

Ασφάλεια Πληροφοριακών Συστημάτων

Τελική Εργασία Μαθήματος:
Εγχειρίδιο Προγραμματιστή

Ακαδημαϊκό Έτος: 2022 – 2023

Ομάδα:



Μπουμπλίνη Αναστασία
(Π19117)



aboublini@gmail.com



ANASTASIA BOUBLINI
(p19117@unipi.gr)



Μπριστογιάννης
Ιωακείμ (Π19048)



ioakeim13@hotmail.gr



IOAKEIM EL-KHATTAB-
BRISTOGIANNIS
(p19048@unipi.gr)



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ
UNIVERSITY OF PIRAEUS

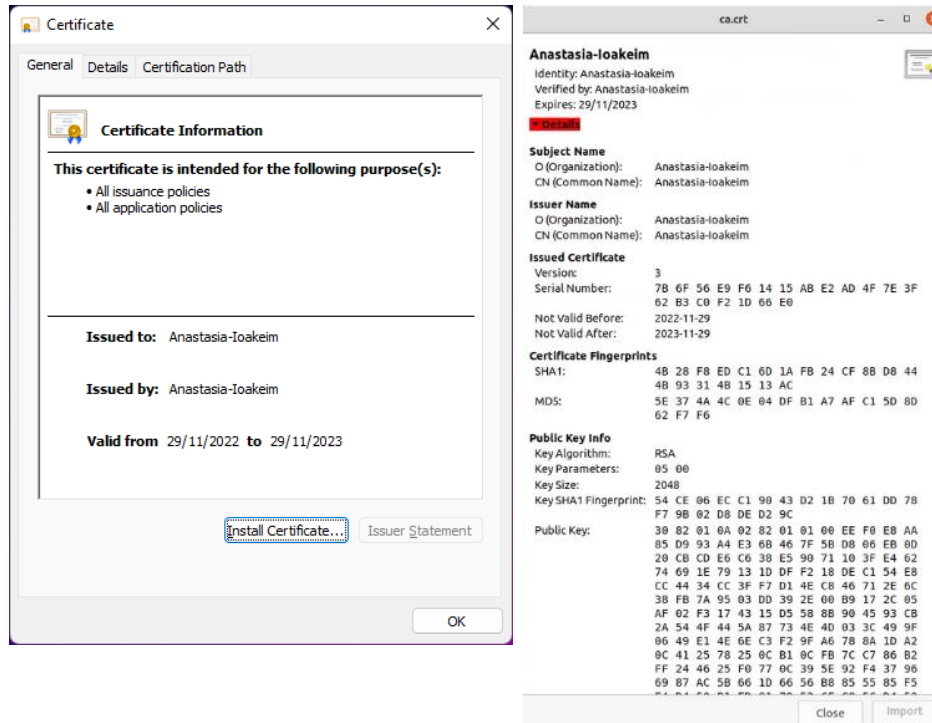
Πίνακας Περιεχομένων

1.	Κρυπτογράφηση SSL	3
1.1.	Δημιουργία Αρχής Πιστοποίησης	3
1.2.	Δημιουργία και Πιστοποίηση Κλειδιών για Server	4
1.3.	Εισαγωγή Πιστοποιητικού στον Server	6
2.	Μέθοδοι Αυθεντικοποίησης και Ελέγχου Πρόσβασης	6
2.1.	Αυθεντικοποίηση Χρήστη (Authentication)	6
2.1.1.	Password Hashing στη Βάση Δεδομένων	7
2.1.2.	Έλεγχος Εγκυρότητας Κωδικού Πρόσβασης	7
2.2.	Έλεγχος Πρόσβασης (Cookie Authorization)	8
3.	Input Filtering and Validation	9
3.1.	Input Filtering και Validation με Regular Expressions	9
3.2.	Ειδική Περίπτωση: Cross Site Scripting - XSS	9
4.	Αυτοματοποιημένη Εύρεση Ευπαθειών	10
4.1.	Port Scanning με Nmap	10
4.2.	Certificate Scanning με Nmap	12
4.3.	Security Scanning με Nikto	13
4.4.	Επιθέσεις με SQLmap	13
4.4.	TLS/SSL Scanning με SSLScan	14
4.5.	Ολικό Scanning με SSLyze	16

1. Κρυπτογράφηση SSL

1.1. Δημιουργία Αρχής Πιστοποίησης

Δημιουργήθηκε το παρακάτω αυτό-υπογεγραμμένο πιστοποιητικό (Εικόνες 1 & 2) σε περιβάλλον Linux κάνοντας χρήση του OpenSSL. Ακολουθήθηκε η διαδικασία που φαίνεται στην Εικόνα 3.



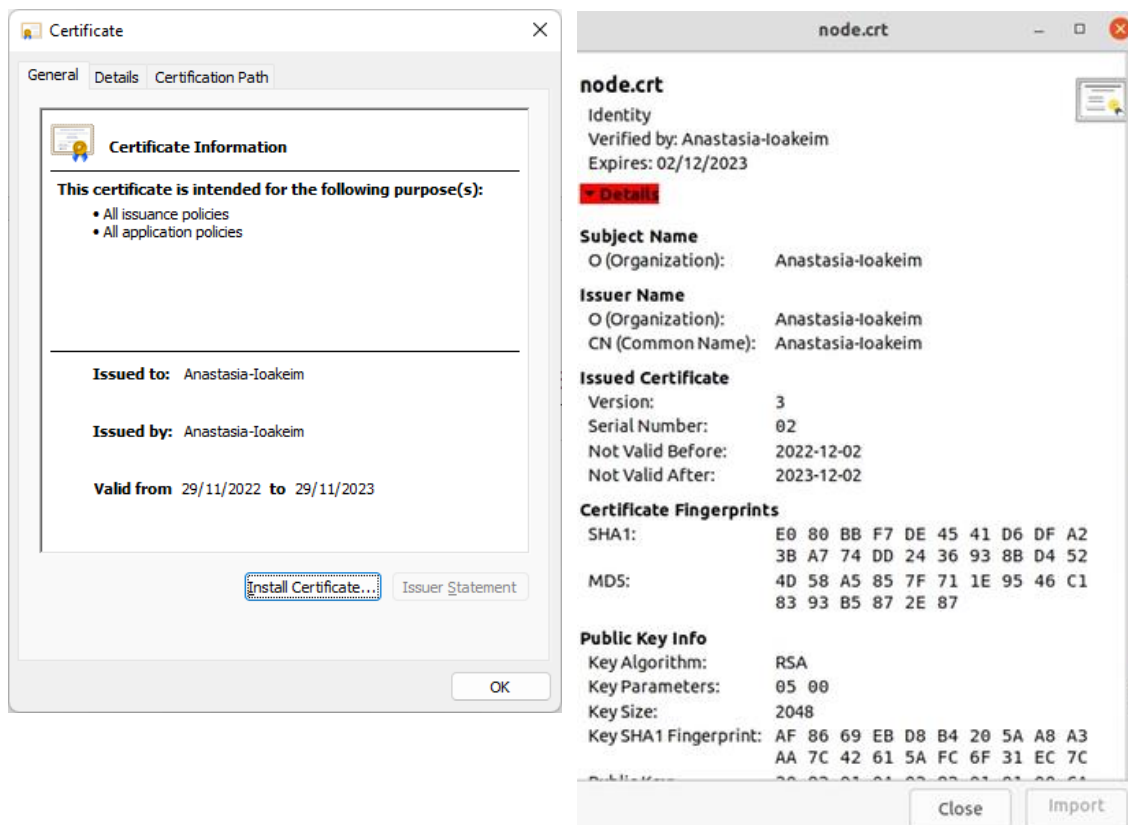
Εικόνες 1 & 2: Προβολή self-signed πιστοποιητικό από Windows OS (αριστερά) και Linux OS (δεξιά).

```
(base) loakeim@loakeim-VivoBook-ASUSLaptop-TP420IA-TM420IA:~/Desktop/asfaleia$ touch ca.cnf
(base) loakeim@loakeim-VivoBook-ASUSLaptop-TP420IA-TM420IA:~/Desktop/asfaleia$ ls
ca.cnf  cacnf  certs  my-safe-directory
(base) loakeim@loakeim-VivoBook-ASUSLaptop-TP420IA-TM420IA:~/Desktop/asfaleia$ openssl genrsa -out my-safe-directory/ca.key 2048
Generating RSA private key, 2048 bit long modulus (2 primes)
.....+++++
e is 65537 (0x010001)
(base) loakeim@loakeim-VivoBook-ASUSLaptop-TP420IA-TM420IA:~/Desktop/asfaleia$ chmod 400 my-safe-directory/ca.key
(base) loakeim@loakeim-VivoBook-ASUSLaptop-TP420IA-TM420IA:~/Desktop/asfaleia$ openssl req -new -x509 -config ca.cnf -key my-safe-dir
ectory/ca.key -out certs/ca.crt -days 365 -batch
(base) loakeim@loakeim-VivoBook-ASUSLaptop-TP420IA-TM420IA:~/Desktop/asfaleia$ openssl genrsa -out my-safe-directory/ca.key 2048
Generating RSA private key, 2048 bit long modulus (2 primes)
.....+++++
e is 65537 (0x010001)
(base) loakeim@loakeim-VivoBook-ASUSLaptop-TP420IA-TM420IA:~/Desktop/asfaleia$ chmod 400 my-safe-directory/ca.key
(base) loakeim@loakeim-VivoBook-ASUSLaptop-TP420IA-TM420IA:~/Desktop/asfaleia$ openssl req -new -x509 -config ca.cnf -key my-safe-dir
ectory/ca.key -out certs/ca.crt -days 365 -batch
problems making Certificate Request
139791690835264:error:0D07A098:asn1 encoding routines:ASN1_mbstring_ncopy:string too short:crypto/asn1/a_mbstr.c:100:minsiz=1
(base) loakeim@loakeim-VivoBook-ASUSLaptop-TP420IA-TM420IA:~/Desktop/asfaleia$ openssl req -new -x509 -config ca.cnf -key my-safe-dir
ectory/ca.key -out certs/ca.crt -days 365 -batch
```

