BOZP PORTAL PENETRATION TEST REPORT

Elnar Yantay, Yaroslav Bakhirkin

yantaeln@fit.cvut.cz, bakhiyar@fit.cvut.cz

Table of Contents

Team Info 3
Project overview 4
Scope description 5
Selected pentesting methodology 6
Scoring system description 7
Threat model 11
Intelligence-gathering outcomes 13
List of findings 15
Executive summary 15
Pentesting process 16
- Information Gathering 16
- Configuration and Deployment Management Testing 22
- Identity Management Testing 23
- Authentication testing 23
- Authorization Testing 26
- Session Management Testing 28
- Testing for Error Handling 28
- Input Validation Testing 29

Vulnerability analysis 31
- Unencrypted transfer of credentials 31
- Using an outdated server 32
- CSRF – Homepage 33
- Improper Access Control 35
- Presence of default files 37
- Server shut down 39

Team Info

- Elnar Yantay
  - Student of the 2nd year of the Information Security program
  - During this project he was responsible for the following tests:
    - Information gathering
    - Configuration and Deployment Management Testing
    - Identity Management Testing
    - Authorization Testing
    - Session Management Testing
    - Vulnerability Analysis

- Yaroslav Bakhirkin
  - Student of the 2nd year of the Information Security program
  - During this project he was responsible for the following tests:
    - Information gathering
    - Authentication Testing
    - Testing for Error Handling
    - Input Validation Testing
    - Vulnerability Analysis

Project overview

In this semester's work, we will focus on the penetration test of the BOZP portal website as a part of Internal Applications for Testing.

Occupational safety and health is a critical aspect of any academic institution, ensuring the well-being of students, faculty, and staff. The BOZP portal serves as a central platform for students to access relevant materials, undertake online assessments and register for events related to occupational safety.

Scope description

The penetration testing scope covers the BOZP portal, excluding the production version. Testing will be conducted on a provided virtual machine with anonymized student data, static ip address 10.0.0.10 and two users - “student” as regular user and “admin” as user with administrator privilege.

The objectives include identifying security vulnerabilities, assessing resilience against cyber threats, evaluating authentication mechanisms, and providing recommendations for remediation.

Selected pentesting methodology

We utilized the OWASP Web Security Testing Guide as the selected penetration testing and report writing methodology for assessing the BOZP Portal.

Scoring system description

For rating risks, we will use OWASP Risk Rating Methodology.

To find out how risky a vulnerability is, we need to figure out two things: how much it can affect things and how likely it is to happen. These things help us measure what might happen if the vulnerability is used and how probably it is to occur.

Likelihood factors.

The first step is to estimate the "likelihood". There are a number of factors that can help determine the likelihood. The first set of factors are related to the threat agent involved. The goal is to estimate the likelihood of a successful attack from a possible attackers. The next set of factors are related to the vulnerability involved. The goal here is to estimate the likelihood of the particular vulnerability involved being discovered and exploited. Each option has a likelihood rating from 0 to 9 associated with it. These numbers will be used later to estimate the overall likelihood.

Threat Agent Factors:

- Skill Level (How technically skilled are threat agents?)
    - No technical skills (1).
    - Some technical skills (3).
    - Advanced computer user (5).
    - Network and programming skills (6).
    - Security penetration skills (9).
- Motive (How motivated are threat agents?)
    - Low or no reward (1).
    - Possible reward (4).
    - High reward (9).
- Opportunity (What resources and opportunities are required for threat agents?)
    - Full access or expensive resources required (0).
    - Special access or resources required (4).
    - Some access or resources required (7).
    - No access or resources required (9).

- Size (How large is this group of threat agents?)
    - Developers (2).
    - System administrators (2).
    - Intranet users (4).
    - Partners (5).
    - Authenticated users (6).
    - Anonymous internet users (9).

Vulnerability factors:

- Ease of Discovery (How easy is it for threat agents to discover this vulnerability?)
    - Practically impossible (1).
    - Difficult (3).
    - Easy (7).
    - Automated tools available (9).
- Ease of Exploit (How easy is it for threat agents to actually exploit this vulnerability?)
    - Theoretical (1).
    - Difficult (3).
    - Easy (5).
    - Automated tools available (9).
- Awareness (How well known is this vulnerability to threat agents?)
    - Unknown (1).
    - Hidden (4).
    - Obvious (6).
    - Public knowledge (9).
- Intrusion Detection (How likely is an exploit to be detected?)
    - Active detection in application (1).
    - Logged and reviewed (3).
    - Logged without review (8).
    - Not logged (9).

Estimating Impact.

Technical impact can be broken down into factors aligned with the traditional security areas of concern: confidentiality, integrity, availability, and accountability. The goal is to guess how much the system would be affected if the weakness was used.

Technical Impact Factors:

- Loss of Confidentiality (how much data could be disclosed and how sensitive is it?)
    - Minimal non-sensitive data disclosed (2).
    - Minimal critical data disclosed (6).
    - Extensive non-sensitive data disclosed (6).
    - Extensive critical data disclosed (7).
    - All data disclosed (9).
- Loss of Integrity (how much data could be corrupted and how damaged is it?)
    - Minimal slightly corrupt data (1).
    - Minimal seriously corrupt data (3).
    - Extensive slightly corrupt data (5).
    - Extensive seriously corrupt data (7).
    - All data totally corrupt (9).
- Loss of Availability (how much service could be lost and how vital is it?)
    - Minimal secondary services interrupted (1).
    - Minimal primary services interrupted (5).
    - Extensive secondary services interrupted (5).
    - Extensive primary services interrupted (7).
    - All services completely lost (9).
- Loss of Accountability (Are the threat agents' actions traceable to an individual?)
    - Fully traceable (1).
    - Possibly traceable (7).
    - Completely anonymous (9).

Determining the Severity of the Risk.

To determine the impact and likelihood, we calculate the average of all factors for each.

Likelihood and Impact Levels	
0 to < 3	LOW
3 to < 6	MEDIUM
6 to 9	HIGH

Overall Risk Severity				
Impact	HIGH	Medium	High	Critical
	MEDIUM	Low	Medium	High
	LOW	Note	Low	Medium
		LOW	MEDIUM	HIGH
	Likelihood			

Threat model

BOZP portal is not very useful resource to hack, but there still are some goals attacker might want to achieve, such as:

- Obtaining administrator permissions to change roles/add events/etc.
- Disrupt the application to sabotage university's "Occupational safety and health".
- Get access to some student's events to know where and when this particular student will be.

User access level	
Guest	No account, no permissions.
Student	Access available events and see events student is signed for.
Administrator	"root" user – admin access, adding/deleting events/users etc.

Assets	
Name	*Description*
Student	
Events	Events from list of available events student is signed up for.
Administrator	
Events	Date, time, description and participators of the event.
Courses	Each course can have an event assigned to it.
Connection	Connection to KOS/SQL database.
Documents	Documents related to safety all students/teachers must sign.

Potential threats			
Threat	*Description*	*STRIDE*	*Impact*
Unauthorized access	Student obtains other student's event management.	E	Low
Unauthorized access	Student obtains admin's privileges	E	High
Repudiation	Student denies changing data/gaining access etc.	R	Medium
Data tampering	Student modifies data available only to admin.	T	High
Unauthorized disclosure	Guest gains access to student's events.	I	Low

Intelligence-gathering outcomes

```
# Nmap 7.94SVN scan initiated Sat Apr 27 04:31:32 2024 as: nmap -sV --top-ports 100 -A -O -oN result.txt 10.0.0.10
Nmap scan report for 10.0.0.10
Host is up (0.00022s latency).
Not shown: 97 closed tcp ports (reset)
PORT     STATE SERVICE VERSION
22/tcp   open  ssh     OpenSSH 7.9p1 Debian 10+deb10u4 (protocol 2.0)
| ssh-hostkey:
|   2048 b1:e8:1e:43:09:e3:1a:d9:03:9f:63:6f:fc:bc:ae:95 (RSA)
|   256 a9:73:07:7d:83:e7:ab:85:c7:22:f7:00:ef:77:73:59 (ECDSA)
|   256 58:4b:84:8f:91:70:84:db:12:6b:38:68:35:59:7f:28 (ED25519)
80/tcp   open  http    Apache httpd 2.4.38 ((Debian))
| http-title: Aktuality | BOZP port\xC3\xA1l
|_Requested resource was http://10.0.0.10/www/
| http-server-header: Apache/2.4.38 (Debian)
3306/tcp open  mysql   MySQL (unauthorized)
MAC Address: 08:00:27:CE:6A:11 (Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 4.X|5.X
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5
OS details: Linux 4.15 - 5.8
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE
HOP RTT     ADDRESS
1   0.22 ms 10.0.0.10

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Sat Apr 27 04:32:29 2024 -- 1 IP address (1 host up) scanned in 57.12 seconds
```

*These are the ports and services used as well as its versions found during the **nmap** scan.*

```
HTTP/1.1 200 OK
Date: Sat, 27 Apr 2024 08:42:14 GMT
Server: Apache/2.4.38 (Debian)
X-Powered-By: Nette Framework
X-Frame-Options: SAMEORIGIN
Set-Cookie: PHPSESSID=khpm37vkqpkcljap87351ia4lq; expires=Sat, 11-May-2024 08:42:14 GMT; Max-Age=1209600; path=/; HttpOnly
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Vary: X-Requested-With,Accept-Encoding
Set-Cookie: PHPSESSID=6nh1sumrmkg6g6pmuh8dgq02h0; expires=Sat, 11-May-2024 08:42:14 GMT; Max-Age=1209600; path=/; HttpOnly
Set-Cookie: PHPSESSID=6nh1sumrmkg6g6pmuh8dgq02h0; expires=Sat, 11-May-2024 08:42:14 GMT; Max-Age=1209600; path=/; HttpOnly
Content-Length: 4312
Connection: close
Content-Type: text/html; charset=utf-8
```

*Web server is indeed **Apache**(2.4.38), and there is a frontend PHP framework **Nette** running.*

```
┌──(kali㉿kali)-[~]
└─$ nikto -h 10.0.0.10
- Nikto v2.5.0
---------------------------------------------------------------------------
+ Target IP:          10.0.0.10
+ Target Hostname:    10.0.0.10
+ Target Port:        80
+ Start Time:         2024-04-27 07:48:40 (GMT-4)
---------------------------------------------------------------------------
+ Server: Apache/2.4.38 (Debian)
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Fram
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fas
anner/vulnerabilities/missing-content-type-header/
+ Root page / redirects to: http://10.0.0.10/www/
+ Apache/2.4.38 appears to be outdated (current is at least Apache/2.4.54). Apache 2.2.34 is the EOL for the 2.x branch.
+ /www/: Retrieved x-powered-by header: Nette Framework.
+ /www/: This might be interesting.
+ /icons/README: Apache default file found. See: https://www.vntweb.co.uk/apache-restricting-access-to-iconsreadme/
+ /composer.json: PHP Composer configuration file reveals configuration information. See: https://getcomposer.org/
+ /composer.lock: PHP Composer configuration file reveals configuration information. See: https://getcomposer.org/
+ /.gitignore: .gitignore file found. It is possible to grasp the directory structure.
+ 9661 requests: 0 error(s) and 9 item(s) reported on remote host
+ End Time:           2024-04-27 07:49:05 (GMT-4) (25 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
```

Nikto output

List of findings

Type	Location	Impact	Likelihood	Severity	Link
Unsecured credentials transfer	Log in form	9.0	8.9	Critical	Analysis
Using an outdated server	All application	8	7	High	Analysis
CSRF	Homepage	6.5	3.1	High	Analysis
Improper Access Control	User's profile	4	5.3	Medium	Analysis
Presence of default files	All application	1.75	3.5	Low	Analysis
Server error	Manage events	0.3	0.8	Note	Analysis

Executive summary

Overall, application is insecure and has some crucial vulnerabilities. For application to become somewhat secure, it should fix unencrypted data transfer, which is not that hard.

