True
                    else:
                        if len(contents) == 1:
                            json_obj[dir][name[0]] = contents[0]
                        else:
                            json_obj[dir][name[0]] = contents
                        f.close()
        break
```

Εικόνα 4.12: Παρόμοια με την Εικόνα 4.11

Το τελευταίο κομμάτι του κώδικα της διεργασίας συγκέντρωσης των αποτελεσμάτων είναι η αποθήκευση της τελικής δομής JSON στο αρχείο output.json. Με αυτόν τον τρόπο όταν χρειαστεί το γραφικό περιβάλλον να προβάλει τα αποτελέσματα της διαδικασίας σάρωσης, ο ενδιάμεσος διακομιστής θα αποστείλει το αρχείο αυτό.

```
# ——— OUTPUT ——————————————————————————————————————————————————————

f = open(os.path.join(loot, "output.json"), "w")
f.write(json.dumps(json_obj, indent=4))
f.close()
print("Created JSON file at ", os.path.join(loot, "output.json"))

```

Εικόνα 4.13: Παρόμοια με την Εικόνα 4.11

ΚΕΦΑΛΑΙΟ 5: ΠΡΟΣΘΕΤΕΣ ΛΕΙΤΟΥΡΓΙΕΣ

Στα πλαίσια της εργασίας αυτής έχουν προστεθεί δύο επιπλέον λειτουργίες στο Sn1per, πέρα από το γραφικό περιβάλλον. Η μία αφορά την προστασία του οργανισμού από επιθέσεις τύπου phishing. Η άλλη αφορά την ανακάλυψη πληροφοριών, μέσω Google Hacking, που μπορεί να βρίσκονται εσφαλμένα ελεύθερα στο διαδίκτυο.

5.1 Phishing

Το "phishing" είναι ένας τύπος επίθεσης κοινωνικής μηχανικής (social engineering) που χρησιμοποιείται συχνά για την κλοπή δεδομένων χρηστών, συμπεριλαμβανομένων των διαπιστευτηρίων συνόδου και του αριθμού πιστωτικών καρτών. Ο τρόπος που εκτελείται είναι όταν ένας επιτιθέμενος, υποδυόμενος μία έμπιστη οντότητα, αποστέλλει κάποιο μήνυμα, για παράδειγμα ηλεκτρονικού ταχυδρομείου ή κάποιο άλλο μήνυμα κειμένου, και προσπαθεί να εξαπατήσει τον λήπτη, ώστε να ανοίξει κάποιον κακόβουλο σύνδεσμο, ο οποίος μπορεί να οδηγήσει στην εγκατάσταση κακόβουλου λογισμικού, ή στην αποκάλυψη ευαίσθητων πληροφοριών στον επιτιθέμενο. Ένας οργανισμός που υποκύπτει σε μια τέτοια επίθεση συνήθως υφίσταται σοβαρές οικονομικές απώλειες καθώς και μεγάλο αρνητικό αντίκτυπο στη φήμη του και στην εμπιστοσύνη των πελατών του. Ανάλογα με την έκταση, μια απόπειρα phishing μπορεί να εξελιχθεί σε ένα περιστατικό ασφαλείας από το οποίο ένας οργανισμός θα δυσκολευτεί να ανακάμψει.

Οι επιτιθέμενοι καταβάλουν μεγάλες προσπάθειες στον σχεδιασμό αυτών των μηνυμάτων phishing, ώστε να μιμούνται μηνύματα ηλεκτρονικού ταχυδρομείου που συνήθως αποστέλλονται από μια έμπιστη οντότητα. Η χρήση της ίδιας διατύπωσης και των ίδιων γραμματοσειρών, λογότυπων και υπογραφών κάνει τα μηνύματα να φαίνονται νόμιμα και αυθεντικά. Ειδικότερα, στην περίπτωση που στο στόχαστρο των επιτιθέμενων έχει μπει ένας συγκεκριμένος οργανισμός, μπορούν ακόμα πιο εύκολα να εξαπατήσουν ανυποψίαστους υπαλλήλους αποστέλλοντας τέτοια μηνύματα μέσω παραβιασμένων λογαριασμών ηλεκτρονικού ταχυδρομείου του ίδιου του οργανισμού. Έτσι, τα κακόβουλα αυτά μηνύματα φαίνεται ότι αποστέλλονται από συναδέλφους [26].

Ένας οργανισμός είναι ιδιαίτερα δύσκολο να μπορέσει να προστατευτεί αποτελεσματικά από επιθέσεις τέτοιου τύπου. Ένα σημαντικό μέτρο προστασίας είναι η χρήση του Two-factor authentication (2FA). Για παράδειγμα ένας υπάλληλος για τη σύνδεση στις ψηφιακές υπηρεσίες του οργανισμού, θα πρέπει να εισάγει και έναν επιπλέον κωδικό, εκτός από τα κοινά διαπιστευτήρια συνόδου, που του έχει σταλεί στο κινητό του τηλέφωνο με αποστολή γραπτού μηνύματος. Έτσι σε περίπτωση που κάποια κακόβουλη οντότητα τυχαίνει να γνωρίζει τα διαπιστευτήρια συνόδου κάποιου υπαλλήλου, δεν θα μπορέσει να συνδεθεί επιτυχημένα εάν δεν έχει πρόσβαση και στο κινητό τηλέφωνο του κατόχου του. Ένα άλλο προληπτικό μέτρο είναι η οργάνωση εκπαιδευτικών σεμιναρίων για τους υπαλλήλους και η ενημέρωσή τους σχετικά με τους κινδύνους που επιφυλάσσει μια τέτοια επίθεση καθώς και τρόπους άμεσου εντοπισμού τέτοιων μηνυμάτων. Ένα τελευταίο μέτρο είναι η επιβολή πολιτικής συχνής αλλαγής κωδικών πρόσβασης. Με αυτό το μέτρο ακόμα και στην περίπτωση που έχει διαρρεύσει ο κωδικός πρόσβασης κάποιου λογαριασμού, δεν θα μπορεί να χρησιμοποιηθεί καθώς θα θεωρείται πλέον ληγμένος.

5.1.1 Haveibeenpwned

Το haveibeenpwned (https://haveibeenpwned.com/) είναι μια ιστοσελίδα η οποία προσφέρει μια απλή αλλά πολύ σημαντική υπηρεσία. Ο χρήστης εισάγει μια διεύθυνση ηλεκτρονικού ταχυδρομείου και η ιστοσελίδα την αναζητά σε μια τεράστια βάση δεδομένων από διευθύνσεις που ήδη έχουν πέσει θύματα σε μια προηγούμενη επιτυχημένη επίθεση και ο κωδικός πρόσβασης τους έχει διαρρεύσει. Σε περίπτωση που η διεύθυνση βρεθεί μέσα σε αυτή τη βάση ο κωδικός πρόσβασης θα πρέπει να αλλαχθεί άμεσα, προκειμένου να αποκλειστούν πιθανές

προσπάθειες από τρίτους για σύνδεση. Αυτός ο ιστότοπος προέκυψε μετά τη μεγαλύτερη, για εκείνη την εποχή, παραβίαση λογαριασμών πελατών, της Adobe (19 Οκτωβρίου 2019). Το haveibeenpwned είναι μια απλή στη χρήση και δωρεάν υπηρεσία που επιτρέπει την πρόσβαση σε όλους. Επιπλέον, η υπηρεσία που προσφέρει δεν είναι απλώς μια θετική ή αρνητική απάντηση στην ερώτηση εάν η δοθείσα ηλεκτρονική διεύθυνση είναι θύμα κάποιας διαρροής κωδικών. Εάν βρεθεί, τότε η υπηρεσία επιστρέφει μια λίστα με όλες τις γνωστές διαρροές στις οποίες έχει ανιχνευθεί ότι έχει υποστεί η συγκεκριμένη διεύθυνση, μια σύντομη περιγραφή για την κάθε μια διαρροή καθώς και την ημερομηνία που αυτή συνέβη. Για την αποφυγή επιθέσεων τύπου phishing από εσωτερικούς λογαριασμούς του δικτύου ενός οργανισμού, που μπορούν εύκολα να κερδίσουν την εμπιστοσύνη των υπόλοιπων υπαλλήλων, είναι σημαντική η τακτική επίσκεψη και αναζήτηση για πιθανούς τέτοιους παραβιασμένους λογαριασμούς στην ιστοσελίδα αυτή και λήψη άμεσων μέτρων [27].

5.1.2 Ενσωμάτωση στο Sn1per της διεργασίας επικοινωνίας με το haveibeenpwned

Η λειτουργία αυτής της υπηρεσίας έχει ενσωματωθεί στο Sn1per. Ήδη το Sn1per, με τη χρήση της λειτουργίας osint, μπορεί να εντοπίσει εκτεθειμένες διευθύνσεις ηλεκτρονικού ταχυδρομείου. Με την επιπρόσθετη λειτουργία, εάν μετά την ολοκλήρωση μίας διαδικασίας σάρωσης έχουν εντοπιστεί εκτεθειμένες διευθύνσεις, τότε ενεργοποιείται μια διεργασία αναζήτησης αυτών των διευθύνσεων στην ιστοσελίδα haveibeenpwned, για πιθανές διαρροές των κωδικών πρόσβασής τους.

Για να αντληθούν οι πληροφορίες που αφορούν κάποια διεύθυνση ηλεκτρονικού ταχυδρομείου γίνεται ένα αίτημα στη διεύθυνση https://haveibeenpwned.com/unifiedsearch/ ακολουθούμενο από την εκτεθειμένη διεύθυνση ηλεκτρονικού ταχυδρομείου. Οι απαντήσεις που μπορεί να ληφθούν είναι είτε ότι δεν βρέθηκε η διεύθυνση στη βάση δεδομένων, είτε ότι βρέθηκε μαζί με τις σχετικές πληροφορίες, είτε ότι η ιστοσελίδα haveibeenpwned έχει αποκλείσει την πρόσβαση στο λογισμικό Sn1per. Το τελευταίο μπορεί να συμβεί διότι η ιστοσελίδα προστατεύεται από ένα ειδικό πρόγραμμα εντοπισμού δικτυακής κίνησης όταν αυτή προέρχεται από αυτοματοποιημένα προγράμματα και όχι από άνθρωπο. Δηλαδή, όταν γίνεται χρήση απλών δικτυακών αιτημάτων στην ιστοσελίδα του haveibeenpwned, τότε το αίτημα απορρίπτεται. Για τον λόγο αυτό, έχει χρησιμοποιήθηκε η ειδική βιβλιοθήκη cloudscraper, ανοικτού κώδικα, της γλώσσας Python, (https://github.com/VeNoMouS/cloudscraper). Η βιβλιοθήκη αυτή προσομοιώνει τη συμπεριφορά που θα είχε ένας φυλλομετρητής εάν τον χρησιμοποιούσε απευθείας ένας άνθρωπος. Εκμεταλλευόμενο αυτή τη λειτουργία, το πρόγραμμα καταφέρνει να παρακάμψει τους ελέγχους του διακομιστή της ιστοσελίδας. Καθώς, όμως, δεν επιτυγχάνεται πάντα η παράκαμψη αυτή, σε περίπτωση που το πρόγραμμα λάβει από την ιστοσελίδα απορριπτική απάντηση, επειδή θεωρήθηκε ως αυτοματοποιημένο πρόγραμμα, τότε μετά την πάροδο κάποιου χρονικού διαστήματος, το πρόγραμμα επαναλαμβάνει την αποστολή του αιτήματος. Εάν η ιστοσελίδα δεχτεί το αίτημα, στην περίπτωση που επιβεβαιωθεί ότι η ηλεκτρονική διεύθυνση είναι εκτεθειμένη, τότε αποστέλλονται σε ένα αρχείο τύπου JSON και οι σχετικές πληροφορίες για την εμπλοκή της σε επισημασμένες διαρροές. Η τελική λίστα, με όλες τις πληροφορίες που αφορούν σε εκτεθειμένες διευθύνσεις ηλεκτρονικού ταχυδρομείου, αποθηκεύεται μέσα στον φάκελο κατηγορίας osint, όπως και η αρχική λίστα των διευθύνσεων που εντοπίστηκαν από τη διαδικασία σάρωσης του Sn1per.