Εικόνα 3: Διαδικασία δημιουργίας self-signed πιστοποιητικού σε Linux.

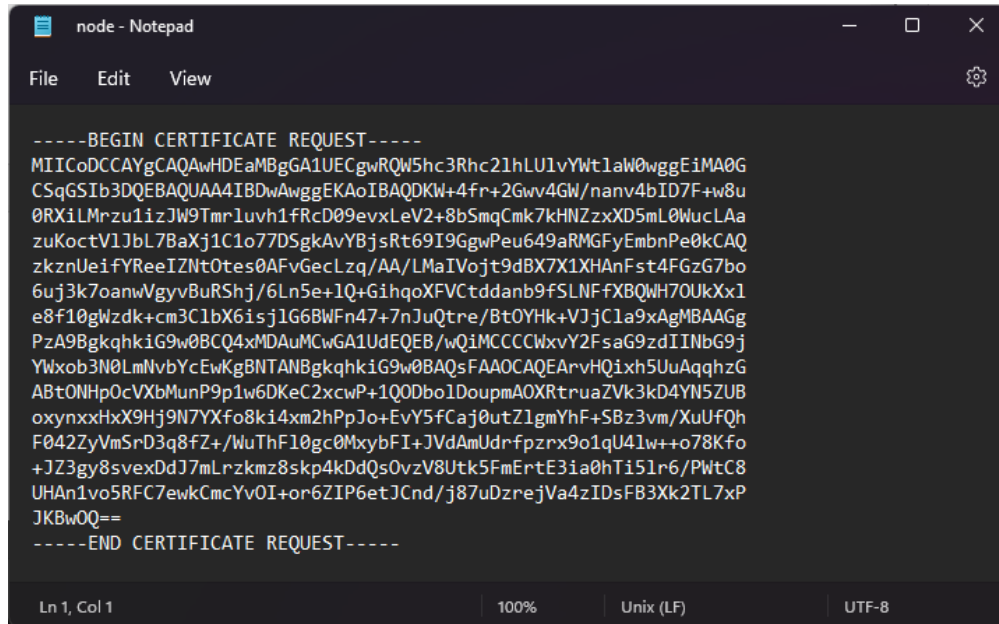
1.2. Δημιουργία και Πιστοποίηση Κλειδιών για Server

Δημιουργήθηκε το παρακάτω ζεύγος κλειδιών για τον web server (Εικόνες 4 & 5).



Εικόνες 4 & 5: Προβολή ζεύγους κλειδιών για τον Kestrel Server από Windows OS (αριστερά) και Linux OS (δεξιά).

Στη συνέχεια, δημιουργήθηκε ένα αίτημα certificate signing request (CSR) προς την δοκιμαστική αρχή πιστοποίησης ώστε να υπογράψει το πιστοποιητικό του server, τροποποιώντας κατάλληλα το αρχείο διαμόρφωσης της ΑΠ ώστε το πιστοποιητικό του server να περιλαμβάνει τα αντίστοιχα constraints (basic constraints, key usage, extended key usage) που αντιστοιχούν σε αυτόν (Εικόνα 6).



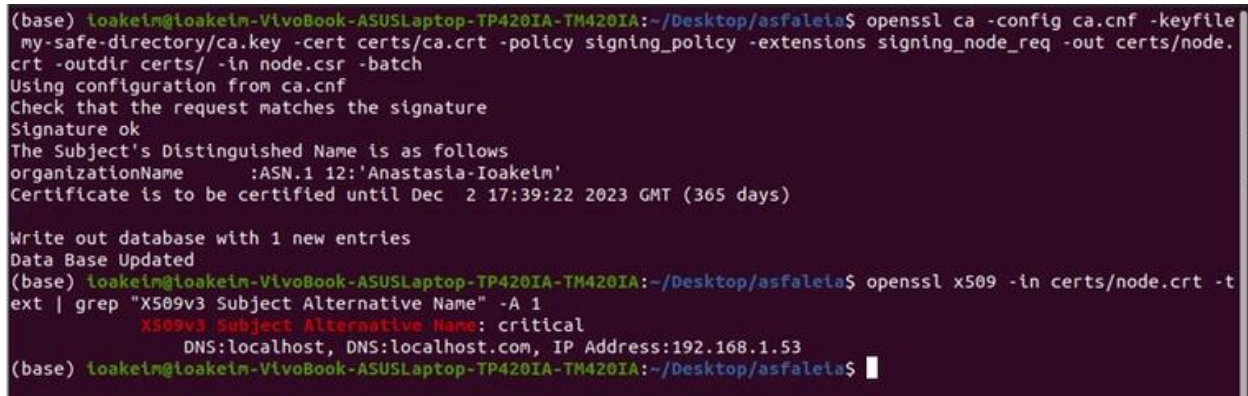
```
node - Notepad
File Edit View

-----BEGIN CERTIFICATE REQUEST-----
MIICoDCCAYgCAQAwHDEaMBGGA1UECgwRQW5hc3RhczI1hLU1vYWtlaW0wggiEiMA0G
CSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQDKW+4fr+2Gwv4GW/nanv4bID7F+w8u
0RXiLMrzu1izJW9Tmr1uvh1fRcD09evxLeV2+8bSmqCmk7kHNZzxXD5mL0WucLAA
zuKoctV1JbL7BaXj1C1o77DSgkAvYBjsRt69I9GgwPew649aRMGFyEmbnPe0kCAQ
zkznUeifYReeIZNt0tes0AFvGecLzq/AA/LMaIVojt9dBX7X1XHAnFst4FGzG7bo
6uj3k7oanwVgyvBuRShj/6Ln5e+1Q+GihqoXFVctddanb9fSLNFfXBQW70UkXx1
e8f10gWzdk+cm3C1bX6isjlG6BWFN47+7nJuQtire/Bt0YHk+VJjC1a9xAgMBAAGg
PzA9BgkqhkiG9w0BCQ4xMDAuMCwGA1UdEQEB/wQIMCCCCWxvY2FsaG9zdIINbG9j
YXxob3N0LmNvbYcEwKgBNTANBgkqhkiG9w0BAQsFAAOCAQEArvHQixh5UuAqghzG
ABtONhp0cVXbMunP9p1w6DKEc2xcwP+1Q0Dbo1DoupMAOXRtruaZVk3kD4YN5ZUB
oxynxxHxX9Hj9N7YXfo8ki4xm2hPpJo+EvY5fCaj0utZlgmYhF+SBz3vm/XuUfQh
F042ZyVmSrD3q8fZ+/WuThF10gc0MxybFI+JVdAmUdrfpzrx9o1qU41w++o78Kfo
+JZ3gy8svexDdJ7mLrzkmz8skp4kDdQs0vzV8Utk5FmErtE3ia0hTi51r6/PWtC8
UHAN1vo5RfC7ewkCmcYvOI+or6ZIP6etJCnd/j87uDzrejVa4zID5FB3Xk2TL7xP
JKBwOQ==
-----END CERTIFICATE REQUEST-----

Ln 1, Col 1 100% Unix (LF) UTF-8
```

Εικόνα 6: Certificate Signing Request προς την δοκιμαστική ΑΠ.