Unencrypted data transfer

The largest vulnerability allows attacker to easily get access to user's credentials, and, considering that application logs you our every now and then, this vulnerability is indeed critical.

Using an outdated server

Outdated software versions are a popular target for attackers because information about them is available in public vulnerability databases.

CSRF – Homepage

Even though it is not easy to determine the exact web request to delete homepage's articles because it isn't done as often and admin user may be aware of running suspicious code, there is a possibility to remove articles by running premade HTML file in admin's session.

Improper Access Control

The vulnerability occurs when the system does not properly control or restrict access to resources based on user rights. As a result, users may access data or perform actions that they do not have permission to do.

Presence of default files

During testing, the presence of default files was discovered, such as /icons/README. These files are often left behind after installing a web server or web application and may contain information about the configuration or structure of the system.

Server shut down – Manage events

This is not a big problem as it is can be done only manually by admin, but still this error persists until fixed and may be inconvenient for some users.

Pentesting process

Information Gathering

Fingerprint Web Server

Server Apache/2.4.38 has been discovered. More information in this section.

Review webserver metafiles for information leakage

After examining the website, some default files were found. More information in this section.

```
┌──(kali㉿kali)-[~]
└─$ nikto -h 10.0.0.10
- Nikto v2.5.0
---------------------------------------------------------------------------
+ Target IP:          10.0.0.10
+ Target Hostname:    10.0.0.10
+ Target Port:        80
+ Start Time:         2024-04-27 07:48:40 (GMT-4)
---------------------------------------------------------------------------
+ Server: Apache/2.4.38 (Debian)
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Fram
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fas
anner/vulnerabilities/missing-content-type-header/
+ Root page / redirects to: http://10.0.0.10/www/
+ Apache/2.4.38 appears to be outdated (current is at least Apache/2.4.54). Apache 2.2.34 is the EOL for the 2.x branch.
+ /www/: Retrieved x-powered-by header: Nette Framework.
+ /www/: This might be interesting.
+ /icons/README: Apache default file found. See: https://www.vntweb.co.uk/apache-restricting-access-to-iconsreadme/
+ /composer.json: PHP Composer configuration file reveals configuration information. See: https://getcomposer.org/
+ /composer.lock: PHP Composer configuration file reveals configuration information. See: https://getcomposer.org/
+ /.gitignore: .gitignore file found. It is possible to grasp the directory structure.
+ 9681 requests: 0 error(s) and 9 item(s) reported on remote host
+ End Time:           2024-04-27 07:49:05 (GMT-4) (25 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
```

Enumerate applications on webserver

Nmap result :

```
┌──(kali㉿kali)-[~]
└─$ nmap -Pn -sT -sV -p0-65535 10.0.0.10
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-06-07 19:47 EDT
Nmap scan report for 10.0.0.10
Host is up (0.0032s latency).
Not shown: 65532 closed tcp ports (conn-refused)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh     OpenSSH 7.9p1 Debian 10+deb10u4 (protocol 2.0)
80/tcp    open  http    Apache httpd 2.4.38 ((Debian))
3306/tcp  open  mysql   MySQL (unauthorized)
33060/tcp open  mysqlx?
1 service unrecognized despite returning data. If you know the service/version, please submit the following fingerprint at https://nmap.org/cgi-bin/submit.cg
SF-Port33060-TCP:V=7.94SVN%I=7%D=6/7%Time=66639C31%P=x86_64-pc-linux-gnu%r
SF:(GenericLines,9,"\x05\0\0\0\x0b\x08\x05\x1a\0");
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 202.63 seconds
```

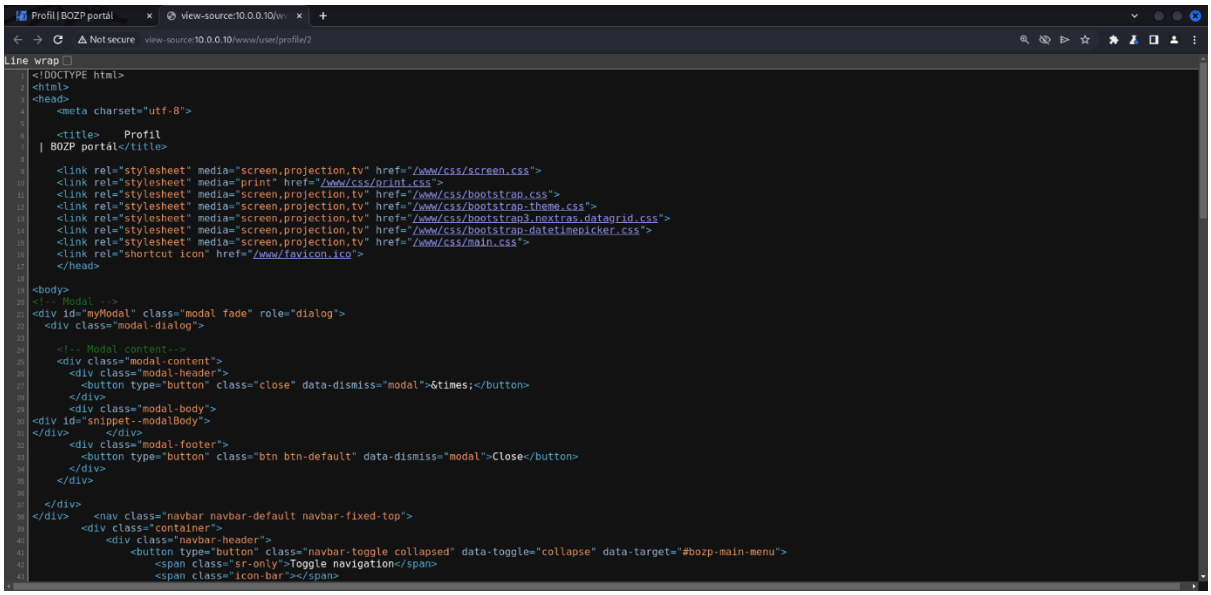
nmap -Pn -sT -sV -p0-65535 10.0.0.10

From the output of the Nmap command, these ports were found:

- 22/tcp open ssh : OpenSSH 7.9p1 Debian 10+deb10u4 (protocol 2.0)
- 80/tcp open http : Apache httpd 2.4.38 ((Debian))
- 3306/tcp open mysql : MySQL (unauthorized)
- 33060/tcp open mysqlx?

Review Webpage Content for Information Leakage

The source code of all possible pages of the portal was studied and found:



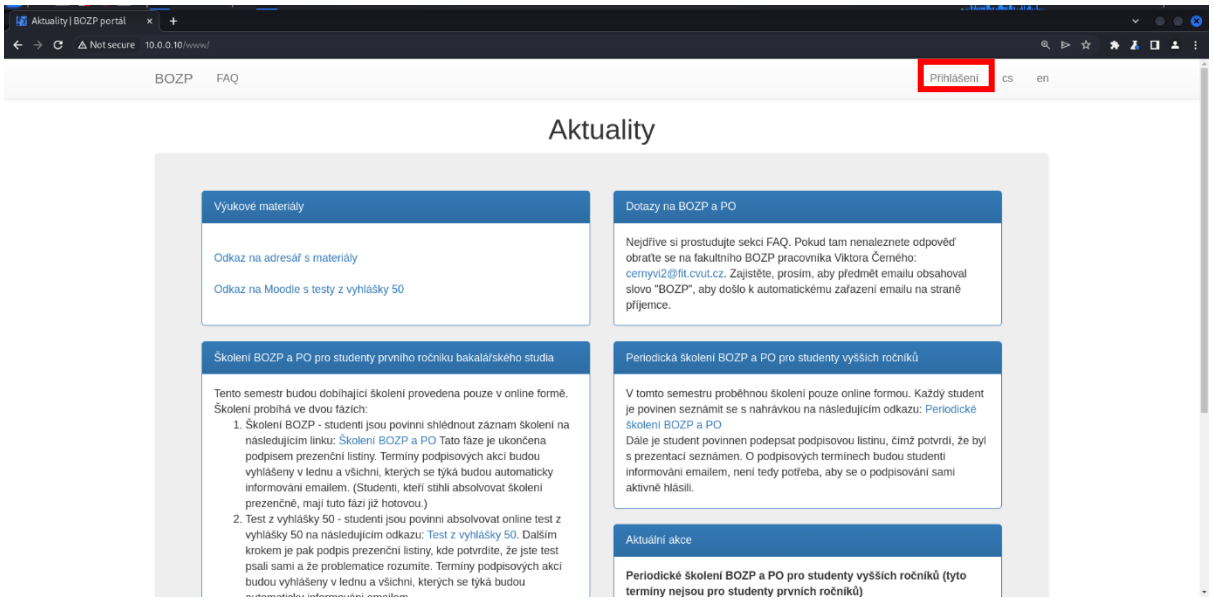
Meta tags

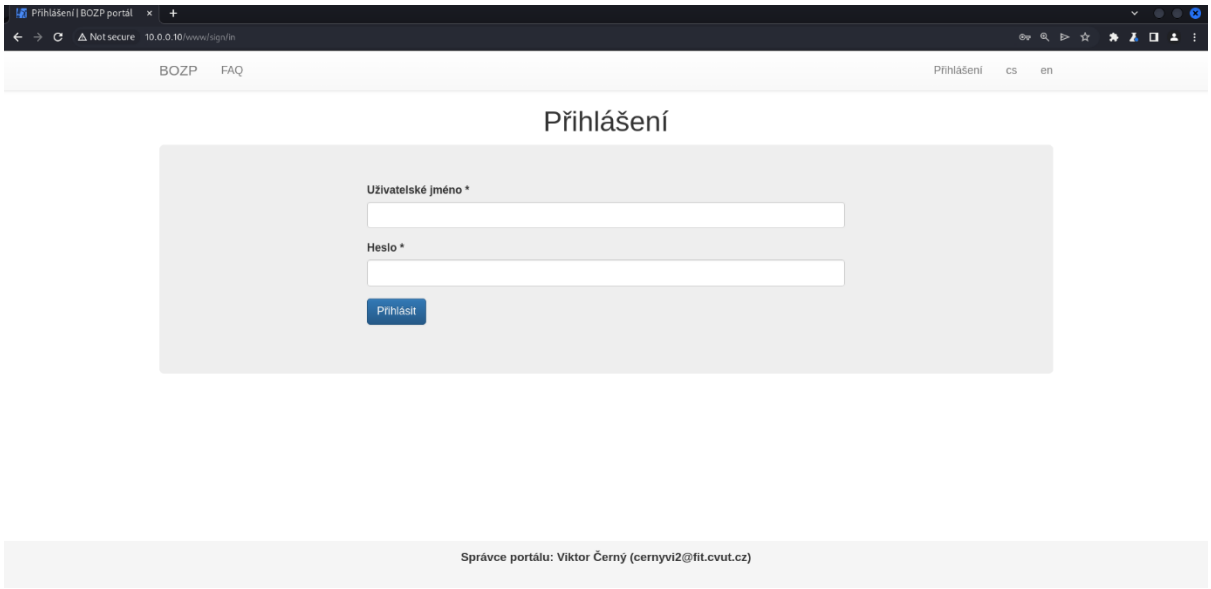
<meta charset="utf-8">

Other interesting meta data or comments were not found in the source code.

