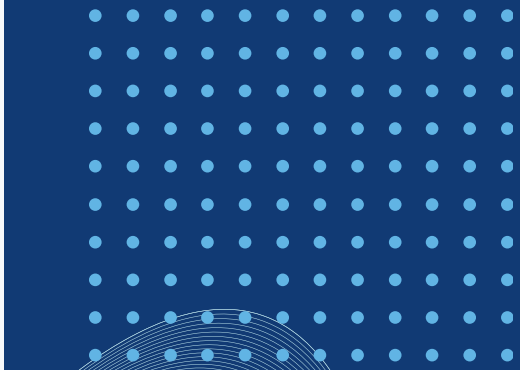


Codesealer

WHITE PAPER

OWASP TOP 10

HOW CODESEALER HELPS TO PROTECT AGAINST THE MOST COMMON VULNERABILITIES

Making the most of web application security is hard. Falling prey to vulnerabilities is unfortunately all too easy. Codesealer makes the choice clear.

BROUGHT TO YOU BY CODESEALER

Introduction

The Open Web Application Security Project (OWASP) is one of the most dedicated open-source projects, originating in the early 2000s. It aims to raise awareness and promote best security practices in the field. OWASP Top 10 is a prominent initiative that has evolved from an overview of penetration testing results to a standard awareness document. The first release of OWASP Top 10 occurred in 2003, a time when IT security and web services were still in the nascent stages. The most recent version of OWASP Top 10 was published in 2021.

The security of web applications is at utmost importance in today's digital landscape. Traditional methods of web security focus primarily on detecting and blocking threats, often leaving critical vulnerabilities exposed. Enter Codesealer, a pioneering force in the realm of cybersecurity.

Codesealer stands out as the industry's only security solution designed to protect both source code and APIs by encrypting both at the application layer. This innovative approach not only fortifies the defense mechanisms but also significantly reduces the attack surface, providing a robust and proactive layer of security.

Codesealer's technology goes beyond conventional security measures by ensuring that sensitive data and application logic remain encrypted even in the client environment. This forward-thinking strategy effectively mitigates risks before they can be exploited, offering unparalleled protection against a wide array of cyber threats.

A01:2021 – Broken Access Control

Broken Access Control refers to a security vulnerability where an application fails to properly enforce restrictions on what authenticated users can do, leading to unauthorized access. This flaw allows attackers to bypass access controls and perform actions that should be restricted, such as viewing or modifying data they are not permitted to access. Common ways this vulnerability is exploited include manipulating URLs, tampering with parameters, altering internal states, or using tools to modify API requests.

In 2019, First American Financial Corp, a major real estate and title insurance company, suffered a data leak exposing approximately 885 million sensitive records. The breach was due to a flaw in the company's website, where documents containing personal information such as Social Security numbers, bank account details, and financial statements were accessible without authentication. The IDOR vulnerability in the website's URL structure allowed unauthorized access to sensitive documents, leading to a significant data breach. The absence of adequate security measuresenabled attackers to exploit this flaw by sequentially changing the numerical IDs in the URLs.

How Codesealer Helps

Access control only works properly when it is enforced by secure server-side code or server-less APIs, where attackers cannot change the access control rules or data.

Codesealer encrypts payloads and consistently utilizes an opaque /x endpoint, effectively concealing API details and metadata. This makes it significantly harder for attackers to gain insights into the API's design and discover potential vulnerabilities. The critical information that could be used to bypass access controls or understand the application's internal workings is removed.

Regulatory fines

are a significant cost of a data breach resulting from broken access control

The General Data Protection Regulation (GDPR) in the EU and the California Consumer Privacy Act (CCPA) are regulatory frameworks that levy fines on organizations that fail to safeguard their customers' data.

A02:2021 – Cryptographic Failures

Cryptographic failures refer to vulnerabilities arising from the incorrect use or flawed implementation of cryptographic techniques. These issues include the use of weak encryption algorithms, poor key management practices, and insufficient protection of sensitive cryptographic materials, improper encryption of data in transit and storage.

In 2022, the Twitter APIs experienced a significant breach due to excessive data exposure. Attackers exploited this vulnerability to sell the information of 5.4 million users on a hacking forum. By January 2023, the situation escalated as attackers scraped and sold the public and private data of 400 million users on the dark web.

The flaw allowed attackers to verify if email addresses and phone numbers were linked to Twitter accounts. This breach exposed numerous users, including high-profile individuals like celebrities, politicians, and activists, to risks such as social engineering, targeted phishing attacks, and identity theft.

Cryptographic failures played a role in this breach, highlighting the importance of strong encryption algorithms, secure key management, and proper protection of cryptographic materials to prevent such vulnerabilities and protect user data.

How Codesealer Helps

Codesealer implements a multilayer defense strategy specifically designed to prevent cryptographic failures and enhance the security of web applications. It employs robust, standardized cryptographic techniques and extends these protections beyond the boundaries of TLS. By ensuring that cryptographic operations are strong and well-designed, Codesealer mitigates the risk of cryptographic failures. This comprehensive approach fortifies your web application's security, providing reliable protection for your data at all levels.