Η διαδικασία που ακολουθήθηκε, σε περιβάλλον Linux, φαίνεται στην Εικόνα 7.



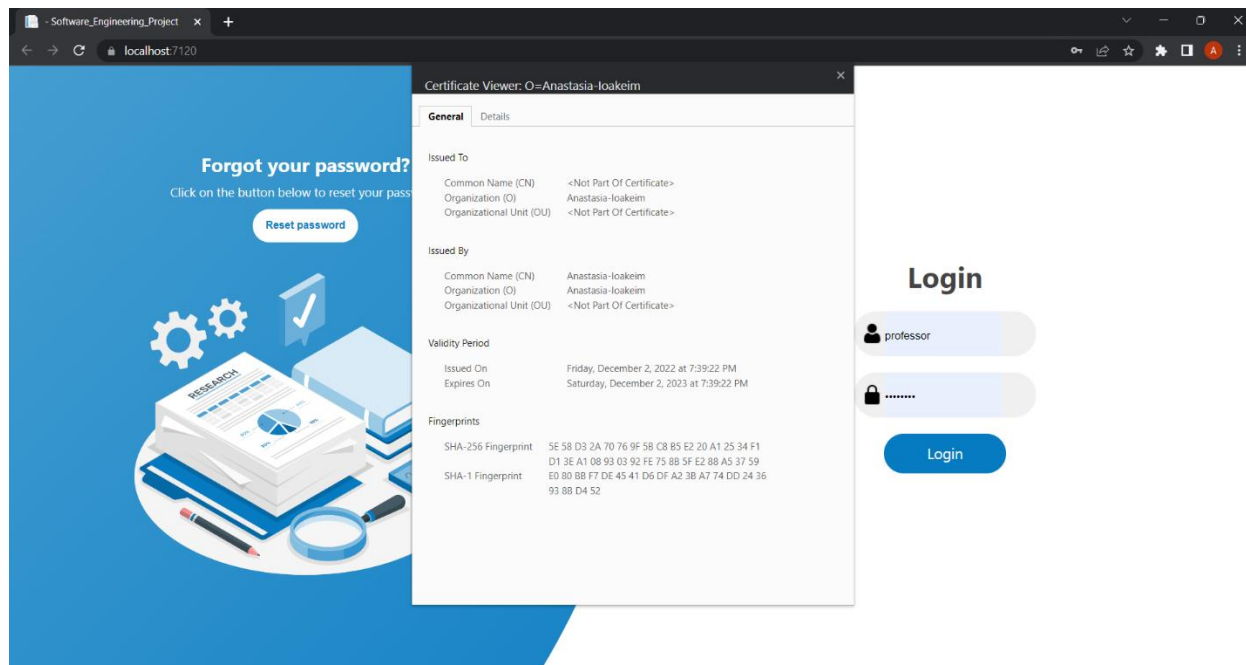
```
(base) loakeim@loakeim-VivoBook-ASUSLaptop-TP420IA-TM420IA:~/Desktop/asfaleia$ openssl ca -config ca.cnf -keyfile
my-safe-directory/ca.key -cert certs/ca.crt -policy signing_policy -extensions signing_node_req -out certs/node.
crt -outdir certs/ -in node.csr -batch
Using configuration from ca.cnf
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
organizationName      :ASN.1 12:'Anastasia-Ioakeim'
Certificate is to be certified until Dec  2 17:39:22 2023 GMT (365 days)

Write out database with 1 new entries
Data Base Updated
(base) loakeim@loakeim-VivoBook-ASUSLaptop-TP420IA-TM420IA:~/Desktop/asfaleia$ openssl x509 -in certs/node.crt -t
ext | grep "X509v3 Subject Alternative Name" -A 1
      X509v3 Subject Alternative Name: critical
      DNS:localhost, DNS:localhost.com, IP Address:192.168.1.53
(base) loakeim@loakeim-VivoBook-ASUSLaptop-TP420IA-TM420IA:~/Desktop/asfaleia$
```

Εικόνα 7: Διαδικασία δεύτερου ερωτήματος σε Linux

1.3. Εισαγωγή Πιστοποιητικού στον Server

Η εισαγωγή του πιστοποιητικού στον Kestrel Server (Εικόνα 8) έγινε τροποποιώντας κατάλληλα το αρχείο appsettings.json της διαδικτυακής εφαρμογής.



Εικόνα 8: Πιστοποιητικό στον Kestrel Server

2. Μέθοδοι Αυθεντικοποίησης και Ελέγχου Πρόσβασης

2.1. Αυθεντικοποίηση Χρήστη (Authentication)

Η αυθεντικοποίηση χρήστη γίνεται κατά την είσοδο του στην εφαρμογή και βασίζεται στον έλεγχο του ονόματος και του συνθηματικού που θα εισάγει ο χρήστης (username, password). Συγκεκριμένα, ο χρήστης εισάγει τα διαπιστευτήριά του και, ανάλογα με την ιδιότητα του (φοιτητής ή καθηγητής) ανακατευθύνεται και στην αντίστοιχη σελίδα (Εικόνες 9, 10).



Εικόνες 9 & 10: Αρχικές σελίδες φοιτητή (αριστερά) και καθηγητή (δεξιά)

2.1.1. Password Hashing στη Βάση Δεδομένων

Οι κωδικοί πρόσβασης όλων των χρηστών είναι αποθηκευμένοι στη βάση δεδομένων σε hashed μορφή. Συγκεκριμένα, στο αρχείο Database.cs υπάρχει η συνάρτηση HashPassword(), η οποία δέχεται ένα plaintext (password) και το μετατρέπει στην αντίστοιχη hashed μορφή του. Ο τρόπος λειτουργίας ανάγεται στη χρήση της μεθόδου Pdkdf2, η οποία χρησιμοποιεί τον αλγόριθμο κρυπτογράφησης SHA512. Αναλυτικότερα, δημιουργείται μια τυχαία ακολουθία από bytes, το γνωστό “salt”, με την χρήση της κρυπτογραφικής μεθόδου Cryptography.RandomNumberGenerator και γίνεται hash ο εκάστοτε κωδικός (plaintext), μαζί με μια τυχαία σταθερά της κλάσης Database (pepper). Τέλος, επιστρέφεται το hashed password, μαζί με τον πίνακα byte του “salt”. Αξίζει να σημειωθεί, πως κατά την λειτουργία αλλαγής κωδικού πρόσβασης από τον χρήστη εφαρμόζεται η ίδια διαδικασία για την αποθήκευση του νέου κωδικού στη βάση (βλ. StudentController.cs, TeacherController.cs). Στην Εικόνα 11, φαίνονται οι αποθηκευμένοι κωδικοί της βάσης δεδομένων σε hashed μορφή.