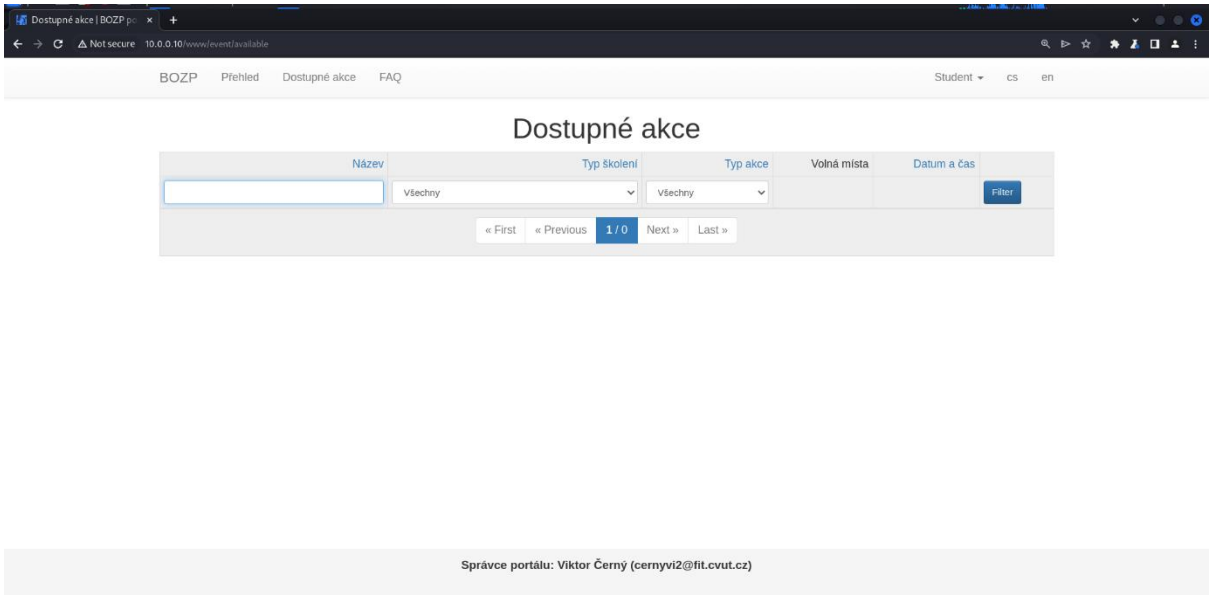
Identify Application Entry Points

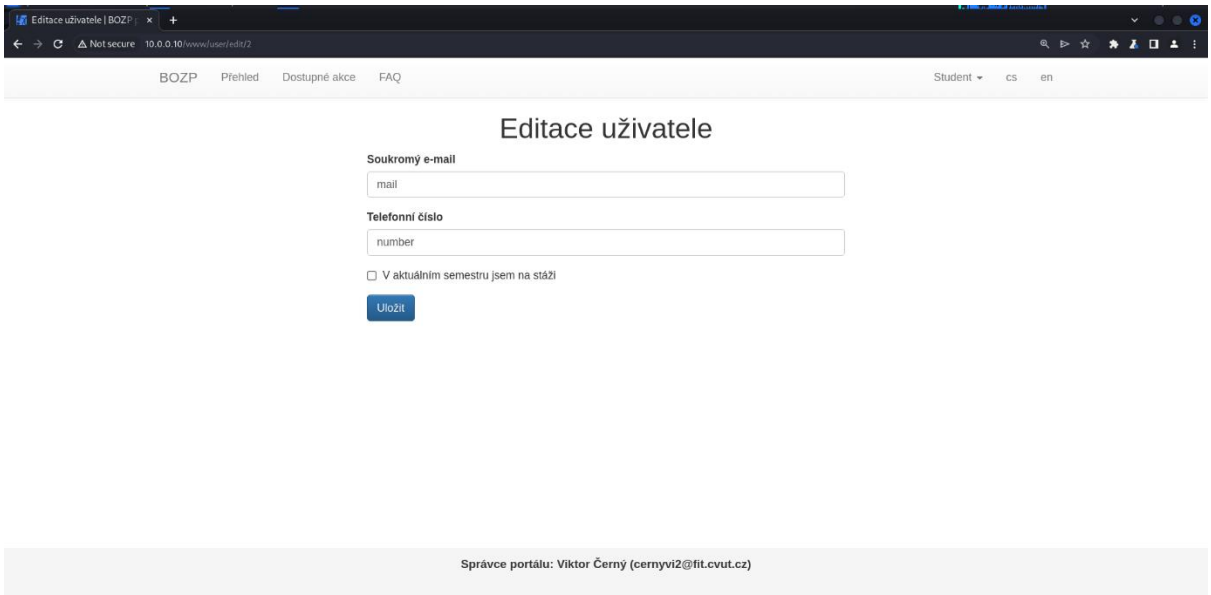
The only entry point for a guest is the login form (All links lead to an external site):





After logging in, the user can enter data into the event search filter or change their personal data such as personal email and phone number.





All input data is sent via a POST request with the following parameters:

```
POST /www/sign/in HTTP/1.1
Host: 10.0.0.10
Content-Length: 157
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
Origin: http://10.0.0.10
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/121.0.6167.85 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b
3;q=0.7
Referer: http://10.0.0.10/www/sign/in
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
Cookie: PHPSESSID=em77e7p7tvj7vs0ucnk5mpf0il
Connection: close

username=student&password=net123&send=P%C5%99ihl%C3%A1sit&_token_=7x7aktvsunnjhnbcvgTsFAX8sikxKy%2FzND95g%3D&remember=
Keep+me+signed+in&_do=signInForm-submit
```

login form

Request

Pretty Raw Hex

```
1  POST /www/event/available HTTP/1.1
2  Host: 10.0.0.10
3  Content-Length: 130
4  Accept: */*
5  X-Requested-With: XMLHttpRequest
6  User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/121.0.6167.85 Safari/537.36
7  Content-Type: application/x-www-form-urlencoded; charset=UTF-8
8  Origin: http://10.0.0.10
9  Referer: http://10.0.0.10/www/event/available
10 Accept-Encoding: gzip, deflate, br
11 Accept-Language: en-US,en;q=0.9
12 Cookie: PHPSESSID=2a85dnudd443jo3vm6jei8j67g
13 Connection: close
14
15 filter%5Bname%5D=all&filter%5Btraining_type_id%5D=&filter%5Btype%5D=&_do=availableEventGrid-form-submit&filter%5Bfilter%5D=
   Filter&
```

Search 0 highlights

Event log (4) All issues

filter form

Request

Pretty Raw Hex

```
1  POST /www/user/edit/2 HTTP/1.1
2  Host: 10.0.0.10
3  Content-Length: 133
4  Cache-Control: max-age=0
5  Upgrade-Insecure-Requests: 1
6  Origin: http://10.0.0.10
7  Content-Type: application/x-www-form-urlencoded
8  User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/121.0.6167.85 Safari/537.36
9  Accept:
   text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b
   3;q=0.7
.0 Referer: http://10.0.0.10/www/user/edit/2
.1 Accept-Encoding: gzip, deflate, br
.2 Accept-Language: en-US,en;q=0.9
.3 Cookie: PHPSESSID=lrd9s7ligdkrkj760tgnnnm39o
.4 Connection: close
.5
.6 personal_mail=mail&telephone=number&send=Ulo%C5%BEit&_token_=3sivtp2s5bZhNFJfN33L8GkeFaGULwmXeJcBw%3D&id=2&_do=
   profileEditForm-submit
```

Search 0 highlights

Event log (4) All issues

changing mail or phone form

Configuration and Deployment Management Testing

Test Network Infrastructure Configuration

During testing, it was revealed that the web application uses the Apache web server version 2.4.38. This version was released on 22.09.2019, and since then many updates have been released to fix various vulnerabilities and security improvements. Using outdated software puts your system at risk because known vulnerabilities can be used by attackers to launch attacks.

Test HTTP Methods

```
┌──(kali㉿kali)-[~]
└─$ nmap --script http-methods 10.0.0.10
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-06-11 09:18 EDT
Nmap scan report for 10.0.0.10
Host is up (0.0037s latency).
Not shown: 997 closed tcp ports (conn-refused)
PORT     STATE SERVICE
22/tcp   open  ssh
80/tcp   open  http
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
3306/tcp open  mysql

Nmap done: 1 IP address (1 host up) scanned in 0.94 seconds
```

nmap --script http-methods 10.0.0.10

The allowed methods are: GET, HEAD, POST, OPTIONS.

Test HTTP Strict Transport Security

```
┌──(kali㉿kali)-[~]
└─$ curl -s -D- 10.0.0.10
HTTP/1.1 302 Found
Date: Tue, 11 Jun 2024 13:33:15 GMT
Server: Apache/2.4.38 (Debian)
Location: http://10.0.0.10/www/
Content-Length: 0
Content-Type: text/html; charset=UTF-8

┌──(kali㉿kali)-[~]
└─$ curl -s -D- 10.0.0.10 | grep -i Strict-Transport-Security

┌──(kali㉿kali)-[~]
└─$
```

curl -s -D- 10.0.0.10 | grep -i Strict-Transport-Security

During testing, it was discovered that the website does not use the HSTS header. This means that connections can be made over HTTP, making them vulnerable to MitM attacks such as SSL stripping.

This is a derived vulnerability from an existing case

Identity Management Testing

Test Role Definitions

There are several roles:

- Guest – no permissions
- Student – viewing and registering for events
- Admin – full permissions

Test user registration process

Our version of the application does not allow registering new users.

