Ασφάλεια διαδικτυακών εφαρμογών:
σχεδιαστικές λύσεις και αξιολόγηση της ασφάλειας

Μπεκιάρη Κωνσταντίνα
ΑΜ: 59913118

Επιβλέπων καθηγητής: Τριανταφύλλου Ιωάννης

Αντικείμενο πτυχιακής εργασίας

Θέμα:

Ασφάλεια διαδικτυακών εφαρμογών:
σχεδιαστικές λύσεις και αξιολόγηση της ασφάλειας

Βάσει των παραπάνω,
έχει πραγματοποιηθεί:

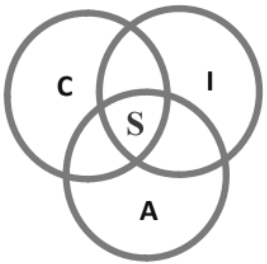
- σχεδίαση και ανάπτυξη μίας διαδικτυακής εφαρμογής και
- αξιολόγηση της ασφάλειας της μέσω hacking

Με στόχο να μελετηθεί το συγκεκριμένο φαινόμενο, έχουν μελετηθεί:

- τα χαρακτηριστικά των διαδικτυακών εφαρμογών
- η ασφάλεια πληροφοριών στο διαδίκτυο
- το hacking (penetration testing) σε σχέση με την αξιολόγηση της ασφάλειας και
- οι προϋποθέσεις σχεδίασης και ανάπτυξης ασφαλών διαδικτυακών εφαρμογών

Θεμελιώδεις αρχές ασφάλειας

1. Εμπιστευτικότητα (**C**onfidentiality)
2. Ακεραιότητα (**I**ntegrity)
3. Διαθεσιμότητα (**A**vailability)



S = Secure

Επιπρόσθετα στοιχεία ασφάλειας:

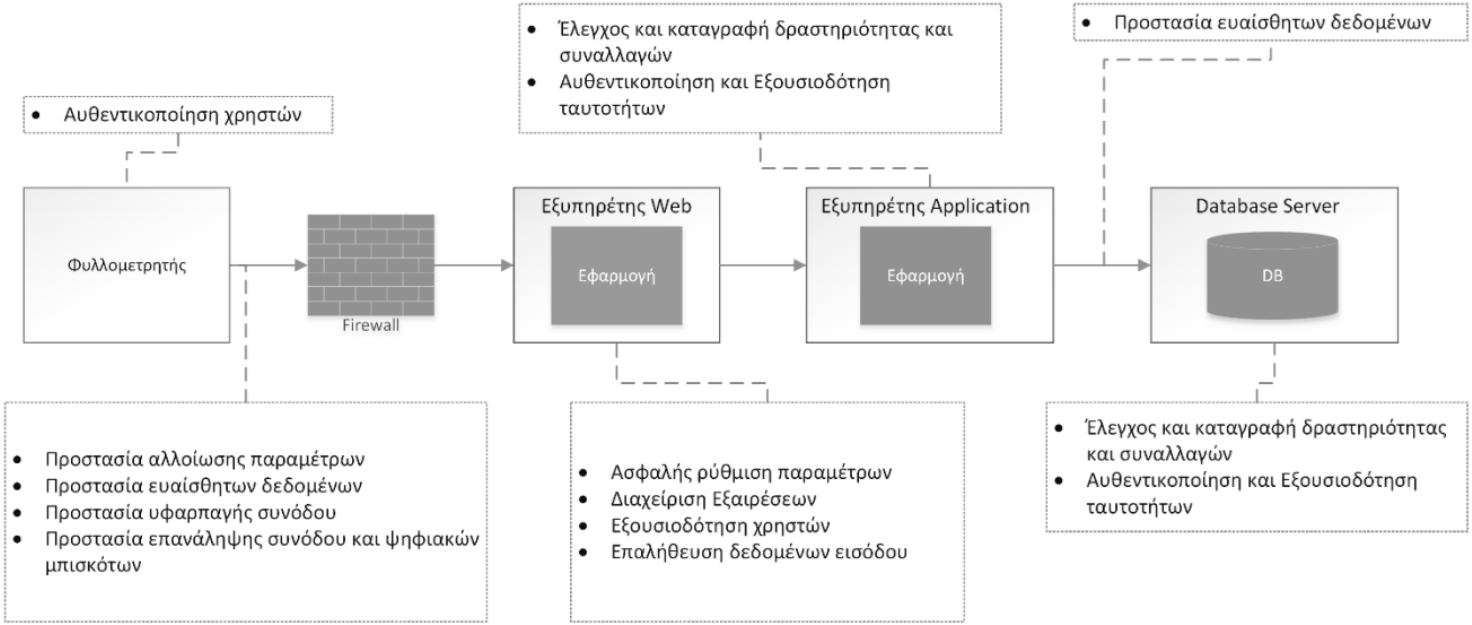
- Αυθεντικοποίηση (Authentication)
- Εξουσιοδότηση (Authorization)
- Μη-αποποίηση (Non-repudiation)
- Λογιστική καταγραφή (Accounting)

Αρχές ασφαλούς προγραμματισμού

- Η **Αρχή της Ελάχιστης Επιφάνειας Επίθεσης**
- Η **Αρχή του Ελάχιστου Προνομίου**
- Η **Αρχή του Διαχωρισμού Καθηκόντων**
- Η **Αρχή της Απλότητας**

+ Η χρήση περισσότερων του ενός μηχανισμών ασφάλειας.
+ **Security by obscurity**.