	username [PK] character varying (30)	password character varying
1		7360F982B95D1D38ABC62D47E96B209BF610B09EDEF2BEB0E92079DE775134F7239BEEDA6D4E1D0113B540BCB54BC100D5419DF4AE05136A2F8ABEA8F3572DB
2		F6561546ACA8BF0A59D6ED2E7B760AB0178DB8F78017F4A8DD8B070050A66BFAFA210BD2816404D210978E2F349D6FB976C49D5A6DC2D5362B142586853345C7
3		4C2001CEC865CD25D22A28344315371F36B042A6B068FCB6A7E539DB5760BA091DAD984DDE0A43713E1883A4C0347BFED354C1D1D81097907030AD8642A02937
4		F3E2F4B4567F7F6A3347C8BDA406B85680C70DF53DBC25E8365F5A57CE0B6BF7ED259EDACC758C5976737A9E1659C7CB04CA22EBB4A80F74FF70E78F5F1D567F
5		24E193F599324BD275DE799490F1CAD8E4B663D0F453CEB1B434308BF7F035DB21D9C3A81F13AFA4C2EED4645FF50EF575ACE4C95F6D3916008DF9FB92C30A44
6		A3AD579C20792617C0BB1B672F7033A2BAEA70FFB1CF3A186B9DCE8BCC84EB44CFE63F92B7CB120683E78154E14F8CE9819F9C3522796574651FC22176D4A333
7		6C61D9E7BCEFC7A4E02CFA083F54E61851A2F3D978C245B069646EC5C9682E92C5BAA21159E4CCEB5B9D424CC9ED978C5E6568275732A9E75465DCA1948645
8		AA9785A1CC2182248CA3E2838D07DE8A8D3D7BB5A003F1043F2CC07930063D693CD349BF727DD6159811E1DB73AD394449F60CB673FB4F729F55F435600EF377
9	student1	79622B924C9A100C1EFDAD12C3EE819B4CF9CEC64893A5050EA83A3B38B48C3D1C4DC82DC28C971CAAEBD2CFBD0A8E7EFEFDD80FD9D40F2EBC5ED3A68885E5A
10	student2	DD4A23D9408DD89655E97E61B797489760F70DE617BB2EC510CB759EFDCE28AF29E76595C2F259DF13A5AB398A87F7ABB87618F04F44A9A56A6F42FAB69DB388

Εικόνα 11: Οι κωδικοί πρόσβασης της βάσης δεδομένων σε hashed μορφή.

2.1.2. Έλεγχος Εγκυρότητας Κωδικού Πρόσβασης

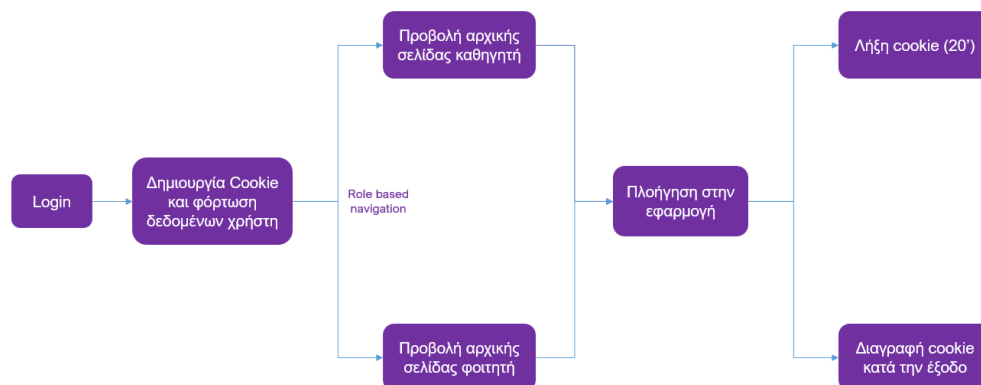
Σε λειτουργίες όπως η είσοδος του χρήστη στην εφαρμογή και η αλλαγή κωδικού πρόσβασης, όπου ο χρήστης πρέπει να εισάγει τον προσωπικό του κωδικό (Εικόνες 12, 13), ώστε να εξακριβωθεί η ταυτότητά του, είναι απαραίτητο να ελέγχεται η εγκυρότητα του κωδικού, σύμφωνα με αυτόν που είναι αποθηκευμένος στη βάση δεδομένων σε hashed μορφή. Αυτή η λειτουργία πραγματοποιείται με τη χρήση της συνάρτησης VerifyPassword(), η οποία βρίσκεται στο αρχείο Database.cs και συγκρίνει το plaintext που εισάγει χρήστης με το αντίστοιχο hash που βρίσκεται στη βάση δεδομένων. Συγκεκριμένα, το plaintext γίνεται hash μαζί με το “salt”, που του αντιστοιχεί από τη βάση και το αποτέλεσμα συγκρίνεται με το, ήδη υπάρχον, hashed password. Τέλος επιστρέφεται μια boolean τιμή, η οποία υποδεικνύει την επιτυχή ή μη επιτυχή αναγνώριση του κωδικού πρόσβασης.



Εικόνες 12 & 13: Λειτουργίες της εφαρμογής που απαιτούν έλεγχο εγκυρότητας του κωδικού πρόσβασης.

2.2. Έλεγχος Πρόσβασης (Cookie Authorization)

Ο έλεγχος πρόσβασης της, εν λόγω, εφαρμογής είναι cookie based. Συγκεκριμένα, κατά την είσοδο ενός χρήστη στο σύστημα, δημιουργείται ένα cookie, το οποίο περιλαμβάνει κάποιες πληροφορίες, όπως το username και ο ρόλος του και αποστέλλεται στον client. Συνεπώς, αν ο client δεν έχει το παραπάνω cookie, ο έλεγχος πρόσβασης αποτυγχάνει. Επιπλέον, το cookie διαγράφεται κατά την έξοδο του χρήστη από την εφαρμογή, διαφορετικά, μένει ενεργό για 20 λεπτά από την ώρα που δημιουργήθηκε. Ο λειτουργία που περιγράφηκε παραπάνω ανάγεται στην τροποποίηση του αρχείου Program.cs, στο οποίο έχουν προστεθεί όλες οι απαραίτητες δυνατότητες για cookie based authorization. Τώρα, όσον αφορά την μεταφορά των πληροφοριών στο cookie, αφού έχει προηγηθεί η αυθεντικοποίηση του χρήστη, στο αρχείο HomeController.cs φορτώνονται τα δεδομένα του χρήστη στο cookie. Δηλαδή, μετά την επιτυχή είσοδο του χρήστη στην εφαρμογή, το cookie φορτώνεται με τα δεδομένα ακριβώς πριν ανακατευθυνθεί ο χρήστης στην αρχική σελίδα, που αντιστοιχεί στον ρόλο του. Ο τρόπος λειτουργίας του cookie based authorization φαίνεται στην παρακάτω εικόνα (Εικόνα 14).



Εικόνα 14: Cookie based authorization

3. Input Filtering and Validation

3.1. Input Filtering και Validation με Regular Expressions

Για να αποφευχθούν διάφορες επιθέσεις, όπως SQL Injection, αλλά και Cross Site Scripting (XSS), γίνεται φιλτράρισμα όλων των user inputs της εφαρμογής. Συγκεκριμένα μετά από κάθε user input πραγματοποιείται ένας built in έλεγχος, από το framework που χρησιμοποιούμε (ASP.NET MVC) που αποσκοπεί στο validation της εισόδου που πληκτρολόγησε ο χρήστης. Επιπλέον, οι κανόνες του ελέγχου είναι διατυπωμένοι (για κάθε user input ξεχωριστά, οπότε και σε κάθε data model) με regular expressions, επομένως με αυτόν τον τρόπο απορρίπτονται κάποιοι ειδικοί χαρακτήρες, όπως για παράδειγμα το ελληνικό ερωτηματικό (;) ή οι μαθηματικοί τελεστές σύγκρισης (<>), οι οποίοι θα μπορούσαν να χρησιμοποιηθούν από έναν κακόβουλο χρήστη, ώστε να πραγματοποιήσει μια επίθεση στην εφαρμογή. Τέλος, αν ο έλεγχος πραγματοποιηθεί με επιτυχία, εκτελούνται οι προκαθορισμένες ενέργειες για το εκάστοτε user input. Σε κάθε άλλη περίπτωση, όπου για οποιονδήποτε λόγο το input δεν περάσει με επιτυχία τον έλεγχο, εμφανίζεται ένα ενημερωτικό μήνυμα για λανθασμένη είσοδο και ο χρήστης καλείται να το εισάγει ξανά.