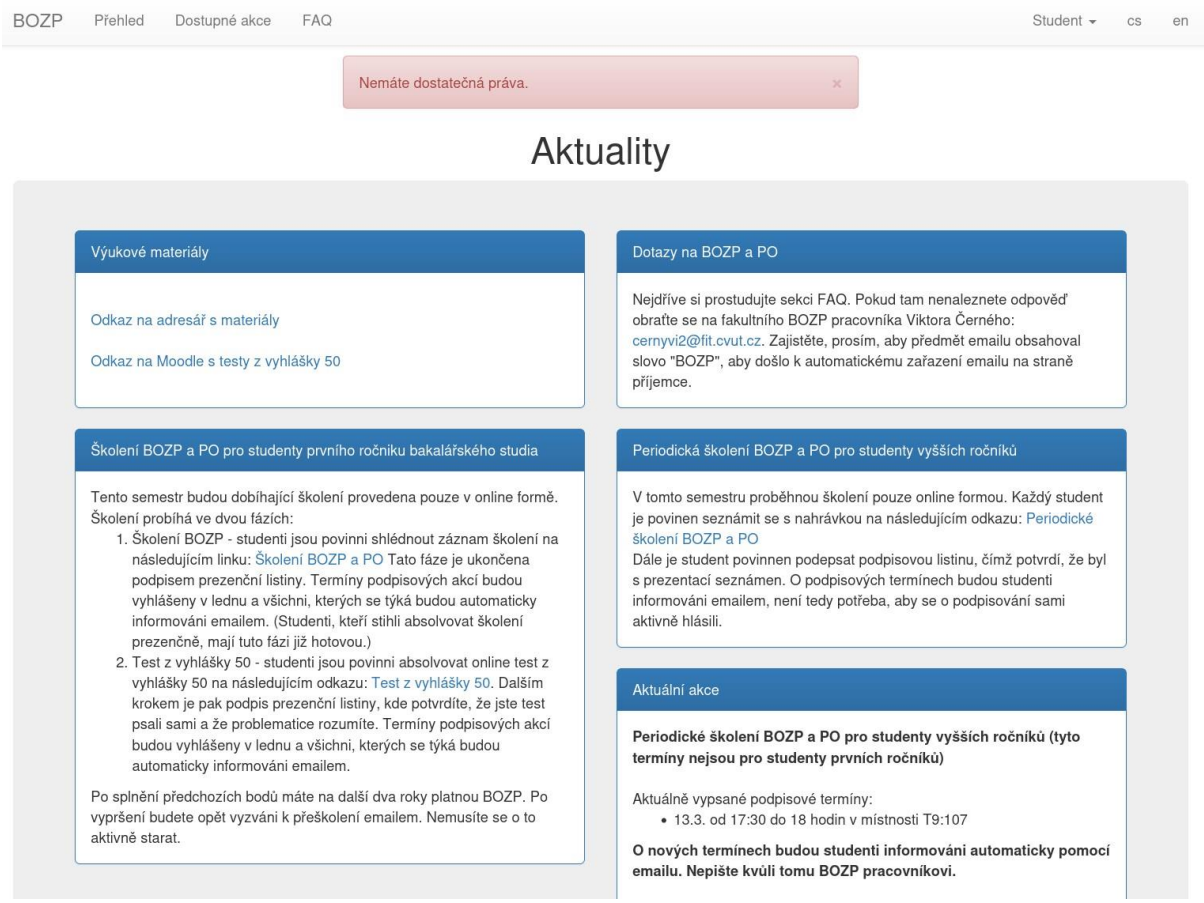
Authentication testing

Testing for Credentials Transported over an Encrypted Channel

Vulnerability found – analysis.

Testing for Bypassing Authentication Schema

No vulnerability was found, after trying to access pages for authenticated users as unauthenticated user or access only admin's pages as some user application returns error:



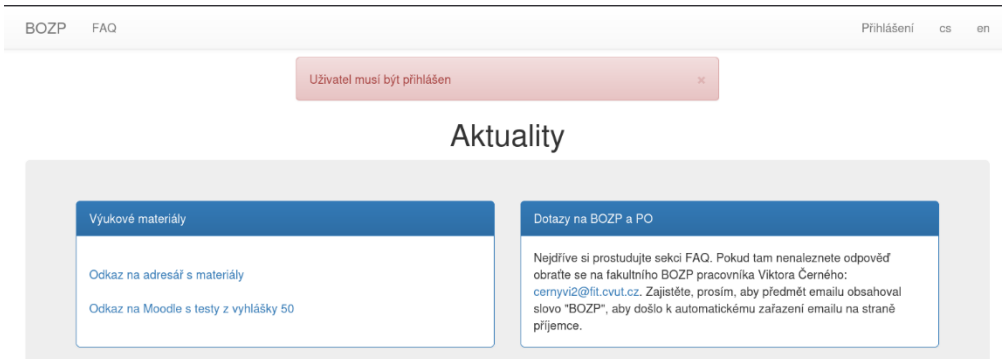
Web page for modifying profile looks like “http://10.0.0.10/www/user/profile/2”, where 2 is a user’s ID. Even if entering really large number (20+ digits) it returns the same error. Admin can access those pages directly.

There was no obvious way to predict session ID, every value seems to be randomly generated each time a new request is sent:

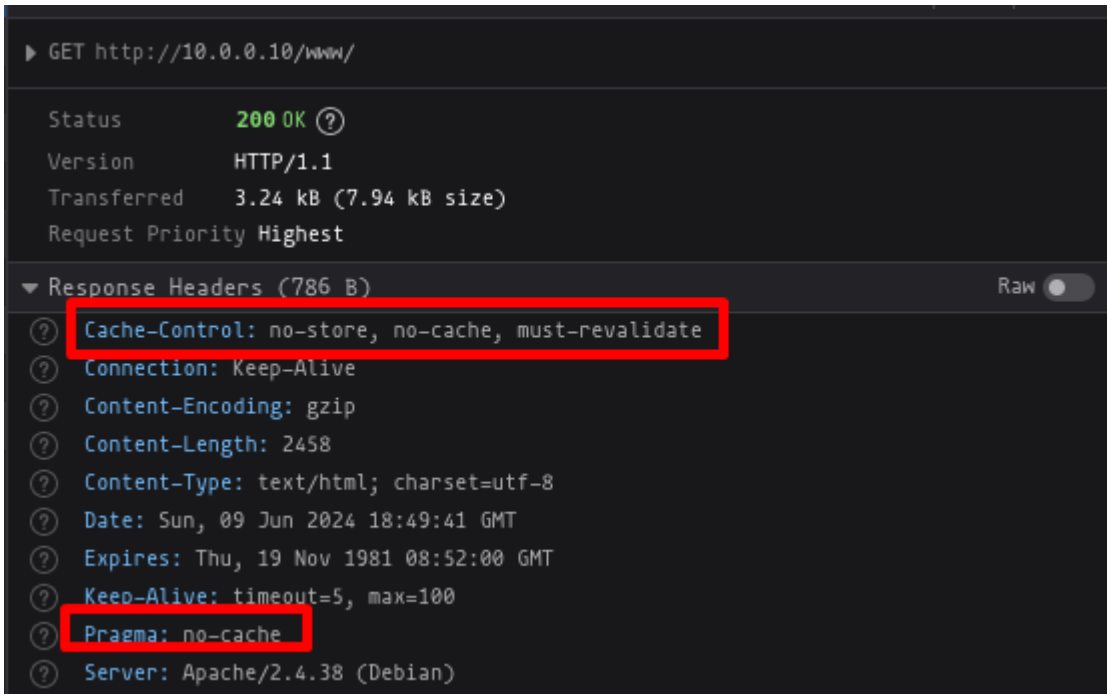
Value	Value
f9ru99q9nqcpbnkarnr9bsnlhv	brr6elvtvuc74c45oac4ng2ne0

Testing for Browser Cache Weaknesses

After logging out, no way to access previous accessible information was found, application returns error:



Also, there are a “Cache-Control: must-revalidate, no-store, no-cache” flag and “Pragma: no-cache” flags that secure browser’s cache and history. Although, additional flags such as “Cache-Control: max-age=0, s-maxage=0” and “Expires: 0” would be great for extra security.



Same approach tested on mobile client using Firefox Response Design Mode, all headers are the same:

```
Response Headers (786 B)                                   Raw
Cache-Control: no-store, no-cache, must-revalidate
Connection: Keep-Alive
Content-Encoding: gzip
Content-Length: 1271
Content-Type: text/html; charset=utf-8
Date: Sun, 02 Jun 2024 17:01:55 GMT
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Keep-Alive: timeout=5, max=100
Pragma: no-cache
Server: Apache/2.4.38 (Debian)
Set-Cookie: PHPSESSID=r1qqalvprvgigc1pjn79udl24m; expires=Sun, 16-Jun-2024 17:01:55 GMT; Ma
x-Age=1209600; path=/; HttpOnly
Set-Cookie: PHPSESSID=eqnls71g91eoaj2u301t8rfieg; expires=Sun, 16-Jun-2024 17:01:55 GMT; Max
-Age=1209600; path=/; HttpOnly
Set-Cookie: PHPSESSID=eqnls71g91eoaj2u301t8rfieg; expires=Sun, 16-Jun-2024 17:01:55 GMT; Max
-Age=1209600; path=/; HttpOnly
Vary: X-Requested-With,Accept-Encoding
X-Frame-Options: SAMEORIGIN
X-Powered-By: Nette Framework
Request Headers (489 B)                                    Raw
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.5
Connection: keep-alive
Cookie: PHPSESSID=r1qqalvprvgigc1pjn79udl24m
Host: 10.0.0.10
Referer: http://10.0.0.10/www/
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (iPhone; CPU iPhone OS 14_6 like Mac OS X) AppleWebKit/605.1.15 (KHTML,
like Gecko) Version/14.0.3 Mobile/15E148 Safari/604.1
```

Testing for Weak Password

Application has 2 users – student and admin, both with the same password, which is highly vulnerable, because user can gain admin's access using its password.

Moreover, password is "net123", which is (as per Kali Linux 2024.2) line 122704 in **rockyou.txt**, so it can be easily compromised using dictionary attacks.

```
-> cat rockyou.txt|
122704  net123
```

Other issues

Application doesn't provide features like lock-out mechanism, password reset, "Forgot my password" or security question so those were not tested.

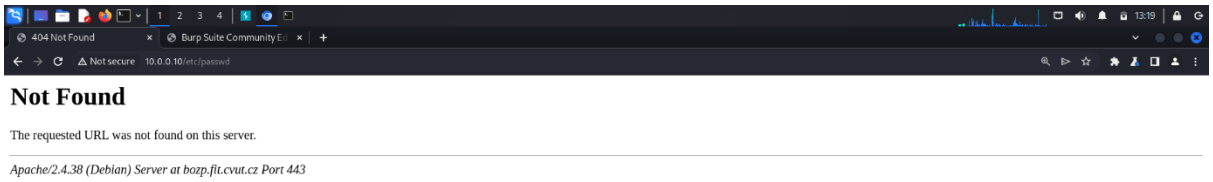
Authorization Testing

Testing Directory Traversal File Include

I tried to access the "/etc/ passwd" via URL

https://10.0.0.10/..%2F..%2F..%2F..%2F..%2F..%2Fetc%2Fpasswd.

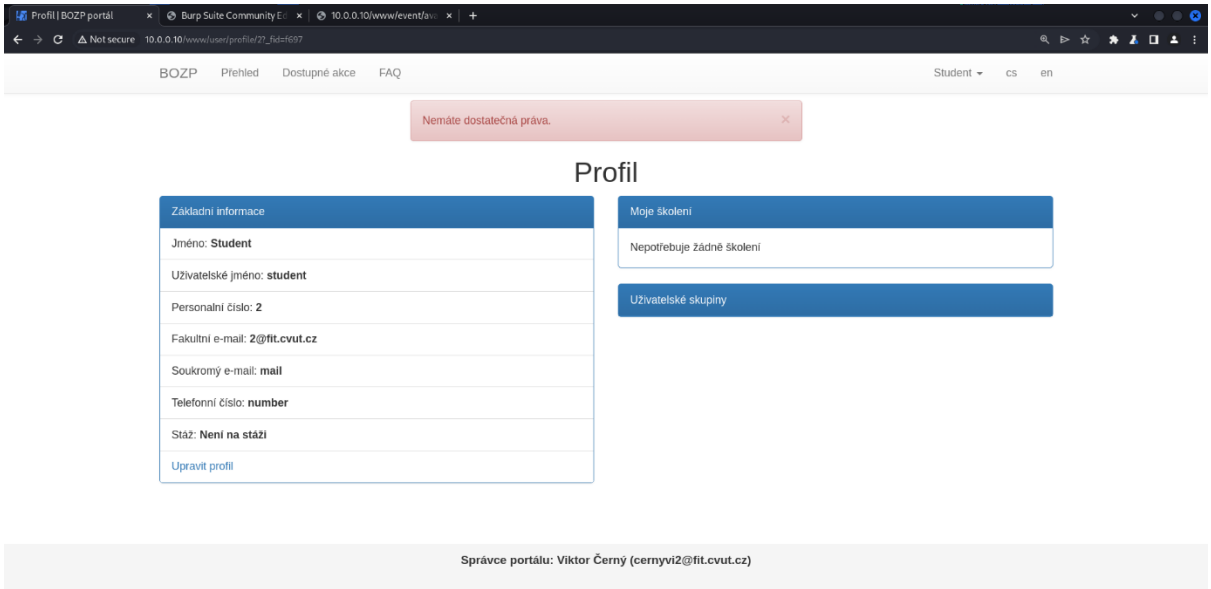
However, it wasn't successful:



Other possible vulnerabilities on the part of the student or guest were not found.