Σημαντικά ζητήματα ασφάλειας

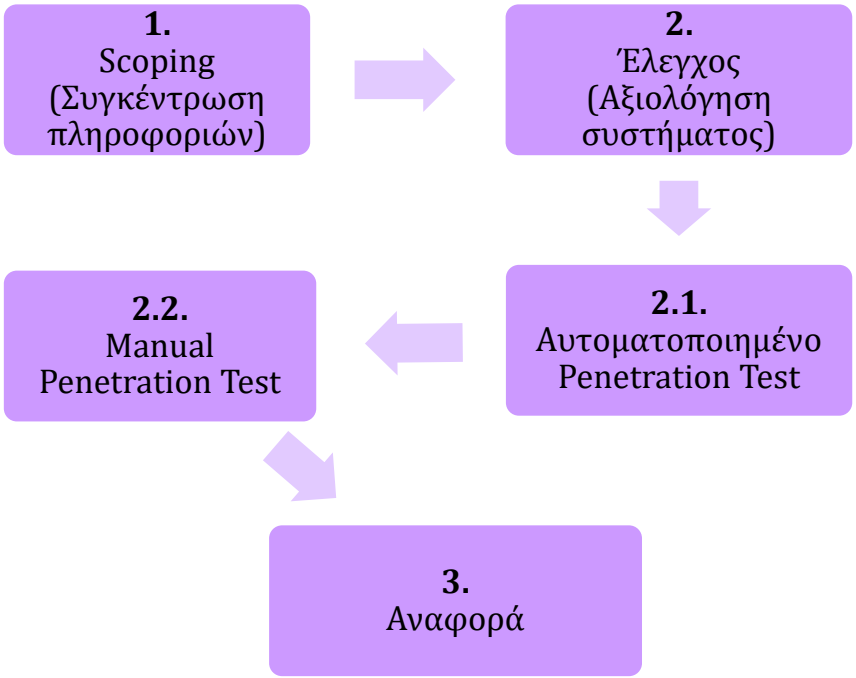


Εικόνα 1. *Μία τυπική αρχιτεκτονική μίας διαδικτυακής εφαρμογής συσχετιζόμενη με τα επιμέρους ζητήματα ασφάλειας διαδικτυακής εφαρμογής (πηγή: Μαυρίδης, 2015)*

TOP 10 αδυναμίες από την OWASP - 2021

- Injection
- Broken Authentication
- Sensitive Data Exposure
- XXE (XML External Entities)
- Broken Access Control
- Security Misconfiguration
- XSS (Cross-Site Scripting)
- Insecure Deserialization
- Using Components with Known Vulnerabilities
- Insufficient Logging and Monitoring

Αξιολόγηση ασφάλειας διαδικτυακών εφαρμογών



Τεχνικές ενίσχυσης ασφάλειας 1/2

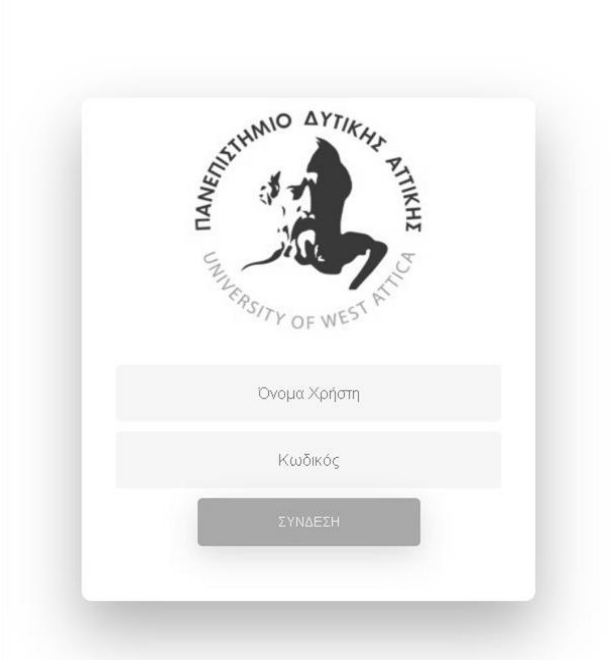
- Back-end κώδικας
- Waf (**W**eb **A**pplication **F**irewall)
- Two-factor Authentication
- Κρυπτογραφία
- LDAP (**L**ightweight **D**irectory **A**ccess **P**rotocol)
- Firewall
- IDS/IPS (**I**ntrusion **D**etection **S**ystem/ **I**ntrusion **P**revention **S**ystem)