3.2. Ειδική Περίπτωση: Cross Site Scripting - XSS

Με τον όρο Cross Site Scripting ή XSS αναφερόμαστε σε μία μορφή επίθεσης, που γίνεται σε διαδικτυακές εφαρμογές και επιτρέπει σε έναν κακόβουλο χρήστη να εγχύσει κώδικα JavaScript σε μία ιστοσελίδα, μέσω των user inputs. Με τη σειρά του, ο κώδικας αυτός θα εκτελεστεί στον browser του χρήστη που θα επιχειρήσει να επισκεφθεί την συγκεκριμένη ιστοσελίδα και έτσι ο επιτιθέμενος μπορεί να τροποποιήσει το περιεχόμενο της σελίδας, να κλέψει την ταυτότητα του χρήστη κ.α. Έτσι, είναι απαραίτητη λήψη ειδικών μέτρων για την προστασία των εφαρμογών έναντι των XSS επιθέσεων. Το framework ASP.NET παρέχει την αυτόματη προστασία των εφαρμογών του, μέσω της κωδικοποίησης των ειδικών χαρακτήρων της JavaScript (<>, ' ') στο σύστημα ASCII. Ένα παράδειγμα κωδικοποίησης φαίνεται στην παρακάτω εικόνα (Εικόνα 15).

Class	Method	Results
HTMLHelper	HTMLEncode(...)	In <script>alert('XSS')</script>
		Out <script>alert('XSS')</script>
	HTMLDecode(...)	In <script>alert('XSS')</script>
		Out <script>alert('XSS')</script>
	EncodeForHtmlAttribute(...)	In "onmouseover=alert(document.cookie)"
		Out "onmouseover='alert(document.cookie)'

Εικόνα 15: Κωδικοποίηση JavaScript από το ASP.NET framework για την πρόληψη XSS επιθέσεων.

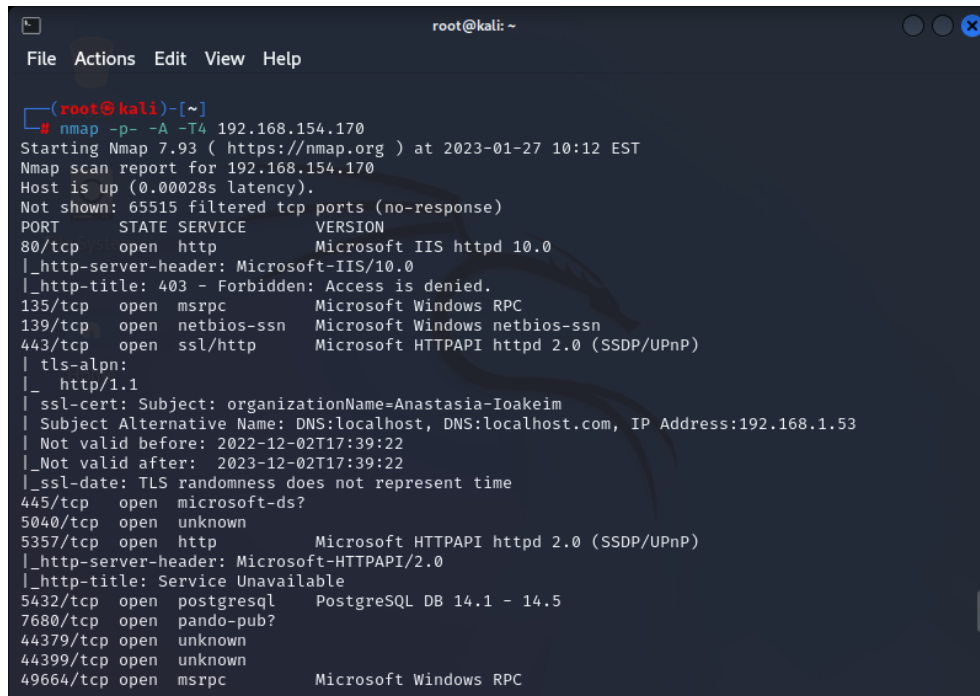
Τέλος, αξίζει να σημειωθεί πως, παρόλο που το ASP.NET παρέχει αυτοματοποιημένη προστασία από XSS επιθέσεις, με το input filtering που έχει προστεθεί στην εφαρμογή και συγκεκριμένα με την απαγόρευση των μαθηματικών τελεστών σύγκρισης (<>) από κάθε user input, μια τέτοιου τύπου επίθεση θα ήταν αδύνατον να πραγματοποιηθεί.

4. Αυτοματοποιημένη Εύρεση Ευπαθειών

4.1. Port Scanning με Nmap

Η Nmap είναι ένα Linux εργαλείο, που αποσκοπεί στην εξερεύνηση δικτύου και στον έλεγχο ασφαλείας μιας εφαρμογής. Στην περίπτωση μας χρησιμοποιήθηκε παραμετροποιημένη (Εικόνες 16,17 & 18) για:

- Ανίχνευση όλων των ανοιχτών θυρών στο δίκτυο που τρέχει η εφαρμογή
- Ανίχνευση λειτουργικού συστήματος, ανίχνευση έκδοσης υπηρεσίας, όπως επίσης script scanning και traceroute
- Πιο σύντομο, χρονικά, scanning, αλλά αξιόπιστο.



```
root@kali: ~  
File Actions Edit View Help  
  
(root@kali)-[~]  
# nmap -p- -A -T4 192.168.154.170  
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-27 10:12 EST  
Nmap scan report for 192.168.154.170  
Host is up (0.00028s latency).  
Not shown: 65515 filtered tcp ports (no-response)  
PORT      STATE SERVICE          VERSION  
80/tcp    open  http             Microsoft IIS httpd 10.0  
|_ http-server-header: Microsoft-IIS/10.0  
|_ http-title: 403 - Forbidden: Access is denied.  
135/tcp   open  msrpc            Microsoft Windows RPC  
139/tcp   open  netbios-ssn      Microsoft Windows netbios-ssn  
443/tcp   open  ssl/http         Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)  
|_ tls-alpn:  
|_ http/1.1  
|_ ssl-cert: Subject: organizationName=Anastasia-Ioakeim  
| Subject Alternative Name: DNS:localhost, DNS:localhost.com, IP Address:192.168.1.53  
| Not valid before: 2022-12-02T17:39:22  
|_ Not valid after: 2023-12-02T17:39:22  
|_ ssl-date: TLS randomness does not represent time  
445/tcp   open  microsoft-ds?      
5040/tcp  open  unknown            
5357/tcp  open  http             Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)  
|_ http-server-header: Microsoft-HTTPAPI/2.0  
|_ http-title: Service Unavailable  
5432/tcp  open  postgresql       PostgreSQL DB 14.1 - 14.5  
7680/tcp  open  pando-pub?        
44379/tcp open  unknown            
44399/tcp open  unknown            
49664/tcp open  msrpc            Microsoft Windows RPC
```

```
root@kali: ~
File Actions Edit View Help
5432/tcp open postgresql PostgreSQL DB 14.1 - 14.5
7680/tcp open pando-pub?
44379/tcp open unknown
44399/tcp open unknown
49664/tcp open msrpc Microsoft Windows RPC
49665/tcp open msrpc Microsoft Windows RPC
49666/tcp open msrpc Microsoft Windows RPC
49667/tcp open msrpc Microsoft Windows RPC
49668/tcp open msrpc Microsoft Windows RPC
49670/tcp open msrpc Microsoft Windows RPC
55354/tcp open http Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-title: Bad Request
|_http-server-header: Microsoft-HTTPAPI/2.0
55358/tcp open http Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-title: Bad Request
|_http-server-header: Microsoft-HTTPAPI/2.0
59935/tcp open msrpc Microsoft Windows RPC
MAC Address: 64:5D:86:45:34:EA (Intel Corporate)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Aggressive OS guesses: Microsoft Windows 10 (96%), Microsoft Windows Server 2008 SP1 (93%), Microsoft Windows 10 1703 (92%), Microsoft Windows 10 1511 - 1607 (92%), Microsoft Windows 10 1607 (91%), Microsoft Windows 10 1511 (91%), Microsoft Windows Server 2008 R2 or Windows 8.1 (91%), Microsoft Windows Server 2016 (91%), Microsoft Windows Vista SP0 or SP1, Windows Server 2008 SP1, or Windows 7 (91%), Microsoft Windows 7 Professional or Windows 8 (90%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 1 hop
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
| smb2-time:
```

```
root@kali: ~
File Actions Edit View Help
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Aggressive OS guesses: Microsoft Windows 10 (96%), Microsoft Windows Server 2008 SP1 (93%), Microsoft Windows 10 1703 (92%), Microsoft Windows 10 1511 - 1607 (92%), Microsoft Windows 10 1607 (91%), Microsoft Windows 10 1511 (91%), Microsoft Windows Server 2008 R2 or Windows 8.1 (91%), Microsoft Windows Server 2016 (91%), Microsoft Windows Vista SP0 or SP1, Windows Server 2008 SP1, or Windows 7 (91%), Microsoft Windows 7 Professional or Windows 8 (90%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 1 hop
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
| smb2-time:
|_ date: 2023-01-27T15:16:52
|_ start_date: N/A
|_ nbstat: NetBIOS name: BOUBLOLAPTOP, NetBIOS user: <unknown>, NetBIOS MAC: 645d864534ea (Intel Corporate)
| smb2-security-mode:
|_ 311:
|_ Message signing enabled but not required

TRACEROUTE
HOP RTT ADDRESS
1 0.28 ms 192.168.154.170

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/.
Nmap done: 1 IP address (1 host up) scanned in 286.25 seconds

(root@kali)-[~]
#
```