Testing for Bypassing Authorization Schema

I was unable to access any resources beyond my designated role permissions. I also attempted to escalate my privileges to root, but this required higher-level permissions that I did not possess.



http://10.0.0.10/www/user/profile/2 - changing 2 to another value is not possible.

However, a vulnerability was discovered due to which a student can change the personal email and phone number of another user (even an admin) - Analysis.

Session Management Testing

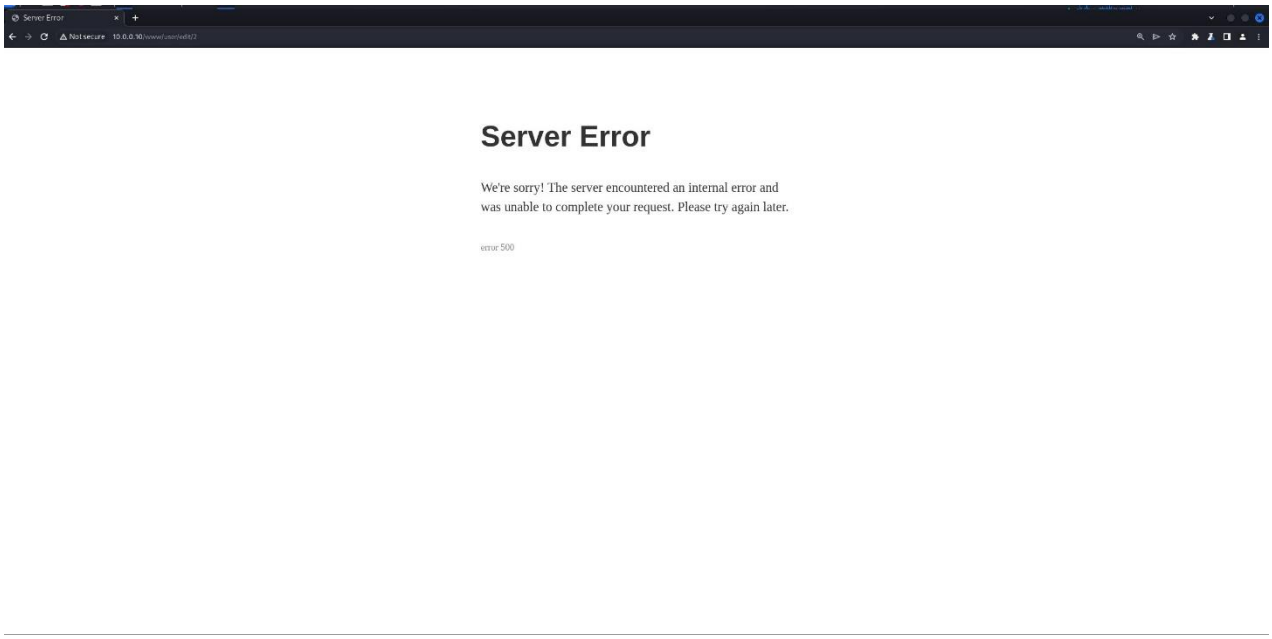
Cookie collection

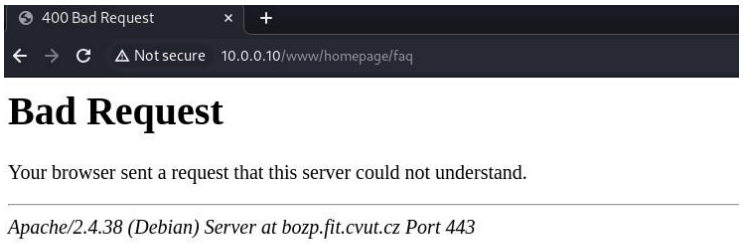
There is one cookies PHPSESSID, which is an autogenerated cookies used by the server to manage sessions.

Testing for Error Handling

Testing for Improper Error Handling

I've tried multiple ways to trigger an error such as changing HTTP header or a very long input but none of them gave me useful information except webserver's name:





Input Validation Testing

XSS: Reflected and Stored

There is no input allowed for an unauthenticated user and only 2 places for input for regular user: email and phone number.

I've tried multiple payloads from [XSS Filter Evasion Cheatsheet]() but none of them worked, both on Chrome and Firefox.

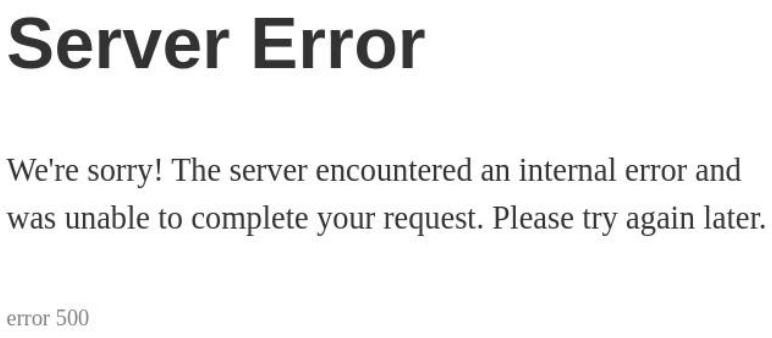
But the issue remains the same - absolutely no sanitation of input whatsoever: every special character is allowed, absence of email's check for "x@x.x" format, same goes for phone number.

CSRF

CSRF vulnerability found – analysis.

SQL injection

No SQL injection was found, whether using admin or regular user inputs, URL or regular forms. The result was either nothing or error 500, probably because server received some special character in string or integer variable.



Other issues

Other issues related to Input Validation such as IMAP SMTP Injection were either not found or application doesn't provide functionality so that those issues could be exploited.

Vulnerability analysis

Unencrypted transfer of credentials

Description

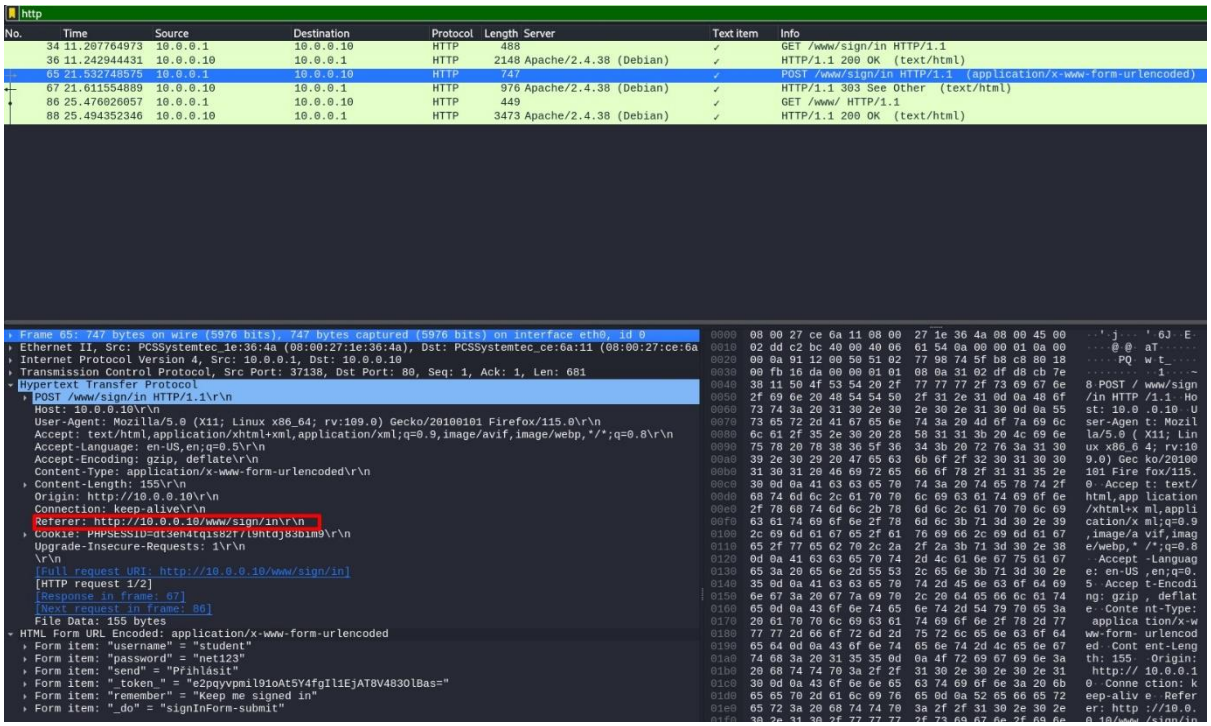
The only place where credential can be entered is login page.

Credentials seem to be passed by HTTP protocol, which is crucial security flaw because user's credentials can be seen clearly.

```
POST http://10.0.0.10/www/sign/in

Status          303 See Other
Version         HTTP/1.1
Transferred     1.47 kB (0 B size)
Referrer Policy strict-origin-when-cross-origin
Request Priority Highest

Response Headers (788 B)                                   Raw
Cache-Control: no-store, no-cache, must-revalidate
Connection: Keep-Alive
Content-Length: 96
Content-Type: text/html; charset=utf-8
Date: Sun, 09 Jun 2024 18:17:53 GMT
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Keep-Alive: timeout=5, max=100
Location: http://10.0.0.10:443/www/
Pragma: no-cache
Server: Apache/2.4.38 (Debian)
Set-Cookie: PHPSESSID=d4u90bujmfvsp4jhp9n7isvb38; expires=Sun, 23-Jun-2024 18:1
7:53 GMT; Max-Age=1209600; path=/; HttpOnly
Set-Cookie: PHPSESSID=q0nno9msp0naj2o6lnptbv6dvq; expires=Sun, 23-Jun-2024 18:1
7:53 GMT; Max-Age=1209600; path=/; HttpOnly
Set-Cookie: PHPSESSID=q0nno9msp0naj2o6lnptbv6dvq; expires=Sun, 23-Jun-2024 18:1
7:53 GMT; Max-Age=1209600; path=/; HttpOnly
Vary: X-Requested-With
X-Frame-Options: SAMEORIGIN
X-Powered-By: Nette Framework

Request Headers (526 B)                                    Raw
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/
webp,*/*;q=0.8
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.5
Connection: keep-alive
Content-Length: 157
Content-Type: application/x-www-form-urlencoded
Cookie: PHPSESSID=d4u90bujmfvsp4jhp9n7isvb38
Host: 10.0.0.10
Origin: http://10.0.0.10
Referer: http://10.0.0.10/www/sign/in
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/11
5.0
```

```
Headers   Cookies   Request   Response   Timings
Filter Request Parameters
Form data
  username: "student"
  password: "net123"
  send: "Přihlásit"
  _token_: "e0l7afs5vrNYrPi8w3fINupknyd/wnGHcu9LU="
  remember: "Keep+me+signed+in"
  _do: "signInForm-submit"
```

As an addition, server return information about session cookie through "Set-Cookie" that has no "Secure" attribute (avoids exposing cookie over unencrypted channels).



Remediation

Use HTTPS everywhere you can. If it is a problem, start with the most sensitive operations and convert the application to HTTPS step-by-step. Implement HSTS and redirect any HTTP request to HTTPS. Set “Set-Cookie: Secure” flag.