Στην Εικόνα 5.1 παρουσιάζεται ο πηγαίος κώδικας σε γλώσσα προγραμματισμού Python που εκτελεί την παραπάνω λειτουργία. Συγκεκριμένα, έχει συνταχθεί μια μέθοδος επικοινωνίας με την ιστοσελίδα του haveibeenpwned. Πρώτα, ελέγχει την απάντηση που λαμβάνει ύστερα από κάθε αίτημα που αποστέλλει, για το εάν εντοπίστηκε ως αυτοματοποιημένο πρόγραμμα και αποκλείστηκε από την ιστοσελίδα. Εάν έχει αποκλειστεί, τότε τερματίζει και επιστρέφει την τιμή “ψευδές”, όπως φαίνεται στις γραμμές 20 με 23, και ύστερα από κάποιο, όχι σταθερό, χρονικό διάστημα (όπως φαίνεται στις γραμμές 36 με 41) επαναλαμβάνει την αποστολή του αιτήματος. Όταν η ιστοσελίδα αποδεχθεί το αίτημα ελέγχου της συγκεκριμένης διεύθυνσης ηλεκτρονικού

ταχυδρομείου, τότε αρχίζει η συλλογή των σχετικών πληροφοριών που αποστέλλει η ιστοσελίδα και η αποθήκευση τους σε ένα αρχείο JSON, όπως φαίνεται στις γραμμές 25 με 29.

```
 8
 9  def checkMails():
10      scraper = cloudscraper.create_scraper(interpreter='nodejs')
11      scraper.get("https://haveibeenpwned.com/unifiedsearch/aaa")
12      sleep(random()*3)
13      if len(sys.argv) == 3:
14          dir = path.dirname(sys.argv[1])
15          with open(sys.argv[1], "r") as emails:
16              result = []
17              for email in emails:
18                  urlencoded = quote(email.strip())
19                  req = scraper.get(f"https://haveibeenpwned.com/unifiedsearch/{urlencoded}")
20                  if req.status_code == 403 or req.status_code == 429:
21                      print(f"Got resonse code {req.status_code}")
22                      del scraper
23                      return False
24                  resp = str(req.text).strip()
25                  if req.status_code == 200:
26                      print(f"{email.strip()} is PWNED")
27                      j = json.loads(resp)
28                      j["email"] = email.strip()
29                      result.append(j)
30          domain = sys.argv[2]
31          with open(path.join(dir, f"pwned_emails-{domain}.json"), "w") as pwned:
32              pwned.write(json.dumps(result, indent=1))
33
34
35  if __name__ == "__main__":
36      res = checkMails()
37      while res == False:
38          w = round(random()*30 + 60, 1)
39          print(f"Retrying after {w} secconds")
40          sleep(w)
41          res = checkMails()
42
```

Εικόνα 5.1: Ο πηγαίος κώδικας επικοινωνίας με την ιστοσελίδα του haveibeenpwned

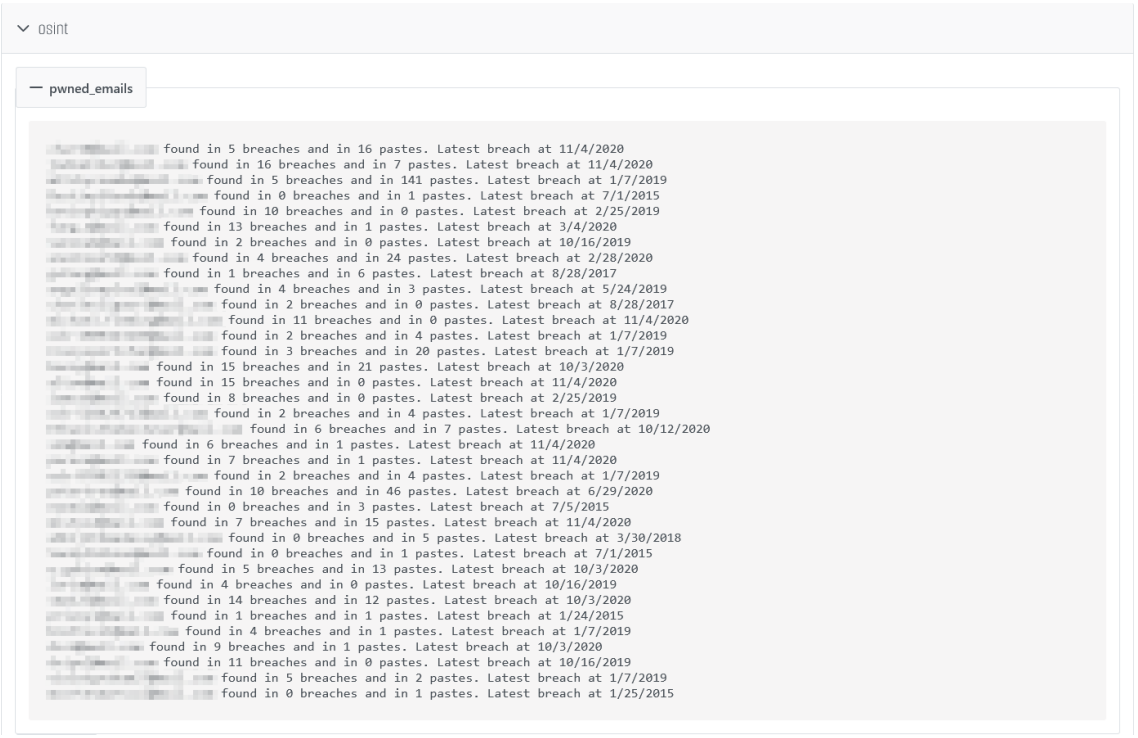
Τέλος, έχει προστεθεί μια επιπλέον παράμετρος στα αρχεία διαμόρφωσης του Sn1per, που σχετίζεται με την ενεργοποίηση ή όχι της λειτουργίας επικοινωνίας με την ιστοσελίδα του haveibeenpwned. Η παράμετρος αυτή ονομάστηκε HAVEIBEENPWNED και μπορεί να λάβει την τιμή 1 ή 0, που υπαγορεύοντας την εκτέλεση ή όχι του ελέγχου αυτού. Στην Εικόνα 5.2 φαίνονται οι σχετικές γραμμές κώδικα που έχουν προστεθεί στο αρχείο εντολών που αφορά τον τρόπο εφαρμογής osint.

```
78  if [[ "$HAVEIBEENPWNED" == "1" ]]; then
79      echo -e "${OKGREEN}==========${RESET}x${OKGREEN}[`date +"%Y-%m-%d](%H:%M)"`${RESET}x"
80      echo -e "$OKRED CHECKING FOR PWNED EMAILS $RESET"
81      echo -e "${OKGREEN}==========${RESET}x${OKGREEN}[`date +"%Y-%m-%d](%H:%M)"`${RESET}x"
82      python3 $INSTALL_DIR/bin/haveibeenpwned.py $LOOT_DIR/osint/email-format-$TARGET.txt $TARGET
83  fi
```

Εικόνα 5.2: Δήλωση της παραμέτρου HAVEIBEENPWNED

Στην Εικόνα 5.3, φαίνεται ένα μέρος των αποτελεσμάτων της διαδικασίας σάρωσης της ηλεκτρονικής διεύθυνσης ενός γνωστού παρόχου υπηρεσίας ηλεκτρονικού ταχυδρομείου, όπως

αυτή φαίνεται μέσα από το γραφικό περιβάλλον που δημιουργήθηκε για το Sn1per. Συγκεκριμένα, στην Εικόνα φαίνονται τα αποτελέσματα της λειτουργίας επικοινωνίας με την ιστοσελίδα του haveibeenpwned. Τα αποτελέσματα παρουσιάζονται σε μορφή λίστας, όπου αναφέρονται η διεύθυνση ηλεκτρονικού ταχυδρομείου, το πλήθος των περιπτώσεων που έχει εντοπιστεί η εν λόγω διεύθυνση σε κάποιο περιστατικό διαρροής δεδομένων και τέλος, η ημερομηνία της εκάστοτε πιο πρόσφατης διαρροής. Ειδικότερα, η τελευταία πληροφορία αποσκοπεί στην υποβοήθηση της έρευνας για εκτεθειμένες ηλεκτρονικές διευθύνσεις και τη λήψη άμεσων μέτρων προστασίας. Σε περίπτωση που οι κωδικοί πρόσβασης κάποιας ηλεκτρονικής διεύθυνσης που βρίσκεται στη λίστα δεν έχουν αλλάξει μετά την ημερομηνία της τελευταίας διαρροής, τότε είναι πολύ πιθανό να υπάρχουν κακόβουλες οντότητες που γνωρίζουν τους τρέχοντες κωδικούς πρόσβασης αυτών των διευθύνσεων. Για λόγους προστασίας των προσωπικών δεδομένων, σύμφωνα με τον Γενικό Κανονισμό Προσωπικών Δεδομένων (GDPR), ο πάροχος της υπηρεσίας και οι εκτεθειμένες διευθύνσεις ηλεκτρονικού ταχυδρομείου δεν αναφέρονται ούτε απεικονίζονται.



Εικόνα 5.3: Αποτελέσματα της διαδικασίας σάρωσης της διεύθυνσης γνωστού παρόχου υπηρεσίας ηλεκτρονικού ταχυδρομείου, που αφορούν τη λειτουργία επικοινωνίας με την ιστοσελίδα του haveibeenpwned

5.2 Google Hacking

Το “Google Hacking” είναι η διαδικασία υποβολής εξειδικευμένων ερωτημάτων, στη γνωστή μηχανή αναζήτησης Google, που αποσκοπεί στην άντληση ενδιαφέρουσων και συνήθως ευαίσθητων πληροφοριών που μπορεί να κυκλοφορούν ελεύθερα στο διαδίκτυο. Στις περισσότερες περιπτώσεις, τέτοιες πληροφορίες έχουν δημοσιευτεί από λανθασμένη ρύθμιση του χρήστη.

Η διαδικασία του “Google Hacking” διαδόθηκε το 2000 από τον Johnny Long, έναν επαγγελματία hacker, ο οποίος άρχισε να καταγράφει αυτά τα ερωτήματα σε μια βάση δεδομένων

γνωστή ως Google Hacking Database. Ο Long επινόησε τον όρο “Google Dork” για να αναφερθεί σε “ένα ανόητο ή ανίκανο άτομο που το αποκαλύπτει το Google”. Αυτό είχε σκοπό να επιστήσει την προσοχή στο γεγονός ότι αυτό δεν ήταν ένα “πρόβλημα της Google”, αλλά μάλλον το αποτέλεσμα, συχνά, ακούσιων, κακών πρακτικών και ρυθμίσεων των διαχειριστών των ιστοσελίδων που πέφτουν θύματα αυτής της τεχνικής. Με την πάροδο του χρόνου, ο όρος “dork” χρησιμοποιείται ως η συντόμευση για ένα ερώτημα αναζήτησης που εντόπιζε τέτοιες ευαίσθητες πληροφορίες [28].

Μετά από περίπου δέκα χρόνια δουλειάς, τον Νοέμβριο του 2010, ο Long παρέδωσε τη βάση δεδομένων Google Hacking Database στην Offensive Security, την εταιρεία που ανέπτυξε το Kali Linux, το Metasploit και πολλά άλλα γνωστά εργαλεία κυβερνοασφάλειας. Η βάση αυτή πλέον συντηρείται και επεκτείνεται ως μια επέκταση της Exploit Database, η οποία είναι μια βάση δεδομένων με την πληρέστερη συλλογή δημοσίων ευπαθειών και αντίστοιχου ευάλωτου λογισμικού. Μέχρι το 2022, η βάση δεδομένων αυτή περιέχει περισσότερα από 7300 “dorks”.