Τεχνικές ενίσχυσης ασφάλειας 2/2

Back-end κώδικας

- Injection
- Insecure Deserialization
- Sensitive Data Exposure
- Broken Authentication
- Broken Access Control
- Security Misconfiguration
- Using Vulnerable Components
- Insufficient Logging and Monitoring

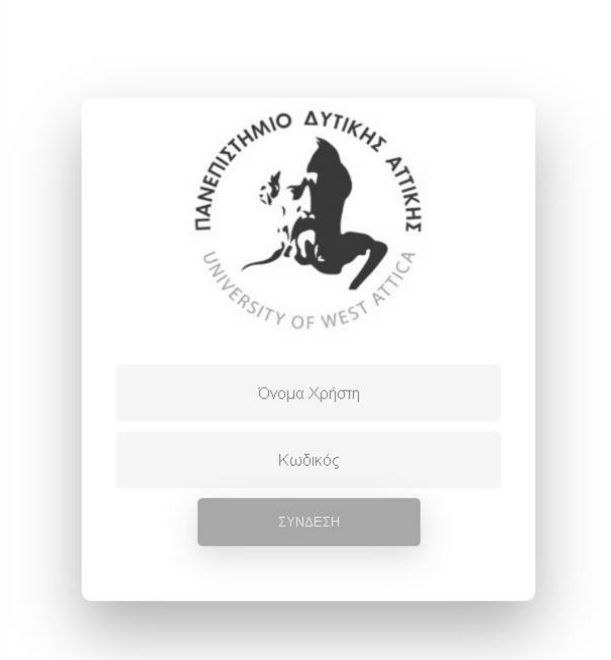
Περιγραφή εφαρμογής *“UniStudent”*



- Πρόκειται για μία εφαρμογή, τύπου Φοιτητολογίου – E-study, η οποία έχει σχεδιαστεί με βάση την ασφάλεια.

- Απευθύνεται στους φοιτητές που είναι εγγεγραμμένοι σε ένα τμήμα και βρίσκεται online στη διεύθυνση *https://www.unistudent.eu*.

