VERCOM

Certyfikat z testów bezpieczeństwa opartych na metodyce OWASP TOP10 i OWASP ASVS

PRZEDMIOT PRAC
Podsumowanie testów bezpieczeństwa aplikacji webowej *.serwersms.pl

DATA WYKONANIA TESTU
21.03.2023 – Audyt wewnętrzny (Skan aplikacji webowej *.serwersms.pl)
26.04.2023 – 04.05.2023 – Audyt wewnętrzny (Aplikacja Webowa + HTTPS API)
04.05.2023 – 11.05.2023 – Audyt zewnętrzny (Aplikacja Webowa)
05.05.2023 – 12.05.2023 – Audyt zewnętrzny (HTTPS API v2)
28.05.2023 – Audyt wewnętrzny (Skan CMS – serwersms.pl)
05.02.2025 - Audyt zewnętrzny (Test Army)
24.11.2025 - 10.12.2025 - Audyt wewnętrzny (Aplikacja Webowa)

DATA PODSUMOWANIA
17.12.2025

AUDYTORZY
Pentester: Gilewicz Michał (Vercom)
Firma zewnętrzna: Securitum Audyty Sp. z o.o. Sp. k., TestArmy Group S.A.

Podsumowanie wykonanych prac

Niniejszy raport jest podsumowaniem testów bezpieczeństwa prowadzonego przez Michała Gilewicz jak również firmę zewnętrzną: Securitum z siedzibą w Krakowie oraz TestArmy z siedzibą w Wrocławiu. Przedmiotem testów były aplikacje webowe z domeny *.serwersms.pl m.in. serwersms.pl, panel.serwersms.pl, api.serwersms.pl, api2.serwersms.pl.

W trakcie audytu szczególny nacisk położono na podatności mające lub mogące mieć negatywny wpływ na poufność, integralność oraz dostępność przetwarzanych danych.

Testy bezpieczeństwa przeprowadzono zgodnie z powszechnie przyjętymi metodykami testowania aplikacji webowymi, takimi jak: OWASP TOP10 czy OWASP ASVS które obejmowały m.in. testy mające na celu znalezienie błędów związanych z możliwością wstrzyknięcia złośliwego kodu (XSS, SQLi, SSTI itp.), błędów związanych z uwierzytelnianiem czy związanych z niepoprawną konfiguracją serwera. Ponadto w trakcie trwania audytu przeskanowano infrastrukturę w celu znalezienia podatnych oprogramowań / bibliotek, a także otwartych portów, które mogły mieć wpływ na bezpieczeństwo sieci.

W ramach audytu wykorzystano podejście polegające na testach manualnych oparte o wyżej wymienione metodyki, jak i wsparcie tych czynności szeregiem narzędzi automatycznych, m.in. Burp Suite Professional, Dirbuster, Nessus Professional, Nmap, sqlmap, ffuf.

Potwierdzono, że wszystkie błędy bezpieczeństwa znalezione w trakcie audytu bezpieczeństwa zostały naprawione i nie istnieje dalsza możliwość ich wykorzystania przez osoby trzecie.

Niniejszy dokument stanowi potwierdzenie wykonania audytu oraz zgodności ze standardami bezpieczeństwa OWASP TOP10 i OWASP ASVS

TestArmy

Attestation of executed security audit on the basis of OWASP ASVS L2 for:

Vercom S.A.

no 002/2025/20

ATTESTATION DETAILS

TestArmy Group S.A. confirms that made security audit and retests on application serwersms.pl:

1. panel.serwersms.pl (client panel)
2. api1.serwersms.pl (XML API)
3. api2.serwersms.pl (REST API)
4. 185.233.160.11(SMPP API)
5. serwersms.pl

including all functionabilities available on applications using black-box method following OWASP ASVS 4.0.3. methodology. Tests made mainly manually with support of automating tools.

All abovementioned applications showed the correct security on level 2 ASVS on day 05/02/2025.

ABOUT THE STANDARD ASVS AND LEVELS

The ASVS is a community-effort to establish a framework of security requirements and controls that focus on normalising the functional and non-functional security controls required when designing, developing and testing modern web applications.
The Application Security Verification Standard is a list of application security requirements or tests that can be used by architects, developers, testers, security professionals, and even consumers to define what a secure application is.

Using the Application Security Verification Standard ASVS has two main goals:
- to help organizations develop and maintain secure applications
- to allow security service, security tools vendors, and consumers to align their requirements and offerings Application Security Verification Levels

The Application Security Verification Standard defines three security verification levels, with each level increasing in depth.

- ASVS Level 1 is meant for all software.
- ASVS Level 2 is for applications that contain sensitive data, which requires protection.
- ASVS Level 3 is for the most critical applications - applications that perform high value transactions, contain sensitive medical data, or any application that requires the highest level of trust.

Each ASVS level contains a list of security requirements. Each of these requirements can also be mapped to security-specific features and capabilities that must be built into software by developers.