5.2.1 pagodo

Το pagodo (PAssive GOogle DOrk) είναι ένα εργαλείο, ανοιχτού κώδικα, το οποίο αντικαθιστά τη χειροκίνητη υποβολή των ερωτημάτων “Google dork” μέσω ενός προγράμματος περιήγησης ιστού. Συγκεκριμένα αποτελείται από δύο μέρη. Το πρώτο μέρος είναι ένα πρόγραμμα το οποίο εκτελείται μια φορά κατά την εγκατάσταση του pagodo στο υπολογιστικό σύστημα. Πρόκειται για ένα πρόγραμμα το οποίο αποθηκεύει όλες τις εγγραφές της βάσης δεδομένων Google Hacking Database τοπικά σε ένα απλό αρχείο κειμένου του οποίου η κάθε γραμμή περιέχει και ένα διαφορετικό “dork”. Το δεύτερο μέρος είναι το πρόγραμμα που κάνει την υποβολή. Λαμβάνει ως είσοδο τη λίστα των “dorks” που συγκεντρώθηκε μετά την εκτέλεση του πρώτου προγράμματος και συγκεντρώνει τα αποτελέσματα που λαμβάνει για το καθένα σε ένα αρχείο.

Για να συνδυαστεί το pagodo με τη λειτουργία του Sn1per, προστέθηκε η εντολή εκτέλεσής του στον τύπο λειτουργίας osint. Το pagodo μπορεί να λάβει αρκετές παραμέτρους που προσαρμόζουν τον τρόπο εκτέλεσής του. Η σημαντικότερη είναι η παράμετρος “-d” που υπαγορεύει την ηλεκτρονική διεύθυνση IP ή το όνομα του προστατευόμενου υπολογιστικού συστήματος. Αυτό επιτυγχάνεται μέσω του ειδικού τελεστή αναζήτησης του Google: “site:example.com”. Με τον τρόπο αυτό τα αποτελέσματα μπορούν να περιοριστούν μόνο σε αυτά που αφορούν το υπολογιστικό σύστημα που έχει θέσει ο χρήστης κατά τη δημιουργία της διαδικασίας σάρωσης. Μια άλλη παράμετρος που χρησιμοποιεί το Sn1per για την καθοδήγηση της εκτέλεσης του pagodo είναι η “-m”. Επειδή με κάθε υποβολή “dork” στο Google μπορεί να επιστραφεί ένα πολύ μεγάλο πλήθος αποτελεσμάτων, είναι απαραίτητος ο προσδιορισμός του αριθμού των αποτελεσμάτων που είναι επιθυμητό να αποθηκευτούν. Η παράμετρος “-m” προσδιορίζει το πλήθος των αποτελεσμάτων από κάθε “dork” που θα αποθηκευτούν στο τελικό αρχείο. Συγκεκριμένα, έχει επιλεχθεί να αποθηκεύονται τα πρώτα δύο αποτελέσματα, ως τα πιο σχετικά με την αναζήτηση που εκτελέστηκε.

Αξίζει να σημειωθεί ότι η υποβολή όλων των 7300 dorks δεν είναι δυνατή μέσα σε ένα σύντομο χρονικό διάστημα. Αυτό δεν έχει να κάνει με την απαιτούμενη υπολογιστική ισχύ, αλλά με την πρακτική που εφαρμόζει το Google. Το Google γνωρίζοντας την ύπαρξη προγραμμάτων που υποβάλλουν τέτοια ερωτήματα (dorks), σε περίπτωση που εντοπίσει την υποβολή μεγάλου αριθμού ερωτημάτων από την ίδια διεύθυνση και σε μικρό χρονικό διάστημα τότε μπλοκάρει τη διεύθυνση αυτή προσωρινά. Για την αποφυγή αυτής της ανεπιθύμητης καθυστέρησης μπορούν να επιστρατευτούν δύο τεχνικές.

Η μια είναι πιο απλή αλλά λιγότερο αποδοτική. Με την προτεινόμενη τεχνική, μεσολαβεί ένας τυχαίος χρόνος αναμονής ανάμεσα σε κάθε υποβολή ερωτήματος. Για τον σκοπό αυτό χρησιμοποιούνται οι παράμετροι “-i” και “-x”, που υπαγορεύουν το ελάχιστο και το μέγιστο χρονικό διάστημα αναμονής σε δευτερόλεπτα, αντίστοιχα. Το πρόγραμμα υποβάλλει τα ερωτήματα ανά χρονικά διαστήματα που ορίζονται τυχαία κάθε φορά και είναι ανάμεσα στον ελάχιστο και μέγιστο χρόνο αναμονής που έχουν οριστεί με τις τιμές των παραμέτρων “-i” και “-x”. Στη συγκεκριμένη εργασία, έχουν επιλεχθεί οι τιμές 30 και 120.

Η άλλη τεχνική είναι η αλλαγή της ηλεκτρονικής διεύθυνσης του συστήματος που υποβάλει τα ερωτήματα (dorks). Το pagodo υποστηρίζει την υποβολή ερωτημάτων είτε μέσω διακομιστή μεσολάβησης (proxy server) με τη χρήση της παραμέτρου '-p', είτε μέσω proxychain.

Ο διακομιστής μεσολάβησης μπορεί να είναι ένα υπολογιστικό σύστημα ή ένας δρομολογητής (gateway) που παρέχει μια πύλη μεταξύ του χρήστη και του διαδικτύου. Καθώς μεσολαβεί ανάμεσα στον τελικό χρήστη και τις ιστοσελίδες που επισκέπτεται στο διαδίκτυο, ο διακομιστής μεσολάβησης συμβάλλει στην αποτροπή της εισόδου επιτιθέμενων από τον κυβερνοχώρο σε ένα ιδιωτικό δίκτυο. Ένας διακομιστής μεσολάβησης μπορεί να ρυθμιστεί να εκτελεί τη λειτουργία τείχους προστασίας (firewall) ή φίλτρου. Στην πιο απλή του μορφή, όμως, ένας διακομιστής μεσολάβησης προστατεύει τα προσωπικά δεδομένα και την ιδιωτική ζωή του χρήστη. Μόνο η ηλεκτρονική διεύθυνση IP του διακομιστή μεσολάβησης γίνεται αντιληπτή από κάποιον hacker και από τις ιστοσελίδες που επισκέπτονται. Με την παρουσία του διακομιστή μεσολάβησης, τα αιτήματα δικτύου πηγαίνουν σε αυτόν, ο οποίος στη συνέχεια τα προωθεί προς το διαδίκτυο, χρησιμοποιώντας τη δική του ηλεκτρονική διεύθυνση. Εάν ο διακομιστής διαθέτει δυνατότητες κρυπτογράφησης, οι κωδικοί πρόσβασης και άλλα προσωπικά δεδομένα αποκτούν μια επιπλέον βαθμίδα προστασίας [29].

Το proxychain είναι ένα εργαλείο, ανοικτού κώδικα, που εφαρμόζεται στο λειτουργικό σύστημα Linux. Το εργαλείο αυτό αναγκάζει κάθε σύνδεση στο διαδίκτυο να γίνεται μέσω κάποιου διακομιστή μεσολάβησης. Ένα χαρακτηριστικό του proxychains είναι η δυνατότητα του να χρησιμοποιεί «αλυσιδωτά» (chain=αλυσίδα) πολλαπλούς διακομιστές μεσολάβησης. Επιπλέον, δίνει τη δυνατότητα σε προγράμματα που δεν υποστηρίζουν μια αυτόνομη χρήση διακομιστή μεσολάβησης, να εκτελούνται μέσω αυτού του εργαλείου και να εκμεταλλευτούν τις ιδιότητες ενός τέτοιου διακομιστή. Το proxychain χρησιμοποιεί ένα αρχείο διαμόρφωσης, μέσα στο οποίο ο χρήστης δηλώνει τους διακομιστές τους οποίους επιθυμεί να χρησιμοποιηθούν. Έτσι δεν υπάρχει η ανάγκη κάθε φορά που εκτελείται να δηλώνονται εκ νέου, κάνοντάς το αρκετά χρηστικό [30].

5.2.2 Ενσωμάτωση στο Sn1per της διεργασίας υποβολής ερωτημάτων Google dorks

Η δυνατότητα για την υποβολή ερωτημάτων Google dorks έχει επίσης ενσωματωθεί στο Sn1per. Συγκεκριμένα, έχει προστεθεί η παράμετρος PAGODO_PROXYCHAIN στα αρχεία διαμόρφωσης του Sn1per. Αν η παράμετρος αυτή έχει την τιμή 0, σημαίνει ότι για την υποβολή των ερωτημάτων θα μεσολαβεί ένας τυχαίος χρόνος αναμονής ανάμεσα σε κάθε υποβολή και δεν θα χρησιμοποιηθεί κάποιος διακομιστής μεσολάβησης ή proxychain. Αν η παράμετρος PAGODO_PROXYCHAIN πάρει την τιμή 1, σημαίνει ότι το Sn1per θα εκμεταλλευτεί τις δυνατότητες που παρέχει το εργαλείο proxychains και η υποβολή των ερωτημάτων θα γίνεται μέσω πολλαπλών διακομιστών μεσολάβησης. Υπάρχει και η δυνατότητα στην παράμετρο PAGODO_PROXYCHAIN να οριστούν οι ηλεκτρονικές διευθύνσεις IP των διακομιστών μεσολάβησης που το Sn1per έχει πρόσβαση και έτσι η υποβολή των ερωτημάτων να γίνει μέσω των συγκεκριμένων διακομιστών.

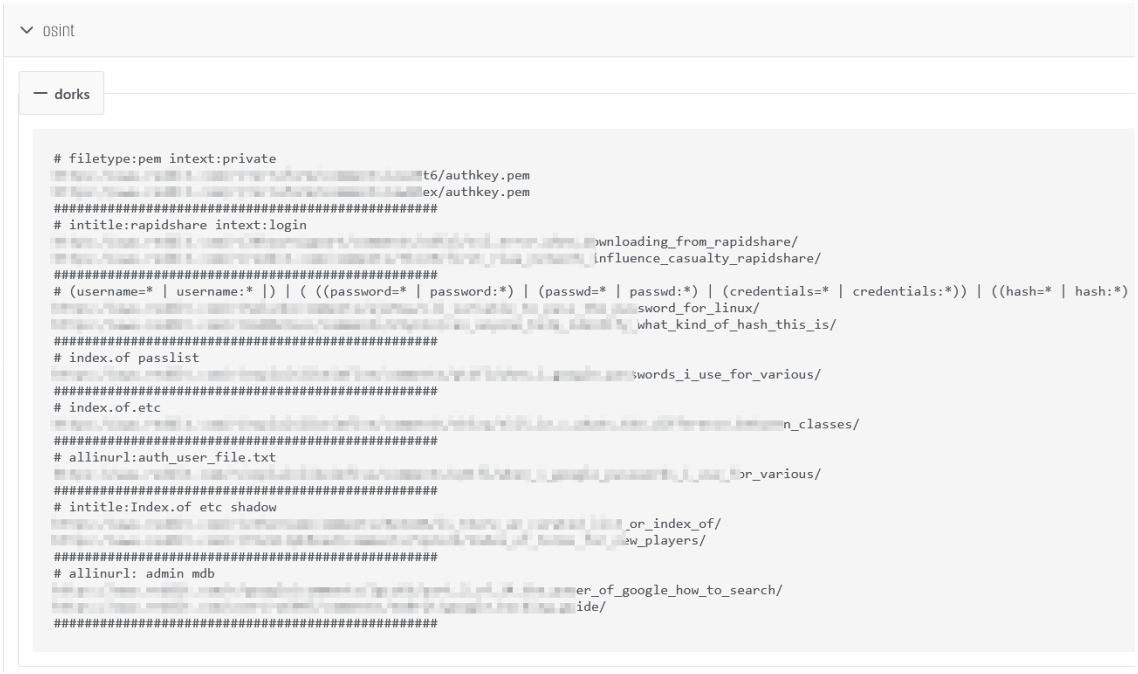
Τέλος, στα αρχεία διαμόρφωσης του Sn1per έχει προστεθεί και η παράμετρος PAGODO, η οποία μπορεί να πάρει τις τιμές 1 ή 0, που ρυθμίζει την εκτέλεση ή όχι του pagodo, αντίστοιχα.