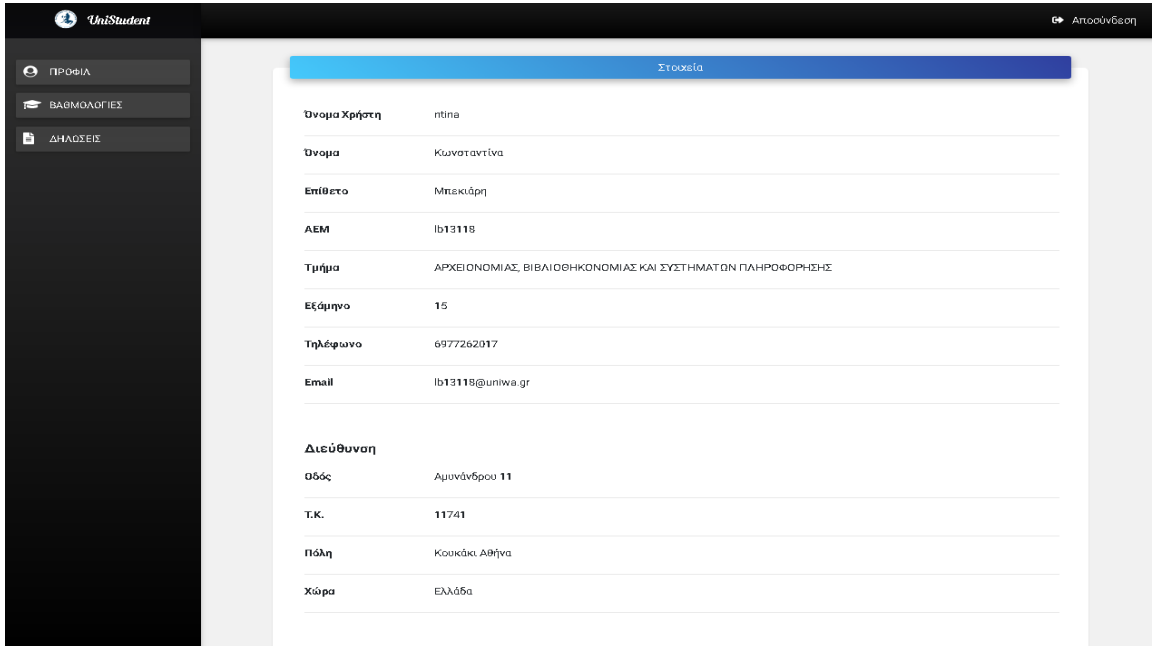
Περιγραφή εφαρμογής *“UniStudent”*



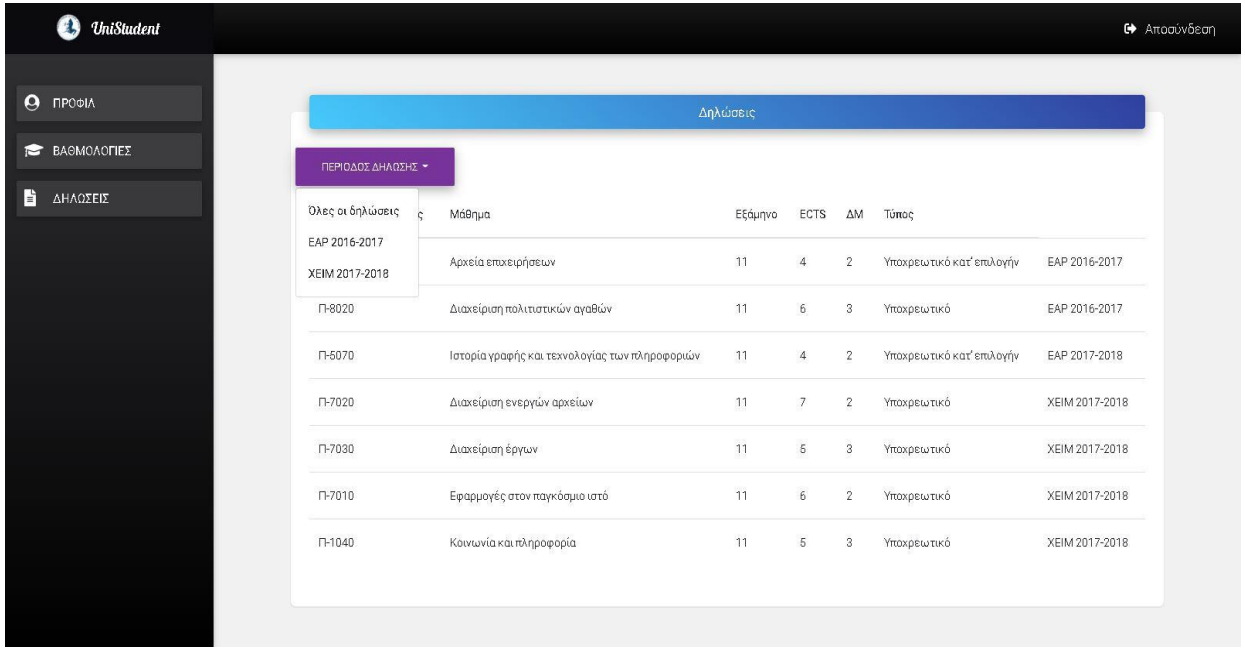
- Αποτελείται από 3 υπηρεσίες (σελίδες):

- Προφίλ,
- Δηλώσεις,
- Βαθμολογίες.