Using an outdated server

Description

Using an outdated version of Apache/2.4.38 poses a significant security risk to your web application. Attackers frequently target outdated software versions because vulnerabilities associated with them are documented in public databases. Exploiting these known vulnerabilities can allow attackers to compromise the server.

Proof Of Concept (POC)

1. I used the nikto tool to analyse the application.

```
┌──(kali㉿kali)-[~]
└─$ nikto -h 10.0.0.10
- Nikto v2.5.0
---------------------------------------------------------------------------
+ Target IP:          10.0.0.10
+ Target Hostname:    10.0.0.10
+ Target Port:        80
+ Start Time:         2024-04-27 07:48:40 (GMT-4)
---------------------------------------------------------------------------
+ Server: Apache/2.4.38 (Debian)
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Fram
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fas
anner/vulnerabilities/missing-content-type-header/
+ Root page / redirects to: http://10.0.0.10/www/
+ Apache/2.4.38 appears to be outdated (current is at least Apache/2.4.54). Apache 2.2.34 is the EOL for the 2.x branch.
+ /www/: Retrieved x-powered-by header: Nette Framework.
+ /www/: This might be interesting.
+ /icons/README: Apache default file found. See: https://www.vntweb.co.uk/apache-restricting-access-to-iconsreadme/
+ /composer.json: PHP Composer configuration file reveals configuration information. See: https://getcomposer.org/
+ /composer.lock: PHP Composer configuration file reveals configuration information. See: https://getcomposer.org/
+ /.gitignore: .gitignore file found. It is possible to grasp the directory structure.
+ 9661 requests: 0 error(s) and 9 item(s) reported on remote host
+ End Time:           2024-04-27 07:49:05 (GMT-4) (25 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
```

Remediation

Update Apache server to the latest stable version. At the time of writing, the latest version is Apache 2.4.54 (or another current version, if it came out later).

CSRF – Homepage

Description

CSRF - is an attack that forces an end user to execute unintended actions on a web application in which they are currently authenticated

The only thing a regular user can change is its email and phone number, which is guarded by Nette's CSRF token.

However, administrator can change homepage of a website and specifically delete some articles. This action is performed by sending GET request. So, running this HTML file in admin's session results in deleting homepage's articles.

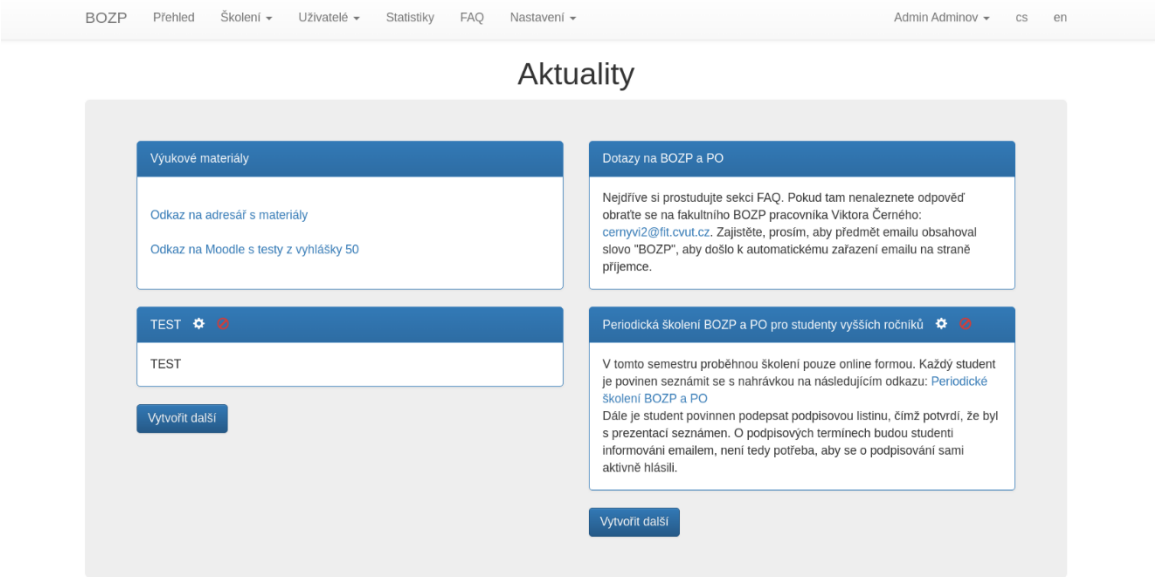
Proof Of Concept (POC)

1. Create a HTML file with this content (33 is an ID for an article, it may differ, number is low so it can be bruteforced):

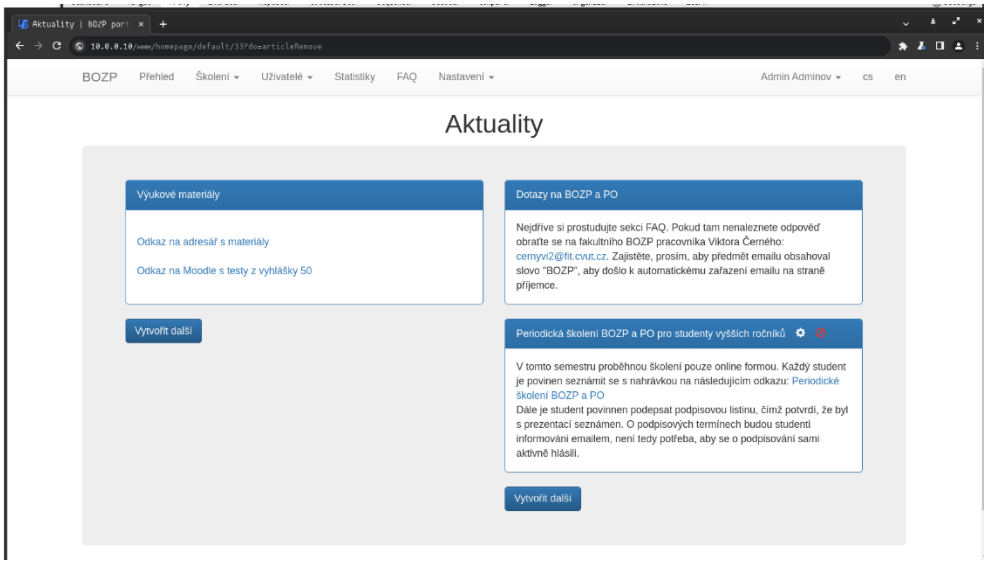
```html
<!DOCTYPE html>
<html>
<body onload="document.CSRF.submit()">
    <form action="http://10.0.0.10/www/homepage/default/33"  method="GET" name="CSRF">
        <input type="hidden" name="do" value="articleRemove">
    </form>
</body>
</html>
```

2. Log in into admin's account

3. Visit Homepage(for clear result)



4. Run premade HTML file with the same browser you have your active session on (there is a great chance that admin is using his default browser for visiting this page, so our HTML file will also be opened by default browser)



As a result, article with ID 33 will be removed from the homepage with no back-up. This way you can make a lot of these requests to remove every article that can be removed.

Remediation

Do not use GET requests to manipulate with sensitive or important data, use Nette framework for CSRF token.

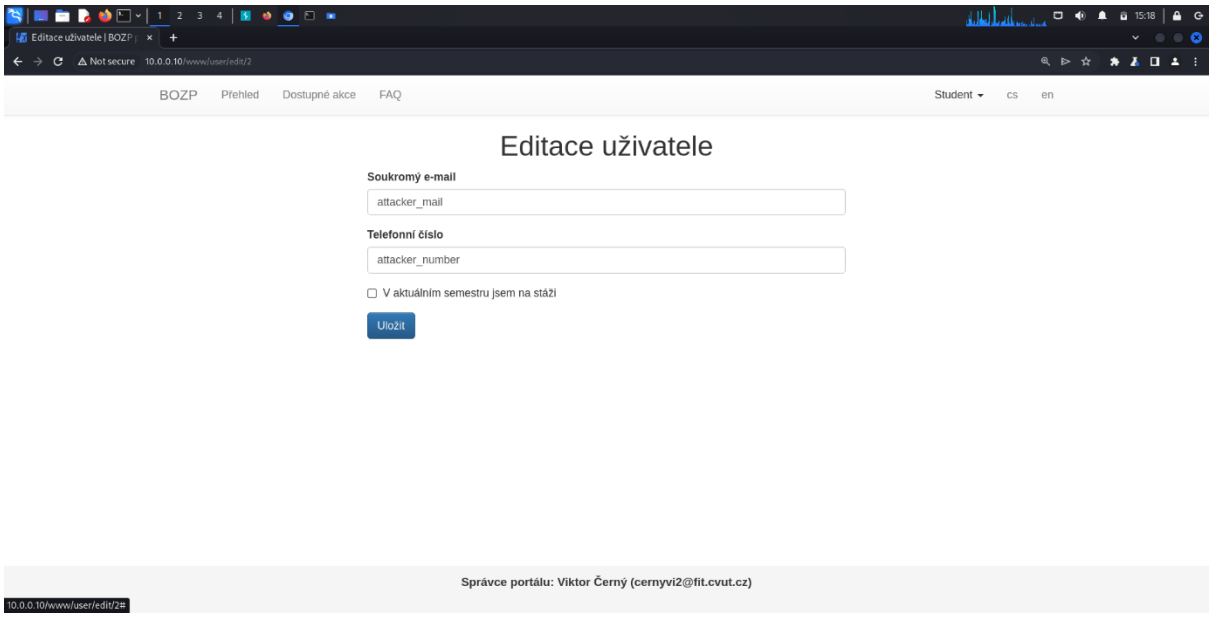
Improper Access Control

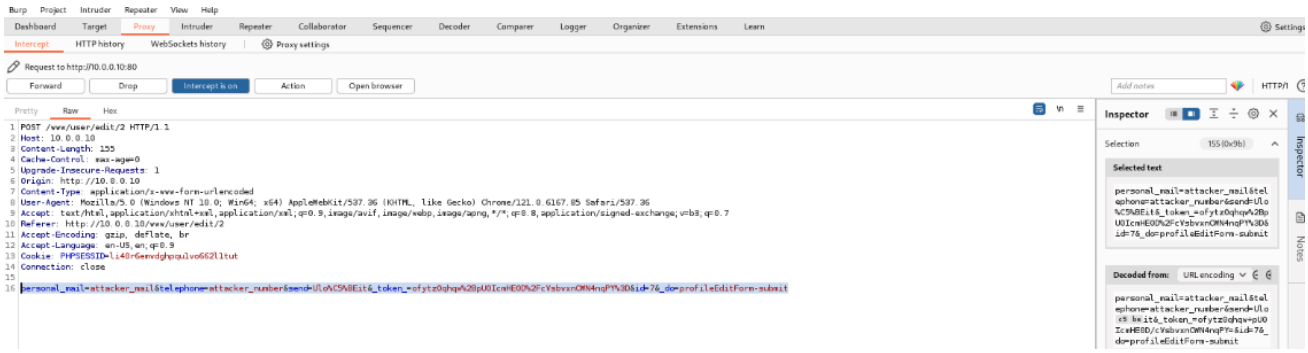
Description

The vulnerability occurs when the system does not properly control or restrict access to resources based on user rights. As a result, users may access data or perform actions that they do not have permission to do. In our case, a User without administrative rights can change or delete another user's data.