Εικόνες 16, 17 & 18: Χρήση της Nmap για ανίχνευση όλων των ανοιχτώ θυρών.

4.2. Certificate Scanning με Nmap

Σε αυτή την περίπτωση η Nmap χρησιμοποιήθηκε για τη σάρωση των SSL πιστοποιητικών στην θύρα που τρέχει η εφαρμογή, με την προσθήκη κάποιων παραμέτρων, όπως πληροφορίες σχετικά με τα πιστοποιητικά και τους αλγορίθμους κρυπτογράφησης που χρησιμοποιούνται. Τα αποτελέσματα απαριθμούν και αξιολογούν τους αλγορίθμους κρυπτογράφησης που χρησιμοποιούνται. Όπως φαίνεται και στις παρακάτω εικόνες (Εικόνες 19 & 20), οι βαθμολογίες των αλγορίθμων είναι από C και πάνω, γεγονός το οποίο, σύμφωνα με το επίσημο documentation της Nmap, θεωρείται ασφαλής. Επιπλέον, αξίζει να σημειωθεί πως δεν υπάρχουν warnings στα αποτελέσματα σχετικά με ευπάθειες.

```
(root@kali)-[~]
# nmap --script ssl-cert,ssl-enum-ciphers -p 443 192.168.154.170
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-27 12:15 EST
Nmap scan report for 192.168.154.170
Host is up (0.00021s latency).

PORT      STATE SERVICE
443/tcp   open  https
| ssl-cert: Subject: organizationName=Anastasia-Ioakeim
| Subject Alternative Name: DNS:localhost, DNS:localhost.com, IP Address:192.168.1.53
| Issuer: commonName=Anastasia-Ioakeim/organizationName=Anastasia-Ioakeim
| Public Key type: rsa
| Public Key bits: 2048
| Signature Algorithm: sha256WithRSAEncryption
| Not valid before: 2022-12-02T17:39:22
| Not valid after: 2023-12-02T17:39:22
| MD5: 4d58a5857f711e9546c18393b5872e87
| SHA-1: e080bbf7de4541d6dfa23ba774dd2436938bd452
| ssl-enum-ciphers:
|   TLSv1.0:
|     ciphers:
|       TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (secp384r1) - A
|       TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (ecdh_x25519) - A
|       TLS_RSA_WITH_AES_256_CBC_SHA (rsa 2048) - A
|       TLS_RSA_WITH_AES_128_CBC_SHA (rsa 2048) - A
|       TLS_RSA_WITH_3DES_EDE_CBC_SHA (rsa 2048) - C
|     compressors:
|       NULL
|     cipher preference: server

|   TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (ecdh_x25519) - A
|   TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 (dh 2048) - A
|   TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 (dh 2048) - A
|   TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 (secp384r1) - A
|   TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 (ecdh_x25519) - A
|   TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (secp384r1) - A
|   TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (ecdh_x25519) - A
|   TLS_RSA_WITH_AES_256_GCM_SHA384 (rsa 2048) - A
|   TLS_RSA_WITH_AES_128_GCM_SHA256 (rsa 2048) - A
|   TLS_RSA_WITH_AES_256_CBC_SHA256 (rsa 2048) - A
|   TLS_RSA_WITH_AES_128_CBC_SHA256 (rsa 2048) - A
|   TLS_RSA_WITH_AES_256_CBC_SHA (rsa 2048) - A
|   TLS_RSA_WITH_AES_128_CBC_SHA (rsa 2048) - A
|   TLS_RSA_WITH_3DES_EDE_CBC_SHA (rsa 2048) - C
|     compressors:
|       NULL
|     cipher preference: server
|     warnings:
|       64-bit block cipher 3DES vulnerable to SWEET32 attack
|   TLSv1.3:
|     ciphers:
|       TLS_AKE_WITH_AES_256_GCM_SHA384 (secp384r1) - A
|       TLS_AKE_WITH_AES_128_GCM_SHA256 (ecdh_x25519) - A
|     cipher preference: server
|   least strength: C
MAC Address: 64:5D:86:45:34:EA (Intel Corporate)

Nmap done: 1 IP address (1 host up) scanned in 0.86 seconds
```

Εικόνες 19 & 20: Αποτελέσματα SSL Scan με την Nmap.

4.3. Security Scanning με Nikto

Το Nikto είναι ένα λογισμικό ανοιχτού κώδικα γραμμένο στη γλώσσα Perl που χρησιμοποιείται για τη σάρωση ενός διακομιστή ιστού για διάφορες ευπάθειες, όπως επικίνδυνα αρχεία, παλιό λογισμικό κ.α. Σαν αποτέλεσμα επιστρέφει τις ευπάθειες που μπορεί να εκμεταλλευτεί ένας επιτιθέμενος ώστε να εισβάλλει στο website. Στην περίπτωση μας, χρησιμοποιήσαμε το Nikto για τη σάρωση του web server IIS/10.0. Τα αποτελέσματα φαίνονται στην παρακάτω εικόνα (Εικόνα 21).

```
(root@kali)~# nikto -h 192.168.154.170
- Nikto v2.1.6

+ Target IP: 192.168.154.170
+ Target Hostname: 192.168.154.170
+ Target Port: 80
+ Start Time: 2023-01-27 10:20:50 (GMT-5)

+ Server: Microsoft-IIS/10.0
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect
  against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the co
  ntent of the site in a different fashion to the MIME type
+ All CGI directories 'found', use '-C none' to test none
+ 26545 requests: 0 error(s) and 3 item(s) reported on remote host
+ End Time: 2023-01-27 10:21:57 (GMT-5) (67 seconds)

+ 1 host(s) tested
```

Εικόνα 21: Αποτελέσματα security scanning με την nikto.