Στην Εικόνα 5.4 φαίνεται το αρχείο εντολών που αφορά τον τρόπο εφαρμογής osint με τις αλλαγές που έχουν γίνει για μπορεί να εκτελείται το pagodo. Είναι σημαντικό να αναφερθεί ότι το pagodo, όπως δηλώνει και το όνομά του passive google dorks, είναι ένα παθητικό εργαλείο. Αυτό σημαίνει ότι η εκτέλεσή του μπορεί να διαρκέσει πολλές ώρες ακόμα και μέρες. Για τον λόγο αυτό δεν θα ήταν λογικό η διαδικασία σάρωσης του Sn1per να περιμένει την ολοκλήρωση του pagodo. Έτσι, έχει τοποθετηθεί ο τελεστής "&" στο τέλος της εντολής εκτέλεσης του pagodo, προκειμένου αυτό να εκτελείται στο παρασκήνιο, επιτρέποντας τις επόμενες εντολές να συνεχίσουν κανονικά.

```
    if [[ "$PAGODO" == "1" ]]; then
        echo -e "${OKGREEN}================================================================================${RESET}•x${OKGREEN}[`date +"%Y-%m-%d](%H:%M)"`${RESET}x•"
        echo -e "$OKRED STARTING PAGODO PASSIVE SCANNER $RESET"
        echo -e "${OKGREEN}================================================================================${RESET}•x${OKGREEN}[`date +"%Y-%m-%d](%H:%M)"`${RESET}x•"
        cd $INSTALL_DIR/plugins/pagodo
        if [[ "$PAGODO_PROXYCHAIN" == "1" ]]; then
            proxychains4 python3 pagodo.py -d $TARGET -g ./dorks/all_google_dorks.txt -i 30 -x 120 -m 2 -s $LOOT_DIR/osint/dorks-$TARGET.txt > /dev/null &
        elif [[ "$PAGODO_PROXYCHAIN" == "0" ]]; then
            python3 pagodo.py -d $TARGET -g ./dorks/all_google_dorks.txt -i 30 -x 120 -m 2 -s $LOOT_DIR/osint/dorks-$TARGET.txt > /dev/null &
        else
            python3 pagodo.py -d $TARGET -g ./dorks/all_google_dorks.txt -i 30 -x 120 -m 2 -s $LOOT_DIR/osint/dorks-$TARGET.txt -p $PAGODO_PROXYCHAIN > /dev/null &
        fi
    fi
```

Εικόνα 5.4: Αρχείο εντολών τρόπου εφαρμογής osint για την εκτέλεση του pagodo

Στην Εικόνα 5.5 φαίνεται ένα παράδειγμα των αποτελεσμάτων της εκτέλεσης του προγράμματος pagodo, όπως παρουσιάζεται μέσα από το γραφικό περιβάλλον που δημιουργήθηκε για το Sn1per. Η συγκεκριμένη διαδικασία σάρωσης από το Sn1per είχε ως στόχο να ελέγξει μια ευρέως διαδεδομένη ιστοσελίδα συγκέντρωσης ειδήσεων, περιεχομένου και συζήτησης, που για λόγους προστασίας των προσωπικών δεδομένων δεν αναφέρεται. Τα αποτελέσματα παρουσιάζονται σε μορφή λίστας. Κάθε ενότητα της λίστας αυτής περιέχει ένα συγκεκριμένο ερώτημα dork που υποβλήθηκε στο Google, ακολουθούμενο από τα πρώτα δύο πιο σχετικά αποτελέσματα της αναζήτησης. Φυσικά, σε περίπτωση που κάποιο ερώτημα δεν επέστρεψε κάποιο εύρημα τότε αυτό παραλείπεται.



Εικόνα 5.5: Παράδειγμα των αποτελεσμάτων από την εκτέλεση του pagodo

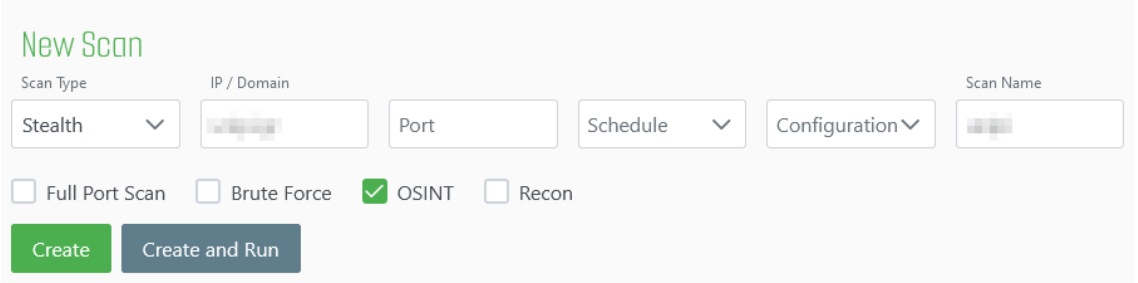
ΚΕΦΑΛΑΙΟ 6: ΠΙΛΟΤΙΚΗ ΕΦΑΡΜΟΓΗ ΤΗΣ ΝΕΑΣ ΕΚΔΟΣΗΣ ΤΟΥ SN1PER

Σε αυτό το κεφάλαιο παρουσιάζεται η πιλοτική εφαρμογή μίας διαδικασίας σάρωσης της νέας αναβαθμισμένης έκδοσης του Sn1per. Η σάρωση εφαρμόστηκε στην ιστοσελίδα ενός ενεργού μεγάλου οργανισμού. Για λόγους προστασίας των προσωπικών δεδομένων, σύμφωνα με τον Γενικό Κανονισμό Προσωπικών Δεδομένων (GDPR), ο Οργανισμός αυτός δεν κατονομάζεται, αλλά αναφέρεται ως Προστατευόμενος Οργανισμός και έχει εφαρμοστεί η διαδικασία της ανωνυμοποίησης, όπου αυτό ήταν εφικτό, π.χ. στις Εικόνες.

Κατά την παρουσίαση, ιδιαίτερη έμφαση δίνεται στις λειτουργίες που αναπτύχθηκαν και ενσωματώθηκαν στο Sn1per στα πλαίσια της διπλωματικής αυτής εργασίας (βλ. Κεφάλαιο 5) και δευτερευόντως στις ήδη υπάρχουσες λειτουργίες του Sn1per.

6.1 Δημιουργία εντολής σάρωσης

Για να ξεκινήσει η διαδικασία σάρωσης της ιστοσελίδας του Προστατευόμενου Οργανισμού θα πρέπει, αρχικά, να δημιουργηθεί η εντολή σάρωσης. Για τη δημιουργία της εντολής σάρωσης χρησιμοποιήθηκε το γραφικό περιβάλλον που αναπτύχθηκε στα πλαίσια αυτής της εργασίας και περιγράφηκε λεπτομερώς στο Κεφάλαιο 4. Συγκεκριμένα, χρησιμοποιήθηκε η φόρμα στο κάτω μέρος της κεντρικής οθόνης, όπως φαίνεται στην Εικόνα 4.2, που αφορά τη δημιουργία νέας εντολής σάρωσης. Στην Εικόνα 6.1 φαίνεται η φόρμα δημιουργίας εντολής σάρωσης όπως αυτή συμπληρώθηκε για τους σκοπούς της πιλοτικής εφαρμογής αυτού του κεφαλαίου.



Εικόνα 6.1: Συμπληρωμένη φόρμα δημιουργίας της εντολής σάρωσης που θα εκτελέσει τη διαδικασία σάρωσης της ιστοσελίδας του Προστατευόμενου Οργανισμού.

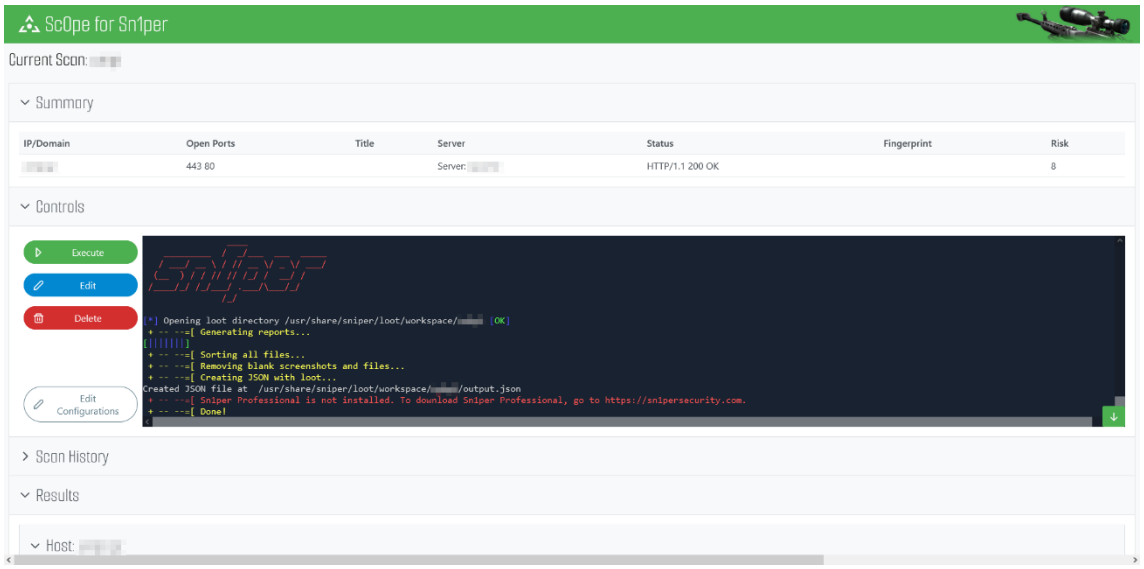
Αναλυτικά, έχουν συμπληρωθεί τα πεδία “Scan Type”, “IP/Domain” και “Scan Name” της φόρμας με τις τιμές “Stealth”, “*yyyyy.zzz*” και “*yyyyy*” αντίστοιχα, καθώς επίσης έχει επιλεχτεί η λειτουργία osint. Σημειώνεται, ότι έχει εφαρμοστεί η διαδικασία ανωνυμοποίησης και όπου “*yyyyy*“ είναι το όνομα του Προστατευόμενου Οργανισμού (domain name) και “*zzz*“ είναι η κατάληξη (top level domain π.χ .org, .com, .gr, .eu κ.λπ.). Επίσης, υπενθυμίζεται ότι το πεδίο “Scan Type” αφορά τον τύπο λειτουργίας της διαδικασίας σάρωσης (βλ. υποκεφάλαιο 3.3), το πεδίο “IP/Domain” αφορά την ηλεκτρονική διεύθυνση IP ή τη δικτυακή περιοχή που θα σαρωθεί κατά τη διαδικασία σάρωσης και το πεδίο “Scan Name” είναι η φιλική ονομασία της διαδικασίας (βλ. υποκεφάλαιο 4.1).

Είναι σημαντικό να σημειωθεί ότι έχει επιλεχθεί ο τύπος λειτουργίας “Stealth”, ο οποίος εκτελεί μια βασική σάρωση αλλά προκαλώντας όσο το δυνατόν λιγότερη δικτυακή κίνηση (βλ. υποκεφάλαιο 3.3). Επιλέχθηκε αυτός ο τύπος λειτουργίας καθώς η ιστοσελίδα του Προστατευόμενου Οργανισμού προστατεύεται από τείχος προστασίας (firewall) το οποίο αποκλείει την πρόσβαση σε ηλεκτρονικές διευθύνσεις IP οι οποίες δημιουργούν πολύ δικτυακή κίνηση μέσα σε μικρό χρονικό διάστημα. Σημειώνεται ότι δοκιμαστικά εκτελέστηκε και ο τρόπος λειτουργίας “Normal”. Ωστόσο, επειδή η εφαρμογή του προκάλεσε αρκετή δικτυακή κίνηση, το

τείχος προστασίας της συγκεκριμένης ιστοσελίδας απέκλεισε την πρόσβαση στην ηλεκτρονική διεύθυνση IP της υπολογιστικής συσκευής στην οποία εκτελέστηκε το Sn1per. Έτσι η διαδικασία σάρωσης δεν ολοκληρώθηκε και δεν επέστρεψε αποτελέσματα.