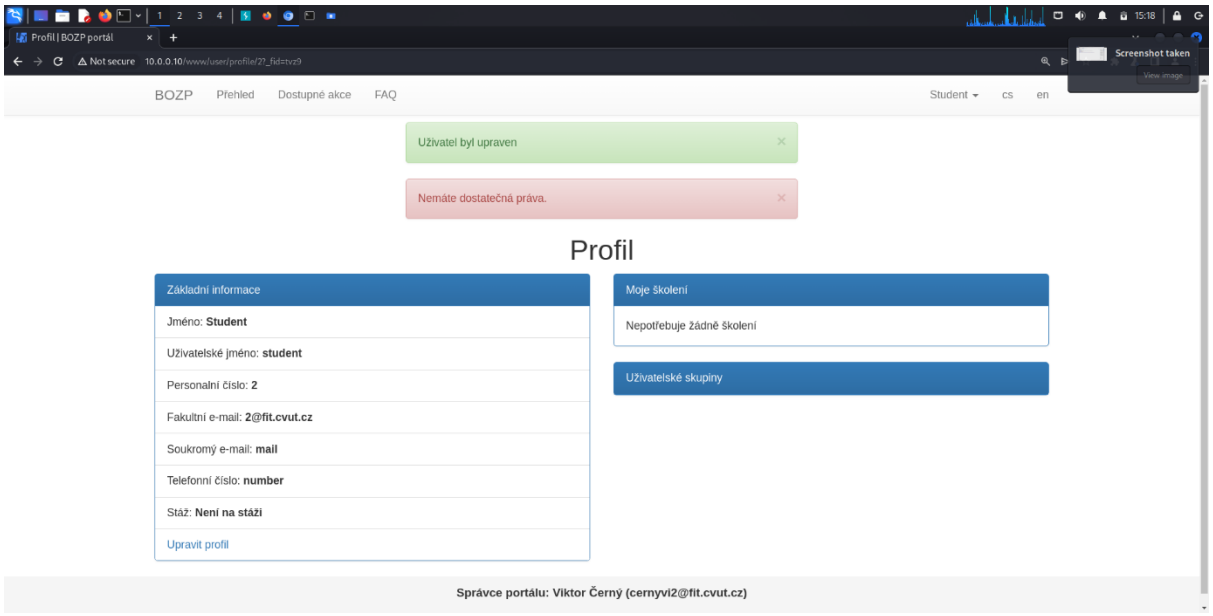
Proof Of Concept (POC)

1. Here in the POST request, I changed the ID to another user (my ID was 2).

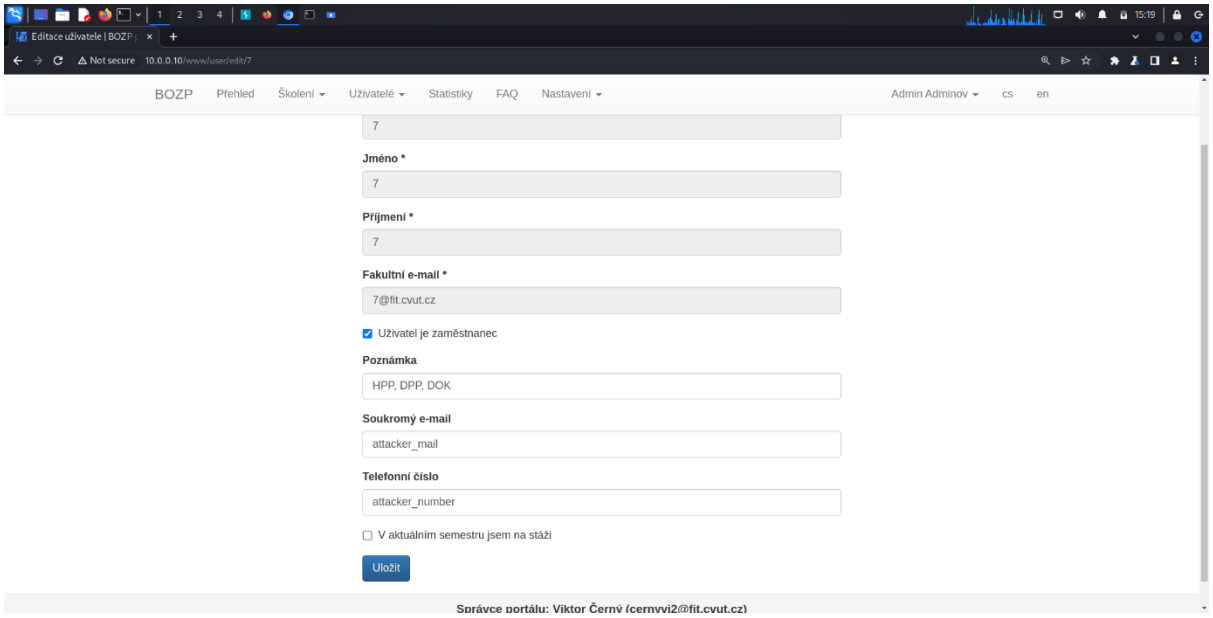




2. Then I get this result:



3. I check the changes under the admin account:



Remediation

- Implement server-side authorization checks for all actions that may change data or configuration.
- Ensure that an authorization check occurs before each critical action.
- Keep logs of all actions related to data changes or privilege escalations, and regularly analyse them for suspicious activity.

Presence of default files

Description

During testing, default files were discovered, such as /icons/README, /.gitignore, /composer.json and /composer.lock. These files are often left behind after installing a web server or web application and may contain information about the configuration or structure of the system.

Proof Of Concept (POC)

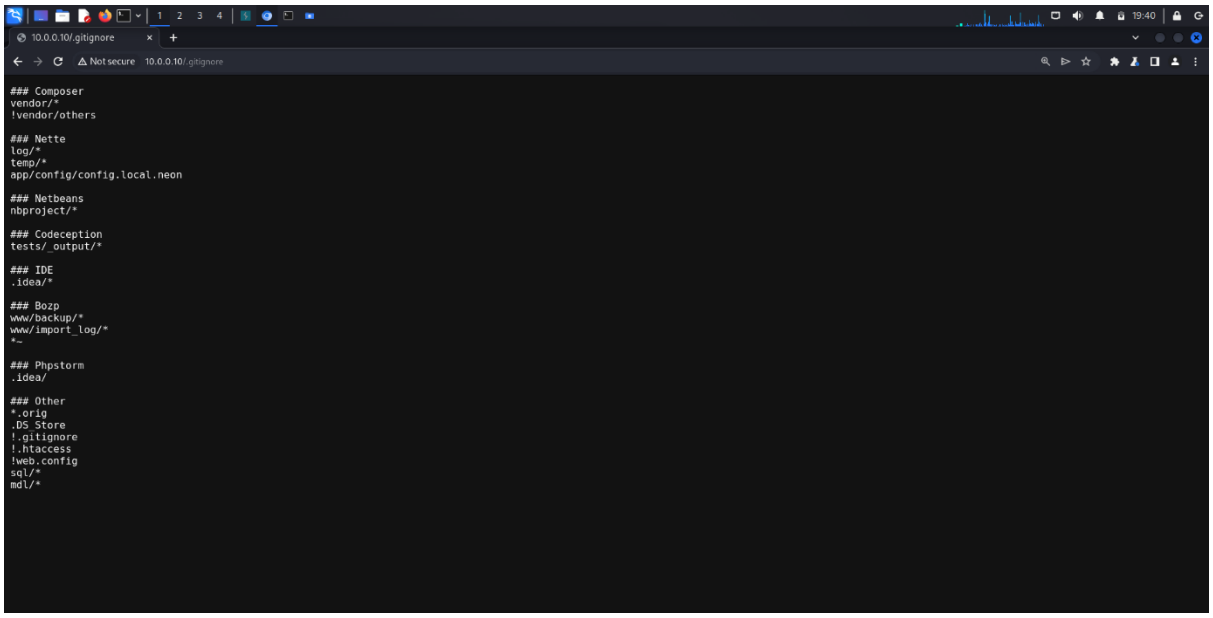
1. Nikto output:

```
┌──(kali㉿kali)-[~]
└─$ nikto -h 10.0.0.10
- Nikto v2.5.0
---------------------------------------------------------------------------
+ Target IP:          10.0.0.10
+ Target Hostname:    10.0.0.10
+ Target Port:        80
+ Start Time:         2024-04-27 07:48:40 (GMT-4)
---------------------------------------------------------------------------
+ Server: Apache/2.4.38 (Debian)
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Fram
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fas
anner/vulnerabilities/missing-content-type-header/
+ Root page / redirects to: http://10.0.0.10/www/
+ Apache/2.4.38 appears to be outdated (current is at least Apache/2.4.54). Apache 2.2.34 is the EOL for the 2.x branch.
+ /www/: Retrieved x-powered-by header: Nette Framework.
+ /www/: This might be interesting.
+ /icons/README: Apache default file found. See: https://www.vntweb.co.uk/apache-restricting-access-to-iconsreadme/
+ /composer.json: PHP Composer configuration file reveals configuration information. See: https://getcomposer.org/
+ /composer.lock: PHP Composer configuration file reveals configuration information. See: https://getcomposer.org/
+ /.gitignore: .gitignore file found. It is possible to grasp the directory structure.
+ 9681 requests: 0 error(s) and 9 item(s) reported on remote host
+ End Time:           2024-04-27 07:49:05 (GMT-4) (25 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
```

/icons/README:

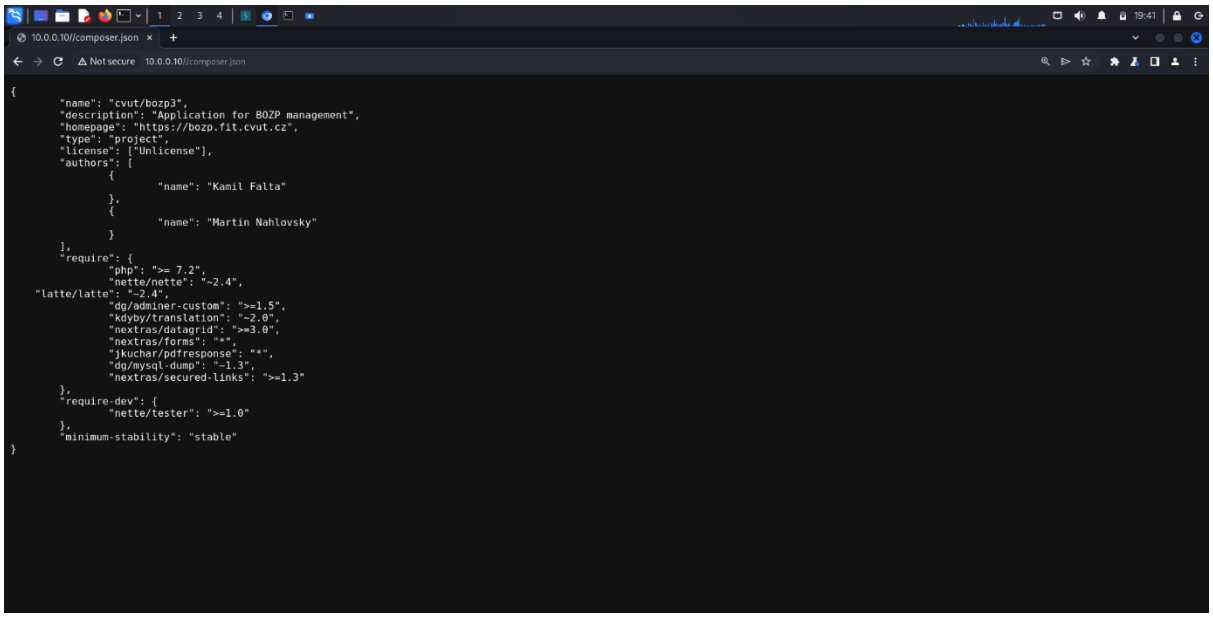
```
Public Domain Icons

     These icons were originally made for Mosaic for X and have been
     included in the NCSA httpd and Apache server distributions in the
     past. They are in the public domain and may be freely included in any
     application. The originals were done by Kevin Hughes (kevinh@kevcom.com).
     Andy Polyakov tuned the icon colors and added a few new images.

     If you'd like to contribute additions to this set, contact the httpd
     documentation project <http://httpd.apache.org/docs-project/>.

     Almost all of these icons are 20x22 pixels in size.  There are
     alternative icons in the "small" directory that are 16x16 in size,
     provided by Mike Brown (mike@hyperreal.org).

Suggested Uses

The following are a few suggestions, to serve as a starting point for ideas.
Please feel free to tweak and rename the icons as you like.

     a.gif
          This might be used to represent PostScript or text layout
          languages.

     alert.black.gif, alert.red.gif
          These can be used to highlight any important items, such as a
          README file in a directory.

     back.gif, forward.gif
          These can be used as links to go to previous and next areas.

     ball.gray.gif, ball.red.gif
          These might be used as bullets.

     binary.gif
          This can be used to represent binary files.

     binhex.gif
          This can represent BinHex-encoded data.

     blank.gif
          This can be used as a placeholder or a spacing element.

     bomb.gif
```