GDPR, PCI DSS

are some of the standards that you can violate with A02:2021

A single security solution is ineffective against Cryptographic Failures risk. That is why Codesealer offers a multi layer approach to ensure security of your digital assets.

A03:2021 – Injection

Injection flaws occur when untrusted data is sent to an interpreter, resulting in unexpected execution of commands or data manipulation. Examples include SQL injection and command injection.
Vulnerabilities arise when user data is not properly validated, dynamic queries lack context-aware escaping, or hostile data is used within search parameters. Injection types include SQL, NoSQL, OS command, ORM, LDAP, and EL/OGNL.

The Cl0p hacking group exploited a zero-day SQL injection vulnerability in Progress Software's MOVEit Transfer app, impacting over 1,000 organizations and 60 million individuals worldwide, including notable brands like British Airways and the BBC. Although Progress Software addressed the issue, reports of data breaches persisted, raising concerns about extensive corporate data exposure and future extortion threats highlighted by CISA. This incident underscores the critical importance of robust security measures to safeguard against such threats.

How Codesealer Helps

Codesealer secures your API by encrypting it, ensuring that it can only be accessed through a validated Codesealer session initiated by a real browser. This approach effectively blocks automated tools used by hackers to discover SQL injection vulnerabilities, safeguarding your application from hostile data manipulation and ensuring the integrity and security of your data and systems.

23,4% of all vulnerabilities

are SQL injections, making it the most frequent attack vector - Statista

Injection attacks were considered the most serious web application risk, taking the first place in OWASP Top 10 for many years in a row. Despite increased awareness and efforts to mitigate these vulnerabilities, they remain prevalent and continue to pose a significant threat to organizations worldwide.

A04:2021 – Insecure Design

Insecure design vulnerabilities arise from flaws in an application's architecture, such as inadequate threat modeling, absence of secure defaults, and overlooking security requirements during the design phase. Secure design is a cultural and methodological approach that continuously assesses threats and ensures that code is meticulously crafted and rigorously tested to preempt known attack methods. It's a proactive strategy ingrained in the development process, prioritizing security from the initial design stages to fortify applications against potential vulnerabilities.

Many popular Content Management Systems (CMS) platforms, like WordPress, overlook setting limits for unsuccessful login attempts on their admin panels. This oversight exposes them to brute force attacks, where hackers systematically try numerous login combinations until they gain unauthorized access. Since attackers can easily use automated tools to launch these assaults, the risk of compromise is high. To enhance defense against such threats, organizations often opt for third-party security extensions. These extensions add extra layers of protection by implementing stricter login controls and other security features. They play a crucial role in mitigating the risks associated with insecure design vulnerabilities found in CMS platforms and other web applications.

How Codesealer Helps

Codesealer mitigates the risk of insecure design by facilitating early integration with development teams.

Imagine this scenario: After extensive development, you conduct a vulnerability scan and realize a significant portion of your code needs rewriting. What if you could protect your application without any code changes.

Codesealer is a seamless reverse proxy, that integrates into existing architectures. It shields online services, protecting end users from attacks while preserving the integrity and confidentiality of sensitive data.

"Shift left"

is the new industry trend. Codesealer makes it happen.

According to the State of DevOps report, companies embracing best DevOps practices experience over a 50% decrease in change failure rates, even with a higher deployment frequency. This means top performers deploy changes multiple times a day, contrasting with bottom performers who deploy changes once every 6+ months.

A05:2021 – Security Misconfiguration

Security misconfigurations pose a significant risk when systems, frameworks, or applications are improperly set up, leaving them vulnerable to exploitation. This vulnerability arises from various factors such as default credentials, unnecessary services or features, and the absence of timely updates or patches. For instance, missing appropriate security hardening across any part of the application stack or improperly configured permissions on cloud services can lead to potential security breaches. It's essential to address security misconfigurations promptly to mitigate the risk of exploitation and ensure the robustness of the overall security posture.

T-Mobile disclosed a security breach on January 19, 2023, impacting approximately 37 million customers. The breach, lasting over six weeks from November 25th, 2022, allowed unauthorized access to a vulnerable API. As a result, sensitive customer information including names, emails, phone numbers, and birthdates was exposed. The compromise extended to account lines and service plans, affecting both prepaid and subscription customers. This incident highlights the critical necessity of robust API discovery measures to avert unauthorized access in a distributed ecosystem.

How Codesealer Helps

Codesealer offers robust protection against security misconfigurations, particularly concerning third-party tools. By safeguarding your source code and APIs, Codesealer effectively mitigates risks associated with these tools. Its implementation significantly reduces the impact of potential misconfigurations by making it difficult to exploit vulnerabilities through encryption.This encryption-based approach adds an extra layer of security, significantly mitigating the impact of potential misconfigurations and enhancing the overall resilience of the system.

23%

of cloud security incidents are a result of cloud misconfiguration - SentinelOne

Cloud resource misconfigurations pose significant concerns for public cloud organizations, often arising from errors during setup and deployment. Key misconfiguration issues include IAM misconfigurations, insecure API keys, inadequate security monitoring, and improper data backup practices.