Τέλος, υπενθυμίζεται ότι οι δύο νέες λειτουργίες που ενσωματώθηκαν στο Sn1per στα πλαίσια της παρούσας διπλωματικής εργασίας, που έχουν αναλυθεί λεπτομερώς στο Κεφάλαιο 5, εκτελούνται μόνο μέσω του τρόπου εφαρμογής osint. Προκειμένου, λοιπόν, να εκτελεστούν οι δύο αυτές λειτουργίες και να παρουσιαστούν τα αποτελέσματα που αντλούνται από την ιστοσελίδα του Προστατευόμενου Οργανισμού, έχει επιλεχθεί να εκτελεστεί η λειτουργία osint κατά τη δημιουργία της εντολής εκτέλεσης.

Μετά τη δημιουργία της εντολής σάρωσης έγινε η εκτέλεση της διαδικασίας σάρωσης. Μετά το πέρας της διαδικασίας, το Sn1per αποθήκευσε τα ευρήματά του στον αντίστοιχο χώρο εργασίας (workspace) με το όνομα *yyyyy*, δηλαδή το όνομα το οποίο δόθηκε στη διαδικασία σάρωσης μέσω του πεδίου “Scan Name” της φόρμας δημιουργίας εντολής σάρωσης. Στην Εικόνα 6.2 φαίνεται ένα απόσπασμα της οθόνης αποτελεσμάτων, όπως φαίνεται μέσα από το γραφικό περιβάλλον που αναπτύχθηκε στα πλαίσια της εργασίας και έχει περιγραφεί στο υποκεφάλαιο 4.3.



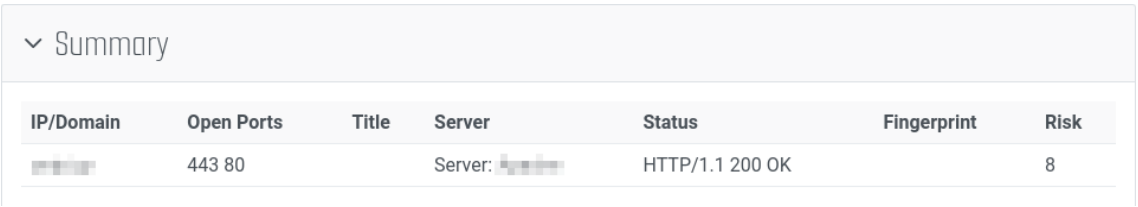
Εικόνα 6.2: Οθόνη προβολής αποτελεσμάτων γραφικού περιβάλλοντος της διαδικασίας σάρωσης του δικτυακού τόπου του Προστατευόμενου Οργανισμού

6.2 Ανάλυση αποτελεσμάτων

Στη συνέχεια θα γίνει ανάλυση των αποτελεσμάτων που άντλησε η διαδικασία σάρωσης του Sn1per. Όπως έχει περιγραφεί στο υποκεφάλαιο 4.3, η οθόνη αποτελεσμάτων αποτελείται από τέσσερα αναπτυσσόμενα τμήματα. Στην πιλοτική εφαρμογή του κεφαλαίου αυτού θα αναλυθούν το πρώτο και το τελευταίο αναπτυσσόμενο τμήμα, δηλαδή το τμήμα “Summary” (Σύνοψη) και το τμήμα “Results” (Αποτελέσματα).

Το τμήμα “Summary” παρουσιάζει έναν συνοπτικό πίνακα με τις κυριότερες πληροφορίες που μπόρεσαν να αποσπαστούν από την ιστοσελίδα του Προστατευόμενου Οργανισμού που σαρώθηκε. Λόγω του τρόπου εφαρμογής “Stealth” δεν έχουν συμπληρωθεί όλα τα κελιά του πίνακα. Συγκεκριμένα, στον πίνακα υπάρχει μόνο μία γραμμή, η οποία αφορά την περιοχή δικτύου “*yyyyy.zzz*”, όπως φαίνεται από την πρώτη στήλη του πίνακα. Στη δεύτερη στήλη

αναφέρονται οι ανοικτές θύρες δικτύου που εντοπίστηκαν στην ιστοσελίδα του Προστατευόμενου Οργανισμού, οι οποίες είναι οι θύρες 80 και 443 που αντιστοιχούν στα πρωτόκολλα δικτυακής επικοινωνίας http και https αντίστοιχα. Η επόμενη στήλη η οποία περιέχει πληροφορίες είναι η στήλη “Server”, η οποία προσφέρει πληροφορίες σχετικά με τον τύπο του διακομιστή που φιλοξενεί την ιστοσελίδα. Πρόκειται, για έναν διακομιστή, του οποίου την έκδοση δεν κατάφερε να ανιχνεύσει το Sn1per. Στην επόμενη στήλη, φαίνεται η κατάσταση της ιστοσελίδας η οποία είναι ενεργή και προσπελάσιμη. Στην τελευταία στήλη φαίνεται ο βαθμός ρίσκου στον οποίο είναι εκτεθειμένη η ιστοσελίδα. Το Sn1per υπολόγισε ότι ο βαθμός αυτός είναι οκτώ, λόγω κάποιων ευπαθειών που ανίχνευσε και παρουσιάζονται στη συνέχεια.

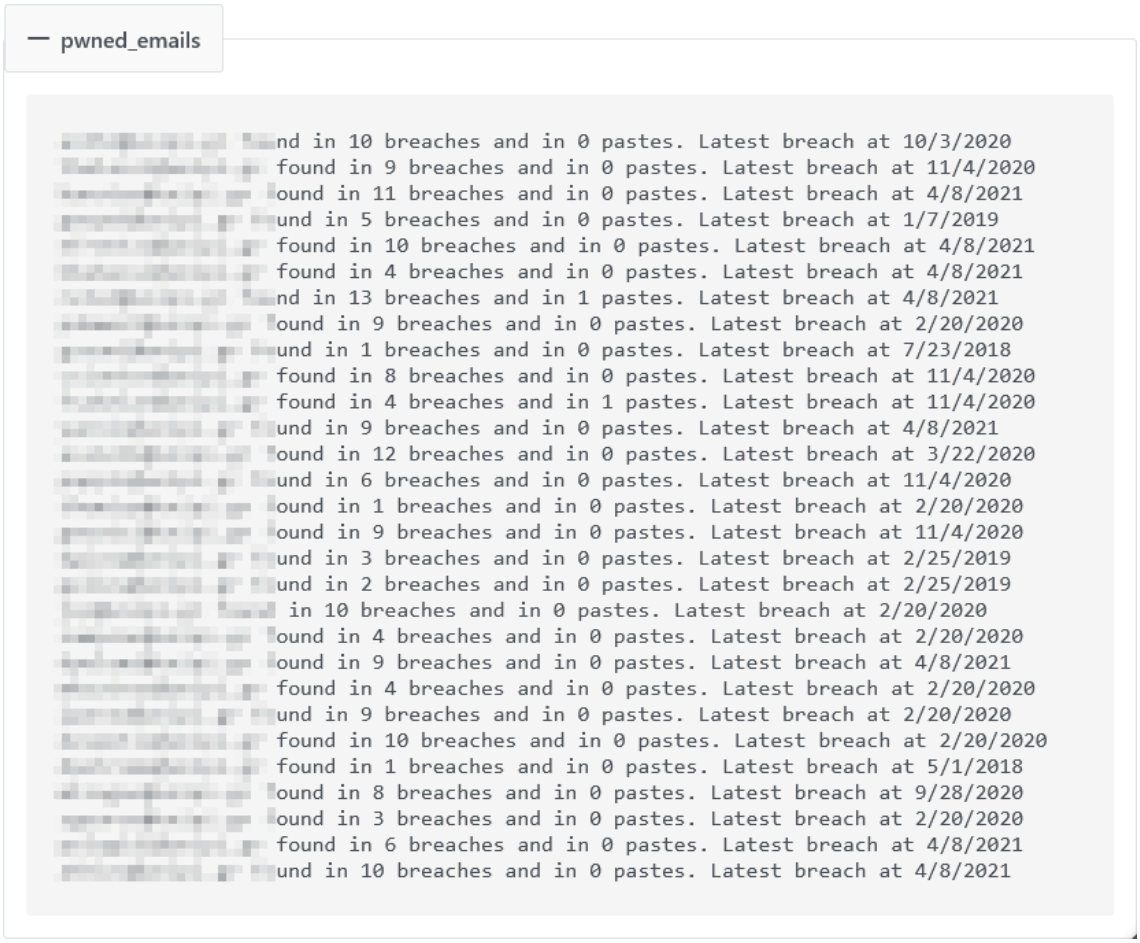


Εικόνα 6.3: Ο πίνακας του τμήματος “Summary”.

Το επόμενο τμήμα που παρουσιάζεται στα πλαίσια της πιλοτικής εφαρμογής του κεφαλαίου αυτού είναι το τμήμα “Results”. Υπενθυμίζεται, ότι στο τμήμα “Results” παρουσιάζονται όλα τα αποτελέσματα της διαδικασίας σάρωσης. Όπως αναφέρθηκε στην αρχή αυτού του κεφαλαίου θα δοθεί έμφαση στα αποτελέσματα που επιστρέφουν οι νέες λειτουργίες που ενσωματώθηκαν στο Sn1per. Αφού αναλυθούν τα αποτελέσματα των δύο νέων λειτουργιών θα αναλυθούν ορισμένα ακόμα αποτελέσματα που επιστρέφονται από ήδη υπάρχουσες λειτουργίες του Sn1per.

6.2.1 Ανάλυση αποτελεσμάτων νέων λειτουργιών

Πρώτα θα αναλυθούν τα αποτελέσματα που επέστρεψε η νέα λειτουργία που ενσωματώθηκε στο Sn1per που αφορά τη διεργασία επικοινωνίας με την ιστοσελίδα haveibeenpwned, όπως περιγράφηκε στο υποκεφάλαιο 5.1.2. Υπενθυμίζεται ότι η λειτουργία αυτή ανιχνεύει παραβιασμένους λογαριασμούς ηλεκτρονικού ταχυδρομείου. Τα αποτελέσματα αυτής της λειτουργίας βρίσκονται στο πεδίο αποτελεσμάτων “pwned_emails” του γραφικού περιβάλλοντος. Στην Εικόνα 6.4 φαίνονται οι διευθύνσεις που ανίχνευσε η νέα λειτουργία. Συγκεκριμένα ανιχνεύτηκαν 29 διευθύνσεις ηλεκτρονικού ταχυδρομείου των οποίων οι κωδικοί πρόσβασης έχουν διαρρεύσει στο παρελθόν. Για παράδειγμα στην πρώτη γραμμή των αποτελεσμάτων βρίσκεται η ανωνυμοποιημένη (για λόγους προστασίας προσωπικών δεδομένων) διεύθυνση *xxxxx@yyyyy.zzz*, η οποία βρέθηκε σε δέκα περιστατικά διαρροής κωδικών πρόσβασης εκ των οποίων η τελευταία έγινε στις 10 Μαρτίου του 2020. Σε περίπτωση που ο κάτοχος αυτού του λογαριασμού δεν έχει αλλάξει τον κωδικό πρόσβασης, μετά από αυτή την ημερομηνία, είναι πολύ πιθανό να υπάρχουν τρίτες οντότητες που γνωρίζουν τον τρέχον κωδικό πρόσβασης και μπορούν να συνδεθούν στον λογαριασμό αυτό. Αντίστοιχες πληροφορίες μπορούν να αντληθούν για κάθε μία διεύθυνση ηλεκτρονικού ταχυδρομείου που εμφανίζεται στη λίστα των αποτελεσμάτων αυτού του πεδίου.