Level 1: Opportunistic

An application achieves ASVS Level 1 (or Opportunistic) if it adequately defends against application security vulnerabilities that are easy to discover, and included in the OWASP Top 10 and other similar checklists.
Level 1 is typically appropriate for applications where low confidence in the correct use of security controls is required, or to provide a quick analysis of a fleet of enterprise applications, or assisting in developing a prioritized list of security requirements as part of a multi-phase effort. Level 1 controls can be ensured either automatically by tools or simply manually without access to source code. We consider Level 1 the minimum required for all applications.
Threats to the application will most likely be from attackers who are using simple and low effort techniques to identify easy-to-find and easy-to-exploit vulnerabilities. This is in contrast to a determined attacker who

will spend focused energy to specifically target the application. If data processed by your application has high value, you would rarely want to stop at a Level 1 review.

Level 2: Standard

An application achieves ASVS Level 2 (or Standard) if it adequately defends against most of the risks associated with software today. Level 2 ensures that security controls are in place, effective, and used within the application. Level 2 is typically appropriate for applications that handle significant business-to-business transactions, including those that process healthcare information, implement business critical or sensitive functions, or process other sensitive assets.

Threats to Level 2 applications will typically be skilled and motivated attackers focusing on specific targets using tools and techniques that are highly practiced and effective at discovering and exploiting weaknesses within applications.

APPROACH TO VERIFICATION

Automatic tools

- Burp Proxy Professional - it's a set of many cooperating tools supporting the performance of penetration tests of web applications. On the one hand, it has some built-in automatisms improving repetitive works and, on the other hand, it provides the pentester with rather large area of freedom of actions. It is composed of the following modules:
    - http proxy (to hijack communication from the internet browsers or other communication software with the use of HTTP protocol);
    - repeater (to manually modify hijacked HTTP requests or to create any completely new HTTP requests);
    - scanner (automatic scanner detecting vulnerabilities in web applications);
    - intruder (http fuzzer to semi-automatic penetration tests);
    - sequencer (to analyze randomness of generated session identifiers);
    - decoder (to convert between popular encryptions - e.g. Base64, URL, HEX)
    - spider ("classic" http spider to automatically collect information on the resources used by a given application - subpage / style files / javascript files/ etc.)
    - comparer (to compare the differences between HTTP requests or replies)
- SOAP UI,
- DirBuster - tool for testing web applications to detect folders and files normally inaccessible from the website navigation level.
- Nmap - open source tool to explore the network and security audits. It was designed to quickly scan big networks; it also operates well with individual addresses. Nmap uses low-level IP packages - to

detect which addresses are accessible in the network, which ones provide access to services (application name and version), which operating systems they use (version system), what types of firewall systems are used and dozens of other features.

- SQLmap - automatic scanner of SQL Injection vulnerabilities
- Metasploit - tool used for penetration tests and for breaching ICT security systems
- Nessus Professional - tool to examine resources quickly, audit configuration, profile objectives, detect malicious software, sensitive data and many other. It scans operating systems, network devices, hyper supervisors, databases, web servers and sensitive infrastructure for vulnerabilities, threats and violations of compliance principles. With the world's biggest and constantly updated library of vulnerabilities and tests configuration as well as with the support of a team of experts for vulnerabilities this tool is a standard of speed and precision in scanning vulnerabilities.
- and our own programming framework

Their use results in more reliable scanning findings to compare and eliminate false alarms (false-positive). Furthermore, the use of many tools also reduces the risk of missing a security gap by one of the programs.

Manual Tests

Major part of the security tester's work consisted in manual verification of the website and checking vulnerabilities. The automatic tools do not detect e.g. logical errors or implemented functionality of application. Carrying out attacks manually provides a possibility of a better evasion/analysis of security filters implemented in the application and firewall systems.

Areas and actions of testing process

The penetration tests of applications/services were based on:

• Injections – SQL/XML injection, vulnerabilities unauthorised access to data. These include SQL, OS Shell, LDAP, XPath Injection. These errors occur when the specially made data is transmitted as part of the interpreter queries or commands (simplified: values are treated as commands). If these data are not are properly filtered, the attacker has the ability to call his own commands/questions. This may lead to accessing confidential or unauthorized information activity in the system.

• File include – unauthorized access (and read) to system files,

• Cross-Site Scripting (XSS) – allows you to execute a number of unauthorized actions in Web application, e. g. : Exchange of page content on the client page, execution of actions as another user or phishing attacks. Errors of this type occur when trying to web browser of the user's data without validating it beforehand. This allows the attacker to execute a script on the victim's rights. The result may be an intercept meeting the logged-in user, sharing the content of the website or forwarding the user to a page that contains other malicious scripts or software.

• Disclosure of information – most often about the system and the security arrangements which allows further attacks against an application or infrastructure,

• Weaknesses related to session management – e. g. capturing a session and accessing the website users, etc. ,

• Testing of authentication mechanisms,

• SSL/TLS vulnerability tests,

• Escalation tests – for errors in authorization systems,

• outdated software,

• Cross Site Request Forgery (CSRF) – allows a range of unauthorized actions in the web application. Attackers lure the victim to his side instructions placed on it, it can make requests on the service to which it is registered cictims. The purpose of the attack is to cause the user who is logged in to a website to launch a link that performs an action on the selected service that the attacker would not have access (e. g. no cookie).

• Testing the application for logical errors,

• Unsafe diversions,

• Testing the HTTP parameter pollution.

CONTACT DETAILS:

TestArmy Group S.A.

T: +48 505 372 810

E: szymon.chruscicki@testarmy.com