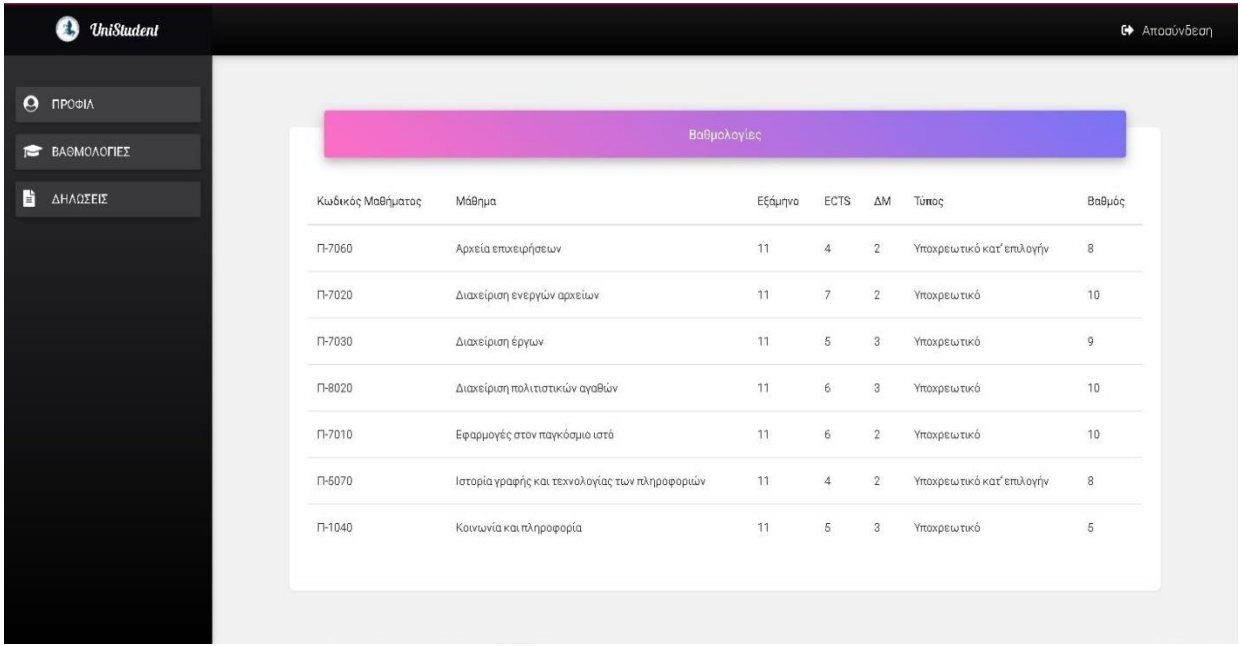
Προφίλ φοιτητή



Δηλώσεις φοιτητή

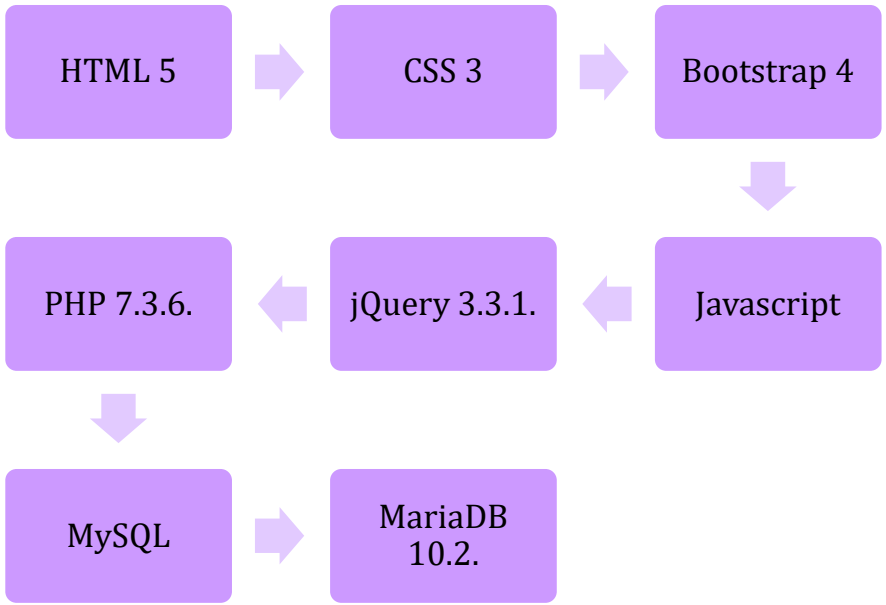


Βαθμολογίες φοιτητή



Σχεδίαση και ανάπτυξη εφαρμογής 1/2

Για την ανάπτυξη της εφαρμογής χρησιμοποιήθηκαν οι εξής τεχνολογίες:

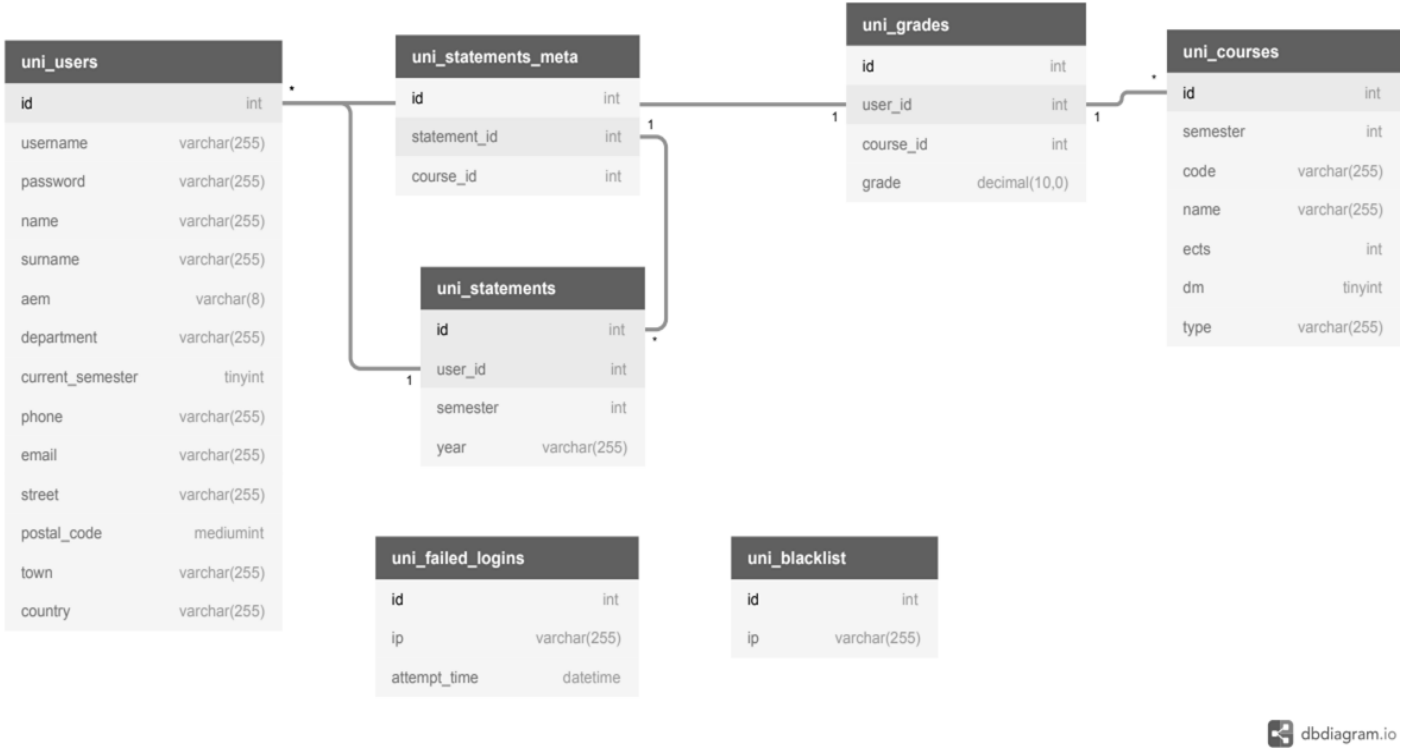


Σχεδίαση και ανάπτυξη εφαρμογής 2/2

Η εφαρμογή έχει αναπτυχθεί με στόχο να δεχθεί και να αμυνθεί σε επιθέσεις τύπου:

- SQL Injection
- PHP Injection
- Broken Authentication
- Sensitive Data Exposure
- Security Misconfiguration
- Brute Force
- XSS

Περιγραφή βάσης



Τεχνικές ασφάλειας εφαρμογής 1/3

- Για την αποφυγή **XSS**, **SQL** και **PHP Injections** φιλτράρεται η εισαγωγή δεδομένων και αποκόπτονται:
  - κενά,
  - ειδικοί χαρακτήρες.
- Οι κωδικοί πρόσβασης είναι αποθηκευμένοι στον πίνακα **uni_users** και κρυπτογραφημένοι με τη χρήση του **Bcrypt**.
- Η εφαρμογή προστατεύεται επίσης από **Επιθέσεις αποκάλυψης πληροφοριών**, αποτρέποντας τον επιτιθέμενο από το να δει τον κατάλογο αρχείων.

Τεχνικές ασφάλειας εφαρμογής 2/3

- Σχετικά με την παραβίαση **Broken authentication**, έχει αποφευχθεί η χρήση **Cookies**. Το session id αποθηκεύεται στον server, οπότε είναι αδύνατο να κλαπούν πληροφορίες της session.

- Για την παρεμπόδιση **Brute Force επιθέσεων**,

- αν η επίθεση περιέχει αλφαριθμητικούς χαρακτήρες, μετά από 5 αποτυχημένες προσπάθειες εντός 1 ώρας, ο χρήστης ανακατευθύνεται στο αρχείο **blocked.php**. Η IP διεύθυνση και η χρονική στιγμή της πρώτης λανθασμένης εισόδου καταγράφονται στον πίνακα **uni_failed_logins**.

- αν η επίθεση περιέχει ειδικούς χαρακτήρες, η IP διεύθυνση καταγράφεται στον πίνακα **uni_blacklist**.

Τεχνικές ασφάλειας εφαρμογής 3/3

- Με την ύπαρξη ενός **honeypot** στη σελίδα σύνδεσης, όταν ο επιτιθέμενος (π.χ. bot) συμπληρώσει οτιδήποτε στο πεδίο “Don’t fill if human..” η IP διεύθυνση του μπλοκάρεται από τον server της εφαρμογής και καταγράφεται στην Blacklist.

- Για περισσότερη ασφάλεια έχει **απενεργοποιηθεί** η δυνατότητα μεταφόρτωσης αρχείων (file upload) από τις ρυθμίσεις του server.

- Τέλος, σε ένα κρυφό αρχείο (.htaccess) έχει οριστεί να **αποτρέπεται** η χρήση αιτημάτων **DELETE PUT UPDATE**.

Αξιολόγηση ασφάλειας εφαρμογής 1/6

- Το penetration test της εφαρμογής:

- βασίστηκε στις **OWASP Top 10** αδυναμίες διαδικτυακών εφαρμογών,
- πραγματοποιήθηκε αρχικά **Black Box** και έπειτα, **White Box**.

- Χρησιμοποιήθηκαν 3 αυτοματοποιημένα εργαλεία:

- Gobuster
- Sqlmap
- Nikto

Αξιολόγηση ασφάλειας εφαρμογής 2/6

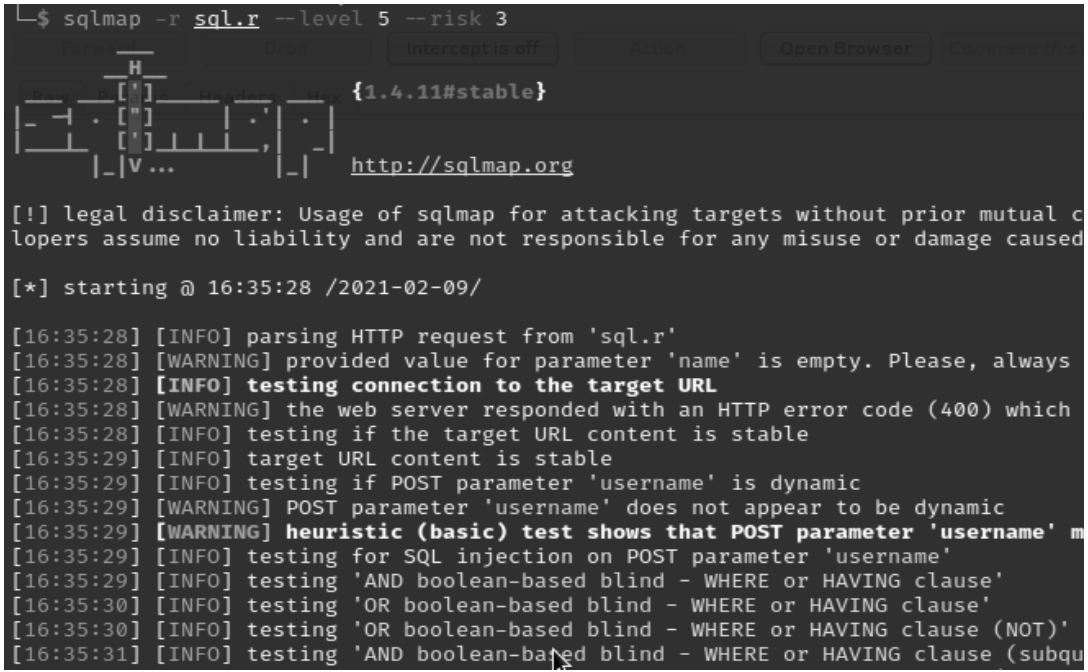
Επιθέσεις εφαρμογής *"UniStudent"*				
OWASP Top 10 αδυναμίες διαδικτυακών εφαρμογών		**Αυτοματοποιημένα εργαλεία**		
		Sqlmap	**Nikto**	**Gobuster**
Injection	✓	✓		
Broken Authentication	✓		✓	
Sensitive Data Exposure	✓		✓	
XML External Entities (XXE)	-	-	-	
Broken Access Control	✓		✓	
Security Misconfiguration	✓		✓	
Cross Site Scripting (XSS)	✓		✓	
Insecure Deserialization	-	-	-	
Using Components with Known Vulnerabilities	-	-	-	
Insufficient Logging and Monitoring	-	-	-	
Επιπρόσθετες Αδυναμίες				
Πιθανές άλλες αδυναμίες	✓			✓

Αξιολόγηση ασφάλειας εφαρμογής 3/6

- Το πιο ενδιαφέρον από τα αποτελέσματα του Gobuster είναι το αρχείο login.php και γι' αυτό εξετάστηκε πρώτο.
- Όταν υπάρχει σελίδα σύνδεσης, σημαίνει ότι υπάρχει κάποια βάση δεδομένων στο back-end, η οποία εξυπηρετεί την αυθεντικοποίηση.

```
└─$ gobuster dir -u http://192.168.1.20/unistudent/ -w /opt/SecLists/Discovery/Web-Content/di
===============================================================
Gobuster v3.0.1
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@_FireFart_)
===============================================================
[+] Url:            http://192.168.1.20/unistudent/
[+] Threads:        30
[+] Wordlist:       /opt/SecLists/Discovery/Web-Content/directory-list-2.3-medium.txt
[+] Status codes:   200,204,301,302,307,401,403
[+] User Agent:     gobuster/3.0.1
[+] Extensions:     txt,html,php
[+] Timeout:        10s
===============================================================
2021/02/09 16:39:22 Starting gobuster
===============================================================
/index.php (Status: 302)
/login.php (Status: 200)
/profile.php (Status: 302)
/img (Status: 301)
/css (Status: 301)
/cgi-bin (Status: 301)
/js (Status: 301)
/logout.php (Status: 200)
/cron.php (Status: 200)
/functions.php (Status: 200)
/statements.php (Status: 302)
/font (Status: 301)
/blocked.php (Status: 200)
/authenticate.php (Status: 200)
/grades.php (Status: 302)
```