.gitignore:



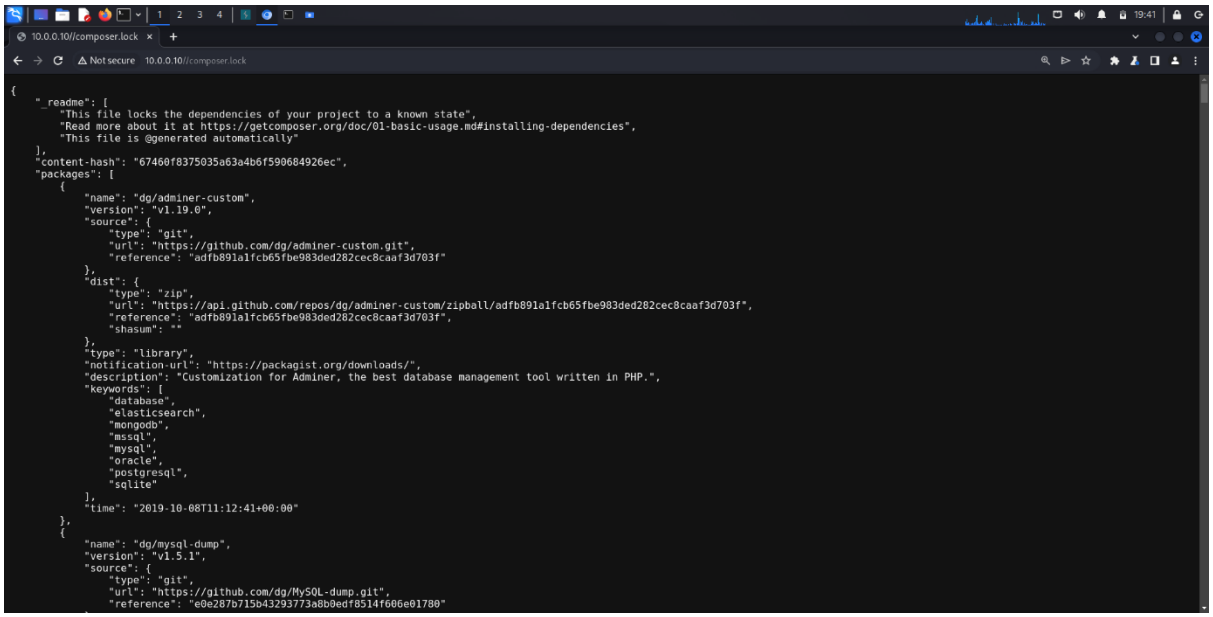
```
### Composer
vendor/*
!vendor/others

### Nette
log/*
temp/*
app/config/config.local.neon

### Netbeans
nbproject/*

### Codeception
tests/_output/*

### IDE
.idea/*

### Bozp
www/backup/*
www/import_log/*
*~

### Phpstorm
.idea/

### Other
*.orig
.DS_Store
!.gitignore
!.htaccess
!web.config
sql/*
mdl/*
```

/composer.json:




```
{
        "name": "cvut/bozp3",
        "description": "Application for BOZP management",
        "homepage": "https://bozp.fit.cvut.cz",
        "type": "project",
        "license": ["Unlicense"],
        "authors": [
                {
                        "name": "Kamil Falta"
                },
                {
                        "name": "Martin Nahlovsky"
                }
        ],
        "require": {
                "php": ">= 7.2",
                "nette/nette": "~2.4",
    "latte/latte": "~2.4",
                "dg/adminer-custom": ">=1.5",
                "kdyby/translation": "~2.0",
                "nextras/datagrid": ">=3.0",
                "nextras/forms": "*",
                "jkuchar/pdfresponse": "*",
                "dg/mysql-dump": "~1.3",
                "nextras/secured-links": ">=1.3"
        },
        "require-dev": {
                "nette/tester": ">=1.0"
        },
        "minimum-stability": "stable"
}
```

/composer.lock:




```
{
    "_readme": [
        "This file locks the dependencies of your project to a known state",
        "Read more about it at https://getcomposer.org/doc/01-basic-usage.md#installing-dependencies",
        "This file is @generated automatically"
    ],
    "content-hash": "67460f8375035a63a4b6f590684926ec",
    "packages": [
        {
            "name": "dg/adminer-custom",
            "version": "v1.19.0",
            "source": {
                "type": "git",
                "url": "https://github.com/dg/adminer-custom.git",
                "reference": "adfb891a1fcb65fbe983ded282cec8caaf3d703f"
            },
            "dist": {
                "type": "zip",
                "url": "https://api.github.com/repos/dg/adminer-custom/zipball/adfb891a1fcb65fbe983ded282cec8caaf3d703f",
                "reference": "adfb891a1fcb65fbe983ded282cec8caaf3d703f",
                "shasum": ""
            },
            "type": "library",
            "notification-url": "https://packagist.org/downloads/",
            "description": "Customization for Adminer, the best database management tool written in PHP.",
            "keywords": [
                "database",
                "elasticsearch",
                "mongodb",
                "mssql",
                "mysql",
                "oracle",
                "postgresql",
                "sqlite"
            ],
            "time": "2019-10-08T11:12:41+00:00"
        },
        {
            "name": "dg/mysql-dump",
            "version": "v1.5.1",
            "source": {
                "type": "git",
                "url": "https://github.com/dg/MySQL-dump.git",
                "reference": "e0e287b715b43293773a8b0edf8514f606e01780"
```


Remediation

- Audit all files and directories on the web server and remove default and unnecessary files that are not used in the application.
- If deleting files is not possible, restrict access to them using web server settings (for example, using .htaccess in Apache or configuration rules in Nginx).

Server shut down

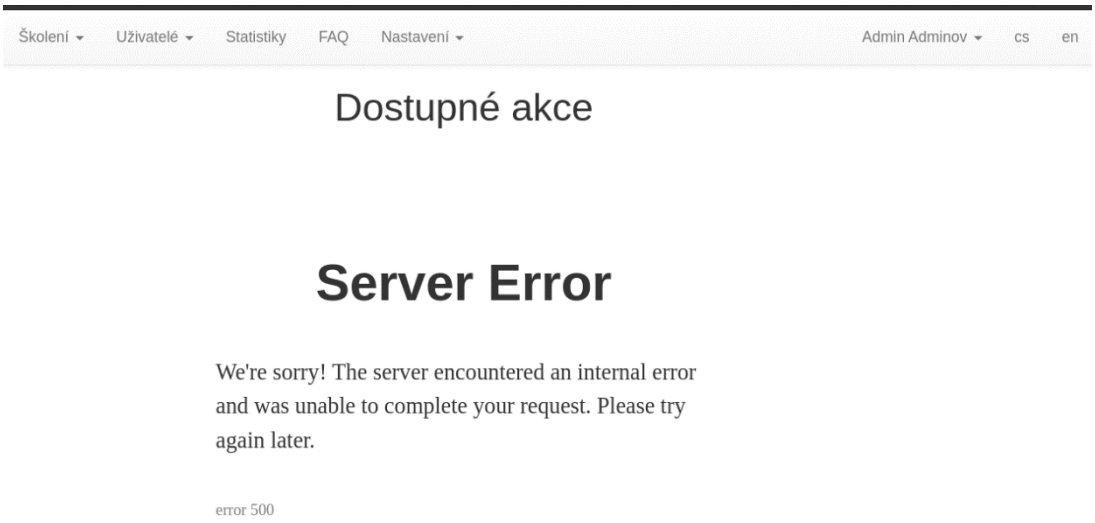
Description

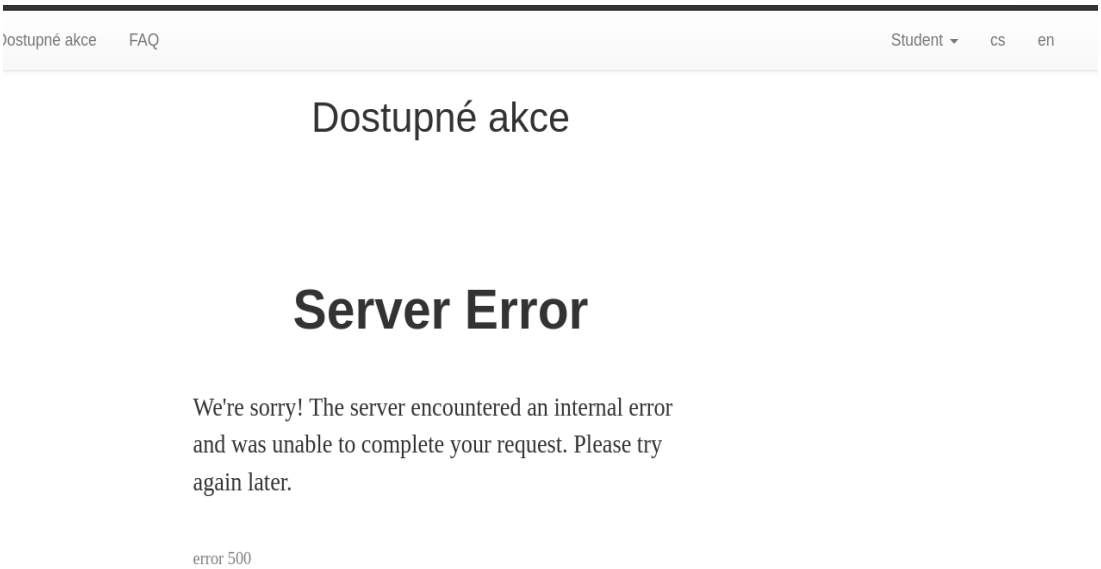
You can access a lot of “filter” forms across the website, and I've found a way that an admin’s “filter” form can crash the server so that students cannot view “Available events”

Proof Of Concept (POC)

1. Visit Manage events (“Správa akce”) as admin.
2. Type “\’ OR 1=1 --” and press “Filter”.

3. Try to visit “Available events” either as a user or an admin





Although it lasts only a minute or so, it still can be inconvenient for some users.

Remediation

Sanitize input so that special characters would behave just like normal ones (without special meaning) or use whitelists and allow only those characters while creating some entities as “Event”.