4.4. Επιθέσεις με SQLmap

Το SQLmap είναι ένα εργαλείο ανοιχτού κώδικα που βρίσκει και εκμεταλλεύεται αυτόματα τα τρωτά σημεία για SQL injection σε μια βάση δεδομένων. Μπορεί να χρησιμοποιηθεί για την δοκιμή διαδικτυακών εφαρμογών σε επιθέσεις SQL injection και την απόκτηση πρόσβασης σε μια ευάλωτη βάση δεδομένων. Στην περίπτωση μας χρησιμοποιήθηκε, αρχικά, για να ανακτηθεί η λίστα με τα ονόματα όλων των βάσεων δεδομένων (Εικόνα 22) και έπειτα, για μια δοκιμαστική dump db επίθεση (Εικόνα 23). Παρ 'όλες τις προσπάθειες που πραγματοποιήθηκαν από πλευράς μας για επίθεση στη βάση, τα αποτελέσματα ήταν πολύ καλά και έδειξαν πως δεν είναι ευάλωτη σε τέτοιες επιθέσεις, όπως φαίνεται και στις παρακάτω εικόνες.

```
[11:17:46] [CRITICAL] all tested parameters do not appear to be injectable. Try to increase valu
es for '--level'/'--risk' options if you wish to perform more tests. If you suspect that there i
s some kind of protection mechanism involved (e.g. WAF) maybe you could try to use option '--tam
per' (e.g. '--tamper=space2comment') and/or switch '--random-agent'
[11:17:46] [WARNING] HTTP error codes detected during run:
403 (Forbidden) - 11062 times

[*] ending @ 11:17:46 /2023-01-27/
```

Εικόνα 22: Χρήση της sqlmap για την ανάκτηση της λίστας με ονόματα των διαθέσιμων βάσεων δεδομένων.

```
[11:38:53] [CRITICAL] all tested parameters do not appear to be injectable. Try to increase values for '--level'/'--risk' options if you wish to perform more tests. If you suspect that there is some kind of protection mechanism involved (e.g. WAF) maybe you could try to use option '--tamper' (e.g. '--tamper=space2comment') and/or switch '--random-agent'
[11:38:53] [WARNING] HTTP error codes detected during run:
403 (Forbidden) - 11062 times

[*] ending @ 11:38:53 /2023-01-27/
```

Εικόνα 23: Χρήση της sqlmap για πραγματοποίηση δοκιμαστικής επίθεσης dump db.

4.4. TLS/SSL Scanning με SSLScan

Το SSLScan είναι ένα εργαλείο γραμμής εντολών που εκτελεί μια μεγάλη ποικιλία δοκιμαστικών ελέγχων σε μια καθορισμένη ιστοσελίδα και επιστρέφει μια ολοκληρωμένη λίστα των πρωτοκόλλων και των αλγορίθμων κρυπτογράφησης που γίνονται δεκτά από έναν διακομιστή SSL/TLS μαζί με κάποιες άλλες χρήσιμες, πληροφορίες σε μια δοκιμή ασφαλείας. Τα αποτελέσματα των ελέγχων (Εικόνες 23, 24 & 25) είναι εύκολα στην ανάγνωση και την κατανόηση καθώς είναι σε χρωματιστή μορφή, ανάλογα με τη σοβαρότητα. Δηλαδή, το κόκκινο υποδηλώνει μη ασφαλή διαμόρφωση (επιτρέποντας το SSLv3 και τη χρήση κρυπτογράφησης DES και RC4 - πορτοκαλί) ενώ το πράσινο ή το λευκό συνιστάται.


```

(root@kali)-[~]
# sslscan https://192.168.154.170
Version: 2.0.15-static
OpenSSL 1.1.1q-dev xx XXX xxxx

Connected to 192.168.154.170

Testing SSL server 192.168.154.170 on port 443 using SNI name 192.168.154.170

SSL/TLS Protocols:
SSLv2      disabled
SSLv3      disabled
TLSv1.0    enabled
TLSv1.1    enabled
TLSv1.2    enabled
TLSv1.3    enabled

TLS Fallback SCSV:
Server does not support TLS Fallback SCSV

TLS renegotiation:
Secure session renegotiation supported

TLS Compression:
Compression disabled

Heartbleed:
TLSv1.3 not vulnerable to heartbleed
TLSv1.2 not vulnerable to heartbleed
TLSv1.1 not vulnerable to heartbleed
TLSv1.0 not vulnerable to heartbleed

```

```

Supported Server Cipher(s):
Preferred TLSv1.3 256 bits TLS_AES_256_GCM_SHA384 Curve 25519 DHE 253
Accepted TLSv1.3 128 bits TLS_AES_128_GCM_SHA256 Curve 25519 DHE 253
Preferred TLSv1.2 256 bits ECDHE-RSA-AES256-GCM-SHA384 Curve P-384 DHE 384
Accepted TLSv1.2 128 bits ECDHE-RSA-AES128-GCM-SHA256 Curve 25519 DHE 253
Accepted TLSv1.2 256 bits DHE-RSA-AES256-GCM-SHA384 DHE 2048 bits
Accepted TLSv1.2 128 bits DHE-RSA-AES128-GCM-SHA256 DHE 2048 bits
Accepted TLSv1.2 256 bits ECDHE-RSA-AES256-SHA384 Curve P-384 DHE 384
Accepted TLSv1.2 128 bits ECDHE-RSA-AES128-SHA256 Curve 25519 DHE 253
Accepted TLSv1.2 256 bits ECDHE-RSA-AES256-SHA Curve P-384 DHE 384
Accepted TLSv1.2 128 bits ECDHE-RSA-AES128-SHA Curve 25519 DHE 253
Accepted TLSv1.2 256 bits AES256-GCM-SHA384
Accepted TLSv1.2 128 bits AES128-GCM-SHA256
Accepted TLSv1.2 256 bits AES256-SHA256
Accepted TLSv1.2 128 bits AES128-SHA256
Accepted TLSv1.2 256 bits AES256-SHA
Accepted TLSv1.2 128 bits AES128-SHA
Accepted TLSv1.2 112 bits DES-CBC3-SHA
Preferred TLSv1.1 256 bits ECDHE-RSA-AES256-SHA Curve P-384 DHE 384
Accepted TLSv1.1 128 bits ECDHE-RSA-AES128-SHA Curve 25519 DHE 253
Accepted TLSv1.1 256 bits AES256-SHA
Accepted TLSv1.1 128 bits AES128-SHA
Accepted TLSv1.1 112 bits DES-CBC3-SHA
Preferred TLSv1.0 256 bits ECDHE-RSA-AES256-SHA Curve P-384 DHE 384
Accepted TLSv1.0 128 bits ECDHE-RSA-AES128-SHA Curve 25519 DHE 253
Accepted TLSv1.0 256 bits AES256-SHA
Accepted TLSv1.0 128 bits AES128-SHA
Accepted TLSv1.0 112 bits DES-CBC3-SHA

```

```

Server Key Exchange Group(s):
TLSv1.3 128 bits secp256r1 (NIST P-256)
TLSv1.3 192 bits secp384r1 (NIST P-384)
TLSv1.3 128 bits x25519
TLSv1.2 128 bits secp256r1 (NIST P-256)
TLSv1.2 192 bits secp384r1 (NIST P-384)
TLSv1.2 128 bits x25519

SSL Certificate:
Signature Algorithm: sha256WithRSAEncryption
RSA Key Strength: 2048

Subject: /O=Anastasia-Ioakeim
AltNames: DNS:localhost, DNS:localhost.com, IP Address:192.168.1.53
Issuer: Anastasia-Ioakeim

Not valid before: Dec 2 17:39:22 2022 GMT
Not valid after: Dec 2 17:39:22 2023 GMT

```

Εικόνες 23, 24 & 25: Αποτελέσματα δοκιμαστικών ελέγχων με την *sslscan*.