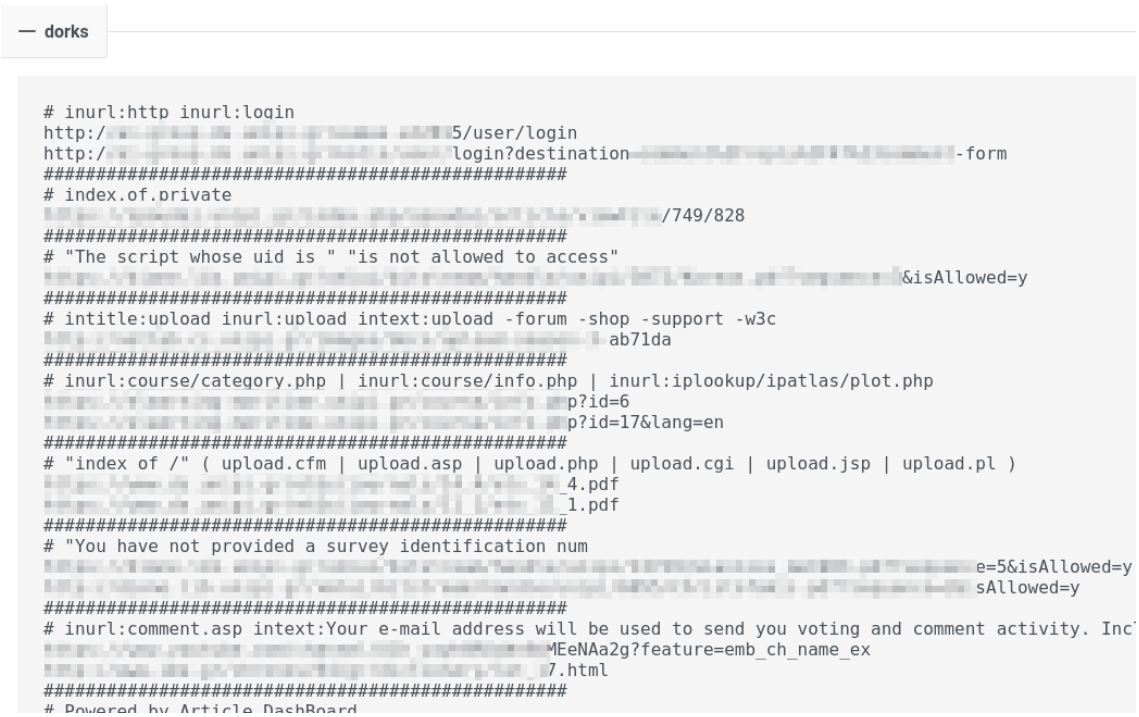


Εικόνα 6.4: Πεδίο αποτελεσμάτων της νέας λειτουργίας που ενσωματώθηκε στο Sn1per επικοινωνίας με την ιστοσελίδα haveibeenpwned

Στη συνέχεια θα αναλυθούν τα αποτελέσματα που επέστρεψε η δεύτερη νέα λειτουργία που ενσωματώθηκε στο Sn1per που αφορά τη διεργασία υποβολής εξειδικευμένων ερωτημάτων “dorks”, στη μηχανή αναζήτησης Google, όπως περιγράφηκε στο υποκεφάλαιο 5.2.2. Υπενθυμίζεται ότι η λειτουργία αυτή απαιτεί πολλές ώρες για να ολοκληρωθεί. Στα πλαίσια της πιλοτικής εφαρμογής αυτού του κεφαλαίου παρουσιάζονται τα αποτελέσματα που ανίχνευσε η νέα λειτουργία μετά από είκοσι ώρες εκτέλεσης. Στην Εικόνα 6.5 φαίνεται ένα απόσπασμα από τα αποτελέσματα της λειτουργίας μέσα από το γραφικό περιβάλλον. Υπενθυμίζεται ότι τα αποτελέσματα αποθηκεύονται σε μορφή λίστας της οποίας οι καταχωρήσεις αναφέρονται στο κάθε ειδικό ερώτημα “dork” κάτω από το οποίο εμφανίζονται οι ηλεκτρονικές διευθύνσεις των αποτελεσμάτων που ανιχνεύθηκαν από το Google.

Ενδεικτικά, το πρώτο ερώτημα “dork” αφορά τον εντοπισμό υποσελίδων στην ιστοσελίδα του Προστατευόμενου Οργανισμού, οι οποίες να περιέχουν φόρμα για σύνδεση σε κάποια υπηρεσία και να χρησιμοποιούν το μη κρυπτογραφημένο πρωτόκολλο επικοινωνίας http. Η ύπαρξη τέτοιων υποσελίδων αποτελούν ένα μεγάλο κενό ασφαλείας, καθώς είναι πολύ εύκολο από κακόβουλες τρίτες οντότητες να υποκλέψουν τα διαπιστευτήρια συνόδου των χρηστών που επιχειρούν να συνδεθούν. Επομένως, μπορεί εύκολα ο χειριστής του Sn1per να ενημερωθεί για την ύπαρξη τέτοιων προβληματικών υποσελίδων της ιστοσελίδας και να προβεί στις ανάλογες ενέργειες για την ασφάλισή τους. Αντίστοιχα, μελετώντας τα υπόλοιπα ευρήματα των ειδικών ερωτημάτων “dorks”, ο χειριστής μπορεί να ανακαλύψει παρόμοια προβλήματα της ιστοσελίδας

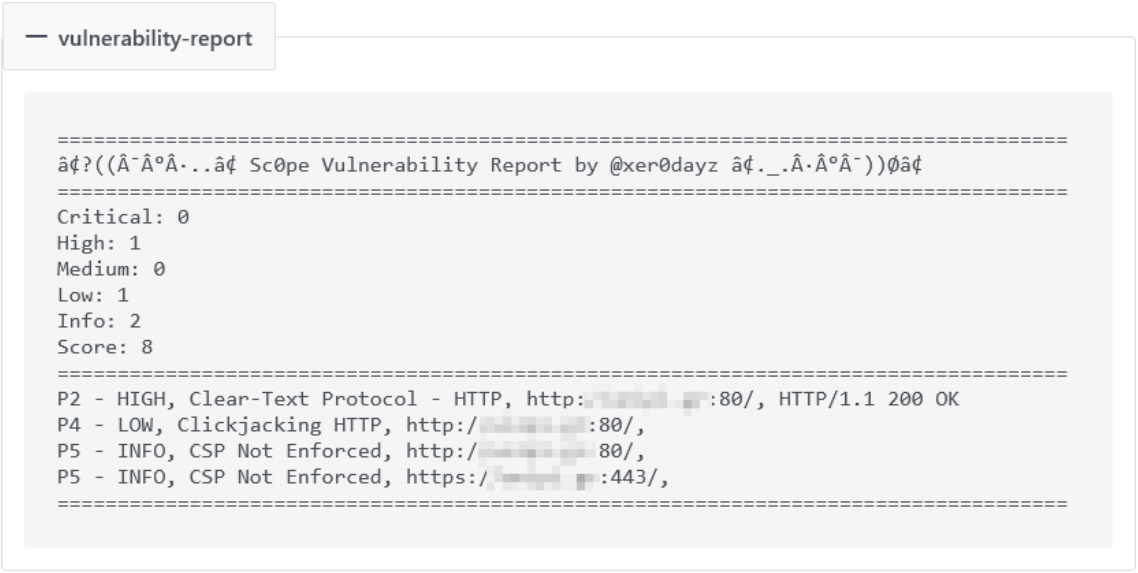
του Προστατευόμενου Οργανισμού ή αρχεία που περιέχουν ευαίσθητα δεδομένα και είναι εσφαλμένα ελεύθερα στο διαδίκτυο και να προβεί στις απαραίτητες ενέργειες.



Εικόνα 6.5: Απόσπασμα του πεδίου αποτελεσμάτων “dorks” της νέας λειτουργίας που ενσωματώθηκε στο Sn1per άντλησης πληροφοριών μέσω της τεχνικής “google hacking”

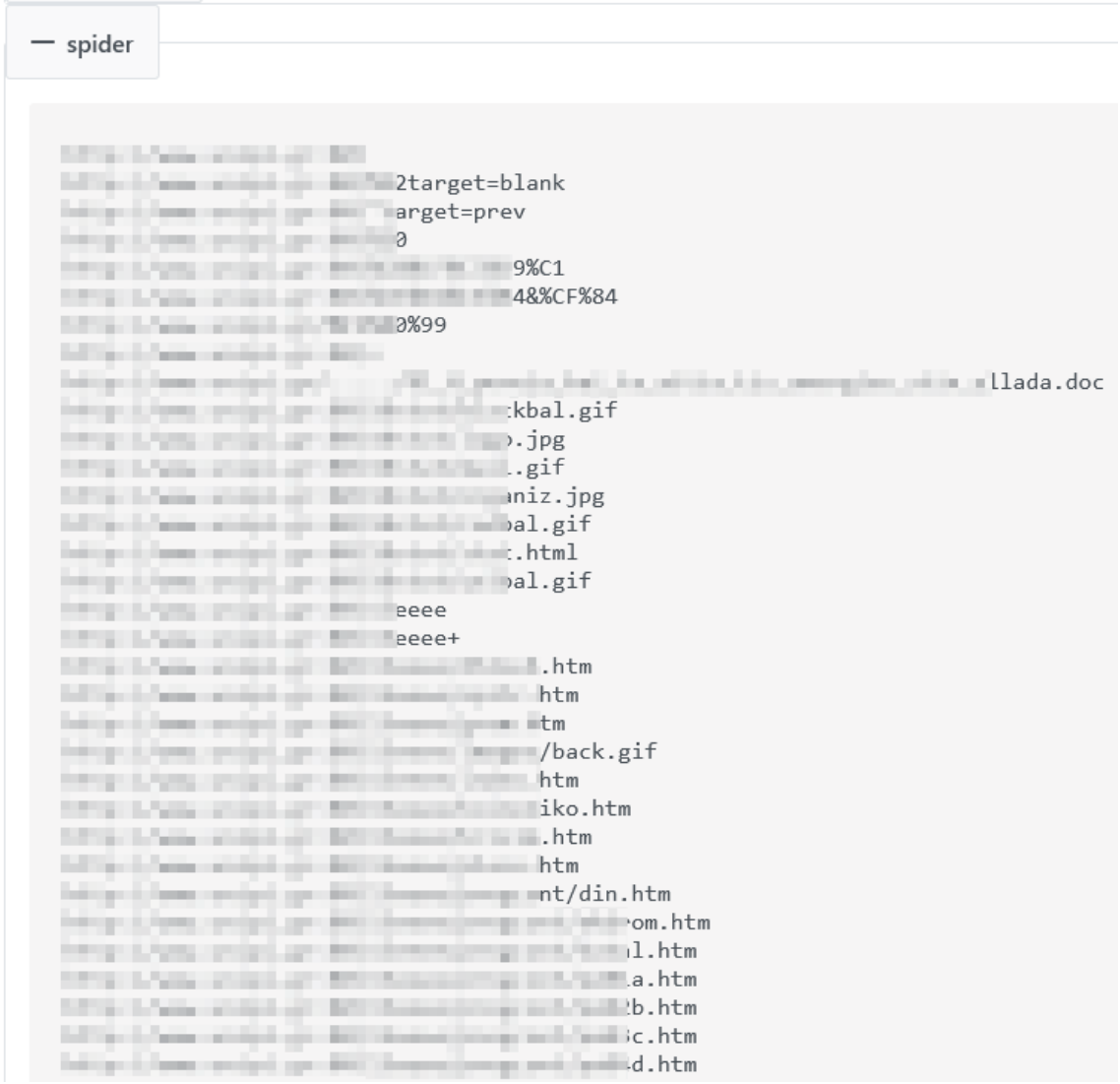
6.2.2 Ανάλυση αποτελεσμάτων ήδη υπαρχουσών λειτουργιών

Ένα από τα πρώτα πεδία που εμφανίζονται στο τμήμα των αποτελεσμάτων είναι το πεδίο “vulnerability-report” (αναφορά ευπαθειών). Όπως φαίνεται στην Εικόνα 6.6, στο πεδίο αυτό παρουσιάζονται όλες οι ευπάθειες που ανιχνεύθηκαν στην ιστοσελίδα του Προστατευόμενου Οργανισμού, κατηγοριοποιημένες σύμφωνα με την κρισιμότητα της κάθε μίας. Συγκεκριμένα, έχουν ανιχνευτεί τέσσερις ευπάθειες εκ των οποίων μία είναι υψηλής κρισιμότητας, μία είναι χαμηλής και δύο είναι απλά ενημερωτικές. Πιο αναλυτικά, η ευπάθεια υψηλής κρισιμότητας επισημαίνει ότι υπάρχει έκδοση της ιστοσελίδας η οποία χρησιμοποιεί πρωτόκολλο επικοινωνίας http, δηλαδή μη κρυπτογραφημένη επικοινωνία. Η επόμενη ευπάθεια που ανιχνεύθηκε είναι το λεγόμενο “Clickjacking”. Το “Clickjacking” είναι μια επίθεση που εξαπατά τον χρήστη στο να πατήσει κουμπιά της ιστοσελίδας που νομίζει ότι είναι ασφαλή ενώ στην πραγματικότητα πρόκειται για κακόβουλα κουμπιά που έχει προσθέσει κάποια κακόβουλη οντότητα. Αυτό μπορεί να συμβεί λόγω των πλαισίων (iframes) που χρησιμοποιεί η γλώσσα σήμανσης html της ιστοσελίδας, τα οποία επιτρέπουν σε μια ιστοσελίδα να φορτώνει άλλες ιστοσελίδες εσωτερικά της. Για να προστατευτεί μια ιστοσελίδα από τέτοιου είδους επιθέσεις μπορεί να ορίσει την αναγκαστική χρήση της πολιτικής “CSP” (Content Security Policy). Η πολιτική αυτή υπαγορεύει σε ποιες περιπτώσεις επιτρέπεται η χρήση των πλαισίων (iframes) της html. Έτσι, το Sn1per ελέγχει αν έχει ενεργοποιηθεί η επιβολή της χρήσης της πολιτικής “CSP” και όπως φαίνεται στο συγκεκριμένο παράδειγμα, ενημερώνει ότι αυτή δεν είναι ενεργή.