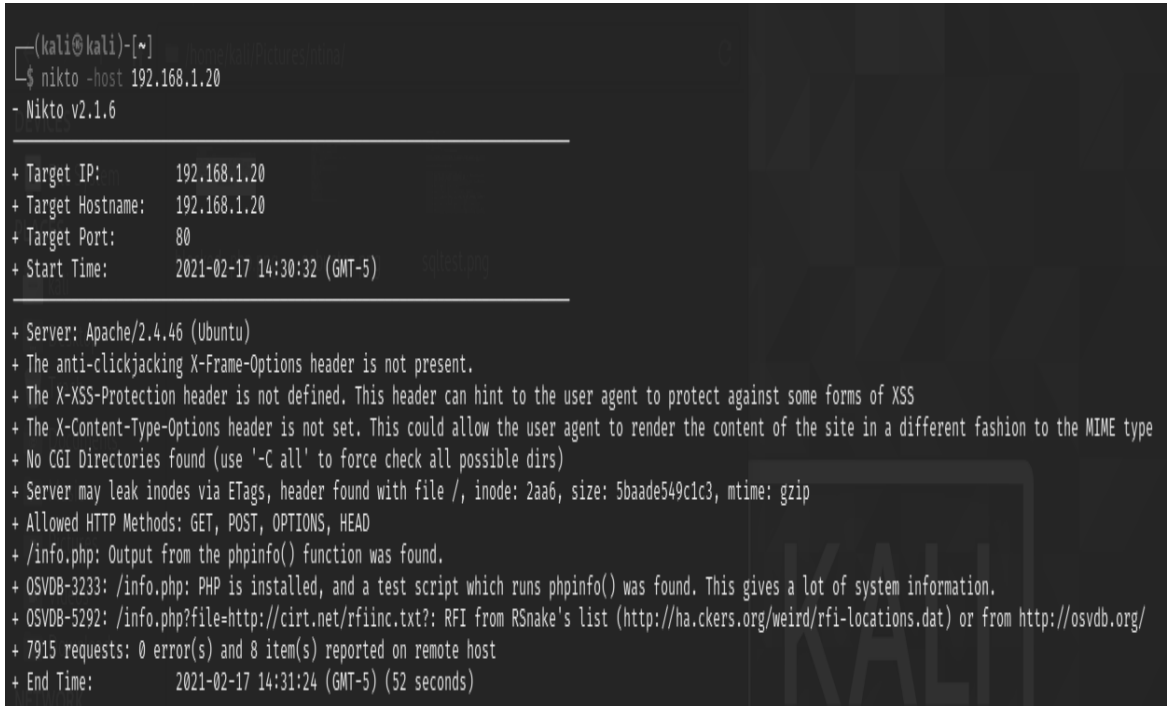
Αξιολόγηση ασφάλειας εφαρμογής 4/6



- Το Sqlmap δεν κατάφερε να βρει κάποιο query.
- Παρατηρήθηκε ότι επιστρέφεται διαφορετικό μήνυμα λάθους, ανάλογα με τα δεδομένα εισόδου.
- Αυτό αποτελεί μία ισχυρή ένδειξη ότι η εφαρμογή ενδεχομένως να διαθέτει κάποιον μηχανισμό ανίχνευσης χαρακτήρων.
- Στην ίδια σελίδα υλοποιήθηκαν έλεγχοι για XSS, αλλά δεν βρέθηκε κάποια αδυναμία.

Αξιολόγηση ασφάλειας εφαρμογής 5/6

- Το μόνο που βρέθηκε στα αποτελέσματα του Nikto είναι το αρχείο info.php.



Αξιολόγηση ασφάλειας εφαρμογής 6/6

Αδυναμίες εφαρμογής *"UniStudent"*		
OWASP Top 10 αδυναμίες διαδικτυακών εφαρμογών	**Βρέθηκαν**	**Παρατηρήσεις**
Injection	×	Έλεγχος δεδομένων εισόδου
Broken Authentication	×	Αποφυγή χρήσης Cookies
Sensitive Data Exposure	×	Αποφυγή σχολίων, back-up αρχείων και δοκιμαστικών directories
XML External Entities (XXE)	-	Η αδυναμία XML External Entities (XXE) απαιτεί την ύπαρξη xml κώδικα.
Broken Access Control	×	Ύπαρξη μόνο 2 απλών χρηστών χωρίς αναβαθμισμένο ρόλο.
Security Misconfiguration	×	Χρήση σωστών ρυθμίσεων και πρακτικών
Cross Site Scripting (XSS)	×	Έλεγχος δεδομένων εισόδου
Insecure Deserialization	-	Η αδυναμία Insecure Deserialization απαιτεί την ύπαρξη αντικειμενοστραφούς προγραμματισμού.
Using Components with Known Vulnerabilities	-	Η αδυναμία Using Components with Known Vulnerabilities απαιτεί την ύπαρξη έτοιμων βιβλιοθηκών και συναρτήσεων.
Insufficient Logging and Monitoring	-	Η αδυναμία Insufficient Logging and Monitoring απαιτεί την ύπαρξη αρχείων logs.
Επιπρόσθετες αδυναμίες		
Information Disclosure	✓	Δεν έχει πραγματοποιηθεί κάποια προσπάθεια αποφυγής της συγκεκριμένης αδυναμίας.

Επίλογος

- Η ασφάλεια διαδικτυακών εφαρμογών κατά τη σχεδίαση και την ανάπτυξη τους θεωρείται πολύ σημαντική.

  Εξίσου σημαντική είναι και η αξιολόγηση της ασφάλειας τους μέσω hacking (penetration testing).

- Υπάρχουν διάφορες τεχνικές και μέθοδοι ασφάλειας, τόσο σε επίπεδο εφαρμογής όσο και σε επίπεδο συστήματος και δικτύου.

Οι επαγγελματίες διαφορετικών ειδικοτήτων, οι οποίοι συμμετέχουν στη σχεδίαση και στην ανάπτυξη μίας εφαρμογής, θα πρέπει:

- να έχουν πρακτική και θεωρητική γνώση
- να εξελίσσονται συνεχώς και
- να είναι σε θέση να συνεργαστούν μεταξύ τους, με στόχο να βρεθεί η “χρυσή τομή”.

Βασικός στόχος τους θα πρέπει να είναι η ύψιστη δυνατή ασφάλεια.