4.5. Ολικό Scanning με SSLyze

Το SSLyze είναι ένα εργαλείο γραμμής εντολών που σαρώνει μια προκαθορισμένη ιστοσελίδα για SSL/TLS ευπάθειες όπως heartbleed, OpenSSL κ.α. και επιστρέφει μια ολοκληρωμένη αναφορά ασφάλειας για διάφορες επιθέσεις. Όπως φαίνεται και στις παρακάτω εικόνες, τα αποτελέσματα από τον έλεγχο SSLyze για την εφαρμογή μας ήταν πολύ καλά, καθώς η εφαρμογή δεν είναι ευπαθής στις γνωστές επιθέσεις (Εικόνες 26, 27, 28, 29, 30 & 31).

```
(root@kali)~# sslyze 192.168.154.170:443

CHECKING CONNECTIVITY TO SERVER(S)

192.168.154.170:443      => 192.168.154.170

SCAN RESULTS FOR 192.168.154.170:443 - 192.168.154.170

* Certificates Information:
  Hostname sent for SNI:      192.168.154.170
  Number of certificates detected: 1

Certificate #0 ( _RSAPublicKey )
  SHA1 Fingerprint:          e080bbf7de4541d6dfa23ba774dd2436938bd452
  Common Name:               O=Anastasia-Ioakeim
  Issuer:                    Anastasia-Ioakeim
  Serial Number:             2
  Not Before:                2022-12-02
  Not After:                 2023-12-02
  Public Key Algorithm:      _RSAPublicKey
  Signature Algorithm:       sha256
  Key Size:                  2048

  Exponent:                  65537
  DNS Subject Alternative Names: ['localhost', 'localhost.com']

Certificate #0 - Trust
  Hostname Validation:       FAILED - Certificate does NOT match server hostname
  Android CA Store (13.0.0_r8): FAILED - Certificate is NOT Trusted: unable to get local issuer certificate
  Apple CA Store (iOS 15.1, iPadOS 15.1, macOS 12.1, tvOS 15.1, and watchOS 8.1): FAILED - Certificate is NOT Trusted: unable to get local issuer certificate
  Java CA Store (jdk-13.0.2): FAILED - Certificate is NOT Trusted: unable to get local issuer certificate
  Mozilla CA Store (2022-09-18): FAILED - Certificate is NOT Trusted: unable to get local issuer certificate
  Windows CA Store (2022-08-15): FAILED - Certificate is NOT Trusted: unable to get local issuer certificate
  Symantec 2018 Deprecation: ERROR - Could not build verified chain (certificate untrusted?)
  Received Chain:            O=Anastasia-Ioakeim
  Verified Chain:            ERROR - Could not build verified chain (certificate untrusted?)
  Received Chain Contains Anchor: ERROR - Could not build verified chain (certificate untrusted?)
  Received Chain Order:      OK - Order is valid
  Verified Chain contains SHA1: ERROR - Could not build verified chain (certificate untrusted?)

Certificate #0 - Extensions
  OCSP Must-Staple:          NOT SUPPORTED - Extension not found
  Certificate Transparency:   NOT SUPPORTED - Extension not found
```

```

Certificate #0 - OCSP Stapling
nse                                NOT SUPPORTED - Server did not send back an OCSP respo

* SSL 2.0 Cipher Suites:
  Attempted to connect using 7 cipher suites; the server rejected all cipher suites.

* SSL 3.0 Cipher Suites:
  Attempted to connect using 80 cipher suites; the server rejected all cipher suites.

* TLS 1.0 Cipher Suites:
  Attempted to connect using 80 cipher suites.

  The server accepted the following 5 cipher suites:
    TLS_RSA_WITH_AES_256_CBC_SHA           256
    TLS_RSA_WITH_AES_128_CBC_SHA          128
    TLS_RSA_WITH_3DES_EDE_CBC_SHA         168
    TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA    256      ECDH: secp384r1 (384 bits)
    TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA    128      ECDH: prime256v1 (256 bits)

  The group of cipher suites supported by the server has the following properties:
    Forward Secrecy                        OK - Supported
    Legacy RC4 Algorithm                   OK - Not Supported

* TLS 1.1 Cipher Suites:
  Attempted to connect using 80 cipher suites.

  The server accepted the following 5 cipher suites:
    TLS_RSA_WITH_AES_256_CBC_SHA          256

    TLS_RSA_WITH_AES_128_CBC_SHA          128
    TLS_RSA_WITH_3DES_EDE_CBC_SHA         168
    TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA    256      ECDH: secp384r1 (384 bits)
    TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA    128      ECDH: prime256v1 (256 bits)

  The group of cipher suites supported by the server has the following properties:
    Forward Secrecy                        OK - Supported
    Legacy RC4 Algorithm                   OK - Not Supported

* TLS 1.2 Cipher Suites:
  Attempted to connect using 156 cipher suites.

  The server accepted the following 15 cipher suites:
    TLS_RSA_WITH_AES_256_GCM_SHA384       256
    TLS_RSA_WITH_AES_256_CBC_SHA256       256
    TLS_RSA_WITH_AES_256_CBC_SHA          256
    TLS_RSA_WITH_AES_128_GCM_SHA256       128
    TLS_RSA_WITH_AES_128_CBC_SHA256       128
    TLS_RSA_WITH_AES_128_CBC_SHA          128
    TLS_RSA_WITH_3DES_EDE_CBC_SHA         168
    TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 256      ECDH: secp384r1 (384 bits)
    TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 256      ECDH: secp384r1 (384 bits)
    TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA    256      ECDH: secp384r1 (384 bits)
    TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 128      ECDH: prime256v1 (256 bits)
    TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 128      ECDH: prime256v1 (256 bits)
    TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA    128      ECDH: prime256v1 (256 bits)
    TLS_DHE_RSA_WITH_AES_256_GCM_SHA384   256      DH (2048 bits)
    TLS_DHE_RSA_WITH_AES_128_GCM_SHA256   128      DH (2048 bits)

```

```

      TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA      128      ECDH: prime256v1 (256 bits)
      TLS_DHE_RSA_WITH_AES_256_GCM_SHA384     256      DH (2048 bits)
      TLS_DHE_RSA_WITH_AES_128_GCM_SHA256     128      DH (2048 bits)

The group of cipher suites supported by the server has the following properties:
  Forward Secrecy      OK - Supported
  Legacy RC4 Algorithm  OK - Not Supported

* TLS 1.3 Cipher Suites:
  Attempted to connect using 5 cipher suites.

The server accepted the following 2 cipher suites:
      TLS_AES_256_GCM_SHA384      256      ECDH: X25519 (253 bits)
      TLS_AES_128_GCM_SHA256     128      ECDH: X25519 (253 bits)

* Deflate Compression:
                                     OK - Compression disabled

* OpenSSL CCS Injection:
                                     OK - Not vulnerable to OpenSSL CCS injection

* OpenSSL Heartbleed:
                                     OK - Not vulnerable to Heartbleed

* ROBOT Attack:
                                     OK - Not vulnerable.

* Session Renegotiation:
  Client Renegotiation DoS Attack:  OK - Not vulnerable

Secure Renegotiation:      OK - Supported

* Elliptic Curve Key Exchange:
  Supported curves:          X25519, prime256v1, secp384r1
  Rejected curves:          X448, prime192v1, secp160k1, secp160r1, secp160r2, sec
p192k1, secp224r1, secp256k1, secp521r1, sect163k1, sect163r1, sect163r2, sect193r1,
sect193r2, sect233k1, sect233r1, sect239k1, sect283k1, sect283r1, sect409k1, sect409r1, sect571k
1, sect571r1

SCANS COMPLETED IN 1.358472 S

COMPLIANCE AGAINST MOZILLA TLS CONFIGURATION

Checking results against Mozilla's "intermediate" configuration. See https://ssl-config.mozila.org/ for more details.

192.168.154.170:443: FAILED - Not compliant.
  * certificate_hostname_validation: Certificate hostname validation failed for O=Anastasi
a-Ioakeim.
  * certificate_path_validation: Certificate path validation failed for O=Anastasia-Ioakei
m.
  * tls_versions: TLS versions {'TLSv1', 'TLSv1.1'} are supported, but should be rejected.
  * ciphers: Cipher suites {'TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256', 'TLS_ECDHE_RSA_WITH_A
ES_256_CBC_SHA', 'TLS_RSA_WITH_AES_128_CBC_SHA256', 'TLS_RSA_WITH_AES_256_CBC_SHA', 'TLS_RSA_WIT
H_AES_128_CBC_SHA', 'TLS_RSA_WITH_AES_256_GCM_SHA384', 'TLS_RSA_WITH_AES_128_GCM_SHA256', 'TLS_E
CDHE_RSA_WITH_AES_128_CBC_SHA', 'TLS_RSA_WITH_AES_256_CBC_SHA256', 'TLS_RSA_WITH_3DES_EDE_CBC_SH
A', 'TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384'} are supported, but should be rejected.

```

Εικόνες 26, 27, 28, 29, 30 & 31: Αποτελέσματα scanning με την sslyze.