NETHEMBA. S R.O., GRÖSSLINGOVÁ 2503/62
811 09 BRATISLAVA

Comprehensive Penetration Test of “Unimus” web application

(Summary)

Customer: NetCore j.s.a.
Author: Nethemba
Consultant: Roman Fulop
Creation date: 2021-12-27
Last update: 2022-04-26

Table of Contents

1 Management Summary

From the 2nd until 21st of March 2022 we have performed a comprehensive penetration test of the Unimus web application according to the RFP of NetCore j.s.a.

The objective of this review was full coverage of the OWASP version 4.2 web application testing guide.

During the review we have identified one critical, two medium-risk and one low-risk issue affecting the application:

- Multiple stored XSS/HTML issues allow remote JavaScript code execution under attacked account — Critical
- Session fixation allows theft of session cookie in special cases — Medium
- Response time based account enumeration allows to find valid application login names — Medium
- No account lock out policy allows password guessing attacks — Low

Due to the fact a critical vulnerability has been found, we **urge to fix the issue for safe production operation** of the application.

We always recommend close inspection of issues listed in this document and consult our suggestions about their possible remediation.

Due to the dynamic character of security state of web applications, desktop and mobile applications, or infrastructure devices, when new advances in IT security field emerge almost daily and bring new, more advanced vulnerabilities and new attack vectors, we recommend to schedule regular penetration testing and security auditing of all mission-critical business applications and corresponding infrastructure units.

For all critical web applications we also recommend to perform a comprehensive source code review and analysis of critical components, libraries or interfaces and review underlying operating systems security in a form of a local OS audit.

2 Scope, Methodology, Tools, Risk Levels, Limiting Factors

2.1 Scope of the Review

The subject of the test was "**Unimus**" web application available at the following base URL:

- https://10.31.3.79:8085/

We have been provided with following accounts for the penetration test.

Login	Role
Unimus	Administrator

The application was accessed privately over provided VPN connection.

No information about the development platform, tools, libraries or frameworks used for the web application operation has been provided in advance.

The following areas were explicitly excluded from the scope of this penetration test on customer's request:

- SSL/TLS configuration
- Missing HTTP response headers

2.2 Methodology and Tools

The comprehensive web application penetration test strictly follows the OWASP version 4.2 web application testing guide that focuses primarily on OWASP Top Ten:

- A01:2021-Broken Access Control
- A02:2021-Cryptographic Failures
- A03:2021-Injection
- A04:2021-Insecure Design
- A05:2021-Security Misconfiguration
- A06:2021-Vulnerable and Outdated Components
- A07:2021-Identification and Authentication Failures
- A08:2021-Software and Data Integrity Failures
- A09:2021-Security Logging and Monitoring Failures
- A10:2021-Server-Side Request Forgery

Although we have used many automated commercial (e.g. BurpSuite Pro, Nessus) and open-source tools (see https://owasp.org/www-community/Vulnerability_Scanning_Tools), many parts of the applications required manual testing and inspection.

The majority of our testing involves manual inspection that follows current best-practices in web application security.

2.3 Risk Levels

To describe a severity of revealed vulnerabilities we define the following risk levels:

- Critical – revealed vulnerabilities can be immediately exploited to take over the application or its users. (Critical vulnerabilities are SQL injection, persistent XSS, serious vulnerabilities in business logic, buffer overflows, privilege attacks, possibility of DOS attacks ...)
- High – revealed vulnerabilities can be critical in combination with other vulnerabilities (e. g. reflected/DOM-based XSS, Cross Site Request Forgery (CSRF), session fixation attacks ...)
- Medium – certain conditions have to be fulfilled to exploit these vulnerabilities or potential damage is limited (e. g. weak SSL ciphers, absence of cookie "Secure" or "HttpOnly" flag, ...)
- Low – minor security issues (e. g. enabled TRACE/TRACK HTTP methods)

2.4 Limiting Factors

No limiting factors were observed during testing.