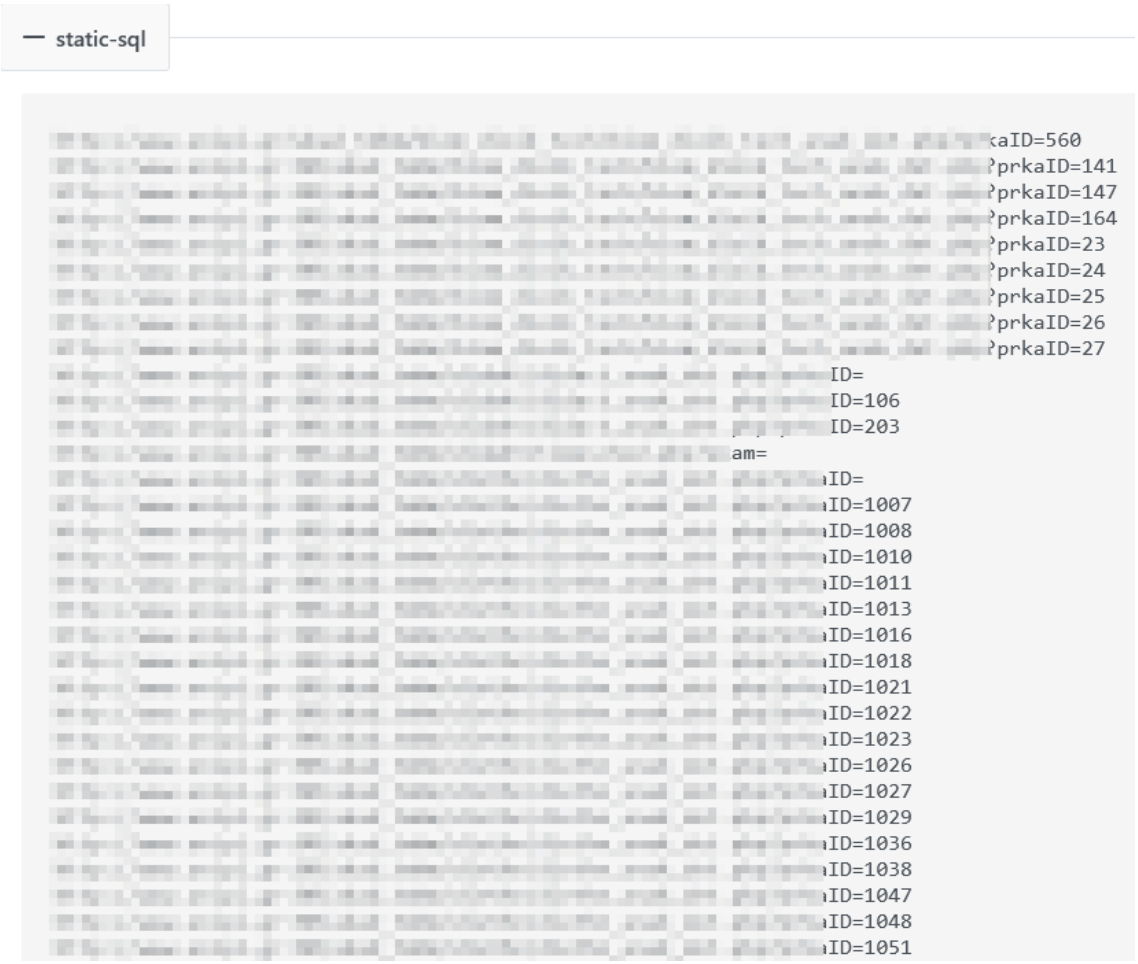
Εικόνα 6.6: Πεδίο αναφοράς ευπαθειών όπως φαίνεται μέσα από το γραφικό περιβάλλον που αναπτύχθηκε

Τέλος, στα πλαίσια της πιλοτικής εφαρμογής αυτού του κεφαλαίου θα αναλυθεί ένα μέρος του τμήματος web. Συγκεκριμένα, θα αναλυθούν τα πεδία του τμήματος web που αφορούν την ανίχνευση Ενιαίων Εντοπιστών Πόρων (Uniform Resource Locator, URL) της ιστοσελίδας του Προστατευόμενου Οργανισμού. Αυτή είναι μια διαδικασία η οποία γίνεται με σκοπό την ανίχνευση ευάλωτων σε διάφορες επιθέσεις URLs, όπως RCE (Remote Code Execution), XSS (Cross Site Scripting) και SQL Injection. Αρχικά, θα πρέπει να συγκεντρωθούν όσο το δυνατόν περισσότεροι τέτοιοι URL. Ο τρόπος με τον οποίον, το Sn1per, συγκεντρώνει τέτοιους URLs είναι μέσω επικοινωνίας με άλλες ιστοσελίδες οι οποίες διαθέτουν έτοιμες λίστες με URLs από διάφορες ιστοσελίδες του διαδικτύου. Τέτοιες ιστοσελίδες είναι η web.archive.org ή η hackertarget.com. Το Sn1per επικοινωνεί με αυτές τις ιστοσελίδες και συγκεντρώνει λίστες με URLs τις οποίες ενώνει σε μια μεγάλη λίστα την οποία αποθηκεύει σε ένα αρχείο με όνομα “spider”. Ένα τμήμα του αρχείου αυτού, όπως φαίνεται στο αντίστοιχο πεδίου του γραφικού περιβάλλοντος, παρουσιάζεται στην Εικόνα 6.7.



Εικόνα 6.7: Τμήμα του πεδίου spider όπως φαίνεται μέσα από το γραφικό περιβάλλον που αναπτύχθηκε

Έχοντας δημιουργήσει αυτή τη συγκεντρωτική λίστα εκτελεί μια στατική ανάλυση του κάθε URL, ώστε να εντοπίσει αυτούς που πιθανόν είναι ευάλωτοι σε συγκεκριμένες επιθέσεις. Το Sn1per, λοιπόν, σε κάθε ένα URL της λίστας αναζητά για λέξεις κλειδιά ή για κάποιο συγκεκριμένο μοτίβο ώστε να αποφανθεί εάν είναι πιθανό να είναι ευάλωτος ο URL σε κάποιου είδους επίθεση. Στην Εικόνα 6.8, φαίνεται ένα μέρος του πεδίου static-sql, το οποίο περιέχει μια λίστα με τους URLs που έκρινε το Sn1per ως ευάλωτους σε επιθέσεις τύπου SQL Injection.



Εικόνα 6.8: Τμήμα του πεδίου static-sql, όπως φαίνεται μέσα από το γραφικό περιβάλλον που αναπτύχθηκε

ΚΕΦΑΛΑΙΟ 7: ΣΥΜΠΕΡΑΣΜΑΤΑ

Με τις Τεχνολογίες Πληροφορικής και Επικοινωνιών (ΤΠΕ) να εξελίσσονται με ραγδαίους ρυθμούς και με επιχειρήσεις και οργανισμούς να υιοθετούν ολοένα και περισσότερες ψηφιακές υπηρεσίες, η ανησυχία για την ασφάλεια των δεδομένων που βρίσκονται στον κυβερνοχώρο μεγαλώνει διαρκώς. Παρότι υπάρχουν πολλές πρακτικές που μπορούν να ακολουθηθούν και πολλά λογισμικά που μπορούν να βοηθήσουν στην αποφυγή κενών ασφαλείας, σε σχεδόν καθημερινή βάση παρατηρούνται διαρροές δεδομένων ακόμα και από μεγάλους οργανισμούς.

Στη διαρκή προσπάθεια αποφυγής διαρροών δεδομένων συμβάλει η παρακολούθηση της εξωτερικής επιφάνειας. Η παρακολούθηση, δηλαδή, του συνόλου όλων των πιθανών σημείων, όπου ένας μη εξουσιοδοτημένος χρήστης μπορεί να αποκτήσει παράνομη πρόσβαση σε ένα υπολογιστικό σύστημα. Για την παρακολούθηση της εξωτερικής επιφάνειας έχουν αναπτυχθεί πολλά λογισμικά, κλειστού και ανοικτού κώδικα. Πολλά από τα λογισμικά αυτά απαιτούν την καταβολή σημαντικών χρηματικών ποσών για την αγορά είτε άδειας χρήσης είτε κάποιας τακτικής συνδρομής. Ωστόσο υπάρχουν και λογισμικά ανοικτού κώδικα που διανέμονται ελεύθερα, αλλά αυτά είναι αρκετά λιγότερα. Ένα τέτοιο λογισμικό, ανοικτού κώδικα, είναι το Sn1per.

Το Sn1per είναι ένα λογισμικό παρακολούθησης εξωτερικής επιφάνειας και είναι διεθνώς το πιο εξελιγμένο και ολοκληρωμένο λογισμικό ανοικτού κώδικα. Έχει τη δυνατότητα να σαρώσει ένα δίκτυο ανιχνεύοντας όλες τις ενεργές υπολογιστικές συσκευές που περιέχει και για κάθε μια από αυτές να ελέγξει εάν έχει κάποιο γνωστό κενό ασφαλείας το οποίο θα μπορούσε να επιτρέψει σε κάποιον μη εξουσιοδοτημένο χρήστη να αποκτήσει παράνομη πρόσβαση. Επιπλέον, έχει τη δυνατότητα να εκτελέσει σαρώσεις για πληροφορίες ανοικτής πηγής (osint) για κάποια δικτυακή περιοχή (domain) που θα του ορίζει ο χειριστής. Αυτό επιτυγχάνεται ενορχηστρώνοντας πολλά άλλα, μικρότερα σε έκταση, λογισμικά ανοικτού κώδικα, καθένα από τα οποία είναι υπεύθυνο για μία συγκεκριμένη διεργασία της διαδικασίας σάρωσης. Διανέμεται σε δύο εκδόσεις, την ελεύθερη έκδοση ανοικτού κώδικα που εκτελείται μόνο μέσω της γραμμής εντολών και την επί πληρωμή επαγγελματική έκδοση κλειστού κώδικα, η οποία διαθέτει ένα φιλικό γραφικό περιβάλλον.

Στα πλαίσια αυτής της διπλωματικής εργασίας χρησιμοποιήθηκε η ελεύθερη έκδοση ανοικτού κώδικα του λογισμικού Sn1per, που προσφέρεται για την ανάπτυξη νέων και τη βελτίωση υπαρχουσών λειτουργιών. Συγκεκριμένα, οι βασικότερες νέες λειτουργίες που ενσωματώθηκαν στο Sn1per είναι τρεις. Η πρώτη είναι η δημιουργία ενός γραφικού περιβάλλοντος, για τη φιλικότερη αλληλεπίδραση με τον χειριστή του. Η ανάπτυξη του γραφικού περιβάλλοντος είχε ως πρότυπο το γραφικό περιβάλλον που διαθέτει η επί πληρωμή επαγγελματική έκδοση. Η δεύτερη νέα λειτουργία που ενσωματώθηκε είναι η επικοινωνία με την ιστοσελίδα haveibeenpwned, ώστε να ανιχνεύονται παραβιάσεις διευθύνσεων ηλεκτρονικού ταχυδρομείου που μπορεί να έχουν καταγραφεί στο παρελθόν. Τέλος, ενσωματώθηκε η χρήση ειδικών ερωτημάτων “dorks” στη μηχανή αναζήτησης Google, ώστε να ανιχνευθούν πιθανόν εσφαλμένα εκτεθειμένα αρχεία ή πληροφορίες του οργανισμού στο διαδίκτυο.

Μέσω της πιλοτικής εφαρμογής, που περιγράφηκε στο Κεφάλαιο 6, παρουσιάστηκε η χρησιμότητα των νέων λειτουργιών που ενσωματώθηκαν στο Sn1per και τον πλούτο πληροφοριών που μπορούν να ανιχνευτούν και να ενημερώσουν τον χειριστή του. Συγκεκριμένα, στο Κεφάλαιο 6 σαρώθηκε η ιστοσελίδα του Προστατευόμενου Οργανισμού μέσω του αναβαθμισμένου λογισμικού Sn1per, με τη χρήση των νέων λειτουργιών που ενσωματώθηκαν σε αυτό. Τα αποτελέσματα της διαδικασίας σάρωσης που εκτελέστηκε αποκάλυψαν ότι υπάρχουν μερικά κενά ασφαλείας στην ιστοσελίδα του Προστατευόμενου Οργανισμού, οπότε έτσι υποδεικνύεται ποια από αυτά απαιτούν ειδική μεταχείριση προκειμένου να διορθωθούν.

Η επικοινωνία με την ιστοσελίδα haveibeenpwned, η νέα διαδικασία που ενσωματώθηκε στο Sn1per, αποδείχθηκε ιδιαίτερα χρήσιμη, κατά την πιλοτική του εφαρμογή. Συγκεκριμένα, το Sn1per μέσω του ελέγχου πληροφοριών ανοιχτής πηγής (osint), που εκτέλεσε κατά τη διαδικασία σάρωσης, εντόπισε κάποιες εκτεθειμένες διευθύνσεις ηλεκτρονικού ταχυδρομείου του Προστατευόμενου Οργανισμού. Συγκεκριμένα, με τη νέα λειτουργία, εντοπίστηκαν ότι οι 29 από αυτές τις διευθύνσεις έχουν παραβιαστεί στο παρελθόν. Μέσω του γραφικού περιβάλλοντος, που

αναπτύχθηκε, ο χειριστής του Sn1per ενημερώνεται με έναν λειτουργικό τρόπο σχετικά με την ημερομηνία της πιο πρόσφατης παραβίασης για κάθε μία από τις παραβιασμένες διευθύνσεις.

Η δεύτερη λειτουργία που ενσωματώθηκε στο Sn1per, η διεργασία υποβολής ερωτημάτων Google dorks, αποσκοπεί στην άντληση ευαίσθητων πληροφοριών του προστατευόμενου υπολογιστικού συστήματος, που μπορεί να κυκλοφορούν εσφαλμένα ελεύθερες στο διαδίκτυο. Κατά την πιλοτική εφαρμογή, η νέα λειτουργία αυτή ανίχνευσε κάποιες χρήσιμες πληροφορίες. Η σημαντικότερη από αυτές είναι ότι εντόπισε υποσελίδες της ιστοσελίδας του Προστατευόμενου Οργανισμού, στις οποίες περιέχεται φόρμα για σύνδεση σε κάποια υπηρεσία και χρησιμοποιούν μη κρυπτογραφημένο πρωτόκολλο επικοινωνίας. Μέσω του γραφικού περιβάλλοντος, που αναπτύχθηκε, ο χειριστής του Sn1per ενημερώνεται με έναν λειτουργικό τρόπο σχετικά με τις ηλεκτρονικές διευθύνσεις των αποτελεσμάτων που ανίχνευσε το ειδικό ερώτημα “dork”.

Τέλος, μέσα από το γραφικό περιβάλλον που αναπτύχθηκε, η χρήση του Sn1per έχει γίνει πολύ πιο λειτουργική, σε σύγκριση με τη χρήση του μέσα από την γραμμή εντολών. Ο χειριστής μπορεί εύκολα να δημιουργήσει και να εκκινήσει διαδικασίες σάρωσης χωρίς την ανάγκη να γνωρίζει όλες τις παραμέτρους που δέχεται το Sn1per. Όλες οι πιθανές τροποποιήσεις που μπορεί να χρειαστούν κατά τη δημιουργία νέας διαδικασίας σάρωσης βρίσκονται συγκεντρωμένες σε μια φιλική διεπαφή αναδυόμενου παραθύρου. Το γραφικό περιβάλλον που αναπτύχθηκε διευκολύνει, επιπλέον, την προβολή των αποτελεσμάτων μιας διαδικασίας σάρωσης. Δεν υπάρχει πλέον η ανάγκη από τον χειριστή να αναζητά ένα προς ένα τα αρχεία αποτελεσμάτων, καθώς όλα τα αποτελέσματα παρουσιάζονται συγκεντρωμένα και κατανεμημένα στις αντίστοιχες κατηγορίες. Τέλος, μέσω του γραφικού περιβάλλοντος ο χειριστής μπορεί να παρακολουθήσει σε πραγματικό χρόνο την πρόοδο των διαδικασιών σάρωσης καθώς και να επεξεργαστεί τα ειδικά αρχεία διαμόρφωσης που χρησιμοποιεί το Sn1per για την παραμετροποίηση του τρόπου με τον οποίο εκτελείται μια διαδικασία σάρωσης.

Με την προσθήκη όλων αυτών των νέων λειτουργιών, η ελεύθερη έκδοση ανοικτού κώδικα του λογισμικού Sn1per γίνεται ένα ακόμη ισχυρότερο εργαλείο για την παρακολούθηση της εξωτερικής επιφάνειας ενός οργανισμού, όπως φάνηκε από την πιλοτική εφαρμογή του που περιγράφηκε στο Κεφάλαιο 6.

Ωστόσο, παρόλο που το ανανεωμένο λογισμικό Sn1per φαίνεται ότι μπορεί να εντοπίσει όλα τα προβλήματα κυβερνοασφάλειας και παρακολούθησης εξωτερικής επιφάνειας, είναι απαραίτητη η τακτική ενημέρωσή του για την αντιμετώπιση νέων επιθέσεων που μπορεί να εμφανιστούν στον κυβερνοχώρο. Άλλωστε, ως λογισμικό ανοικτού κώδικα, επιδέχεται βελτιώσεις και αναβαθμίσεις, αυξάνοντας έτσι την αξία του ως λογισμικό παρακολούθησης εξωτερικής επιφάνειας.

Πράγματι, ο τομέας της κυβερνοασφάλειας είναι ένας δυναμικός τομέας που μεταβάλλεται συνεχώς, καθώς κακόβουλοι χρήστες ανακαλύπτουν διαρκώς νέους τρόπους να παραβιάσουν τα υπολογιστικά συστήματα οργανισμών και εταιριών. Έτσι, ένα φαινομενικά ασφαλές υπολογιστικό σύστημα είναι, συνήθως, θέμα χρόνου να αποδειχθεί ανασφαλές. Επομένως, τα λογισμικά παρακολούθησης εξωτερικής επιφάνειας πρέπει να παραμένουν ενήμερα και να μπορούν να εκτελούν σύγχρονες διαδικασίες σάρωσης που να παρέχουν πληροφορίες σχετικά με όλους τους τύπους επιθέσεων, από τους πιο παλιούς στους πιο σύγχρονους.

ΒΙΒΛΙΟΓΡΑΦΙΑ

[1] Kemmerer, R. A. (2003). Cybersecurity. 25th International Conference on Software Engineering, 2003. Proceedings

[2] Thimbleby, H. (n.d.). The computer science of everyday things. Proceedings Second Australasian User Interface Conference. AUIC 2001

[3] Cybersecurity, Critical Infrastructure. "Framework for improving critical infrastructure cybersecurity." URL: https://nvlpubs. nist. gov/nistpubs/CSWP/NIST. CSWP 4162018 (2018)

[4] Armstrong, Robert, Jackson Mayo, and Frank Siebenlist. "Complexity science challenges in cybersecurity." Sandia National Laboratories SAND Report (2009)

[5] https://www.cisco.com/c/en/us/products/security/what-is-cybersecurity.html

[6] Kritzinger, Elmarie, and Sebastiaan H. von Solms. "Cyber security for home users: A new way of protection through awareness enforcement." Computers & Security 29.8 (2010): 840-847

[7] Beyer, Richard E., and B. Brummel. "Implementing effective cyber security training for end users of computer networks." Society for Human Resource Management and Society for Industrial and Organizational Psychology (2015)

[8] KIM, Sunghwan, et al. Secure collecting, optimizing, and deploying of firewall rules in software-defined networks. IEEE Access, 2020, 8: 15166-15177.

[9] THEISEN, Christopher, et al. Attack surface definitions: A systematic literature review. Information and Software Technology, 2018, 104: 94-103.

[10] MANADHATA, Pratyusa; WING, Jeannette M. Measuring a system's attack surface. Carnegie-Mellon Univ Pittsburgh pa School of Computer Science, 2004.

[11] ASHLEY, Travis, et al. Aggregate attack surface management for network discovery of operational technology. Computers & Security, 2022, 123: 102939.

[12] KYLMAKOSKI, Roope. Efficient authoring of software documentation using RaPiD7. In: 25th International Conference on Software Engineering, 2003. Proceedings. IEEE, 2003. p. 255-261.

[13] https://www.digitalshadows.com/searchlight/

[14] ELLIS, Casey. Combining the Power of Builders and Breakers. In: Enigma 2018 (Enigma 2018). 2018.

[15] https://www.immuniweb.com/products/discovery/

[16] PERENS, Bruce, et al. The open source definition. Open sources: voices from the open source revolution, 1999, 1: 171-188.

[17] BENNETTS, Simon. Owasp zed attack proxy. AppSec USA, 2013.

[18] GUHA, Arjun; SAFTOIU, Claudiu; KRISHNAMURTHI, Shriram. The essence of JavaScript. In: European conference on Object-oriented programming. Springer, Berlin, Heidelberg, 2010. p. 126-150.

[19] https://insights.stackoverflow.com/survey/2021#section-most-popular-technologies-web-frameworks

[20] BIERMAN, Gavin; ABADI, Martín; TORGERSEN, Mads. Understanding typescript. In: European Conference on Object-Oriented Programming. Springer, Berlin, Heidelberg, 2014. p. 257-281.

[21] ROSE, M. E. The analysis of angular correlation and angular distribution data. Physical Review, 1953, 91.3: 610.

[22] SATHEESH, Mithun; D'MELLO, Bruno Joseph; KROL, Jason. Web development with MongoDB and NodeJs. Packt Publishing Ltd, 2015.

[23] FETTE, Ian; MELNIKOV, Alexey. The websocket protocol. 2011.

[24] PEZOA, Felipe, et al. Foundations of JSON schema. In: Proceedings of the 25th International Conference on World Wide Web. 2016. p. 263-273.

[25] SANNER, Michel F., et al. Python: a programming language for software integration and development. J Mol Graph Model, 1999, 17.1: 57-61.

[26] DHAMIJA, Rachna; TYGAR, J. Doug; HEARST, Marti. Why phishing works. In: Proceedings of the SIGCHI conference on Human Factors in computing systems. 2006. p. 581-590.

[27] HAUNTS, Stephen. What Are Data Breaches?. In: Applied Cryptography in. NET and Azure Key Vault. Apress, Berkeley, CA, 2019. p. 1-10.

[28] LONG, Johnny; GARDNER, Bill; BROWN, Justin. Google hacking for penetration testers. Elsevier, 2011.

[29] LUOTONEN, Ari. Web proxy servers. Prentice-Hall, Inc., 1998.

[30] https://github.com/haad/proxychains