A06:2021 – Vulnerable and Outdated Components

This risk arises from the usage of vulnerable, outdated, or unsupported third-party software within your web application. Vulnerable components pose a significant challenge as they are often difficult to test and assess for risk. This category is unique in that it lacks any Common Vulnerabilities and Exposures (CVEs) mapped to the included Common Weakness Enumerations (CWEs).

In December 2021, a Remote Code Execution (RCE) vulnerability was discovered in the widely-used Apache logging package Log4j2 versions 2.14.1 and below. This vulnerability was significant due to Log4j2's prevalence in online applications and services, with major providers like Amazon, Microsoft, IBM, and Google using it extensively. Exploiting the vulnerability required minimal expertise and could lead to severe consequences, including data theft, malware installation, and even ransomware attacks. Microsoft reported state-sponsored actors and hackers capitalizing on this vulnerability. The Apache Software Foundation released updates to address the vulnerability, but eradicating the threat will be a long-term effort due to Log4j2's widespread use. In addition to applying patches and updates, organizations are advised to conduct compromise assessments and penetration testing to safeguard against potential exploitation.

How Codesealer Helps

Codesealer effectively manages the risks associated with vulnerable, unsupported, or outdated software components by securing all aspects of your source code, including third-party tools. Through comprehensive encryption of all data transmissions, Codesealer adds an additional layer of security, ensuring sensitive information remains protected, even if vulnerable or outdated components are present in the application. This approach conceals the presence of such components, making it challenging for attackers to identify and exploit potential vulnerabilities.

Open doors?

Outdated components in software are like open windows inviting security risks.

Identifying vulnerabilities in outdated or vulnerable components is notoriously challenging. These vulnerabilities can exist for years before they are discovered and patched. At the same time, the consequences of using vulnerable and outdated components can be severe.

A07:2021 – Identification and Authentication Failures

Identification and authentication failures pertain to vulnerabilities stemming from weaknesses in user authentication mechanisms. These vulnerabilities include inadequate password policies, insufficient authentication factors, and susceptibility to brute-force attacks. Weaknesses in user authentication mechanisms introduce the risk of weak passwords, absence of additional login security settings, and vulnerabilities in session identifiers. These issues collectively undermine the effectiveness of the authentication process, potentially enabling unauthorized access to sensitive systems and data.

In 2021, one of the most significant data breaches in history occurred with the LinkedIn API breach. Personal records of over 700 million users, constituting 92% of the user base, were illicitly obtained from the platform and made available for sale on a hacker forum. The breach occurred due to the discovery of a public API lacking proper authentication measures, allowing attackers to access and scrape user content without authorization.

How Codesealer Helps

Codesealer plays a crucial role in preventing breaches like the LinkedIn API incident by encapsulating all API endpoints behind a single encrypted access point, accessible exclusively through a validated Codesealer session. This approach ensures that sensitive session data, including identifiers, remains secure against interception or tampering by malicious actors.

49M

of Americans were victims of identity fraud in 2021

By encrypting URLs, Codesealer strengthens the security of web applications, effectively mitigating the risk of session hijacking and unauthorized access to user sessions. This encryption-centric approach bolsters the confidentiality and integrity of session data, thereby offering users a more secure browsing experience overall.

A08:2021 – Software and Data Integrity Failures

The risk of software and data integrity failures arises when applications rely on plugins, libraries, or modules from untrusted sources, repositories, and content delivery networks (CDNs). These dependencies can introduce vulnerabilities if they are compromised or tampered with, leading to unauthorized modifications or deletions of software and data. Such failures often result from inadequate validation checks, insufficient data integrity controls, or insecure storage mechanisms. A new category for 2021 highlights the dangers of making assumptions related to software updates, critical data, and CI/CD pipelines without verifying their integrity. This category is significant, with vulnerabilities that carry substantial impact ratings according to Common Vulnerability and Exposures (CVE) and Common Vulnerability Scoring System (CVSS) data.

A notable example is the SolarWinds cyber attack in 2020. SolarWinds provides the Orion platform used by many Fortune 500 companies and government agencies. Hackers inserted malicious code into Orion, which SolarWinds unknowingly distributed in updates. These compromised updates created backdoors, allowing attackers to install additional malware and spy on affected organizations. This incident highlights the importance of verifying the integrity of software updates to prevent such breaches.

How Codesealer Helps

Codesealer offers robust protection against vulnerabilities by encrypting and securing all data transmissions, including those involving third-party tools. This encryption shields sensitive information from potential exploits within these components. While our protection against supply chain attacks varies depending on the situation, Codesealer's encryption obscures data structure, preventing attackers from introducing malicious code via insecure deserialization.

$4.45M

is the average cost of a data breach in 2023

According to a recent report by Data Theorem, a staggering 91% of organizations were affected by a software supply chain attack within the past year, highlighting the pervasive nature of this threat across various industries.

A09:2021 – Security Logging and Monitoring Failures

Security logging and monitoring failures arise from a lack of effective logging and monitoring tools. This category moved up from tenth position in the OWASP Top 10 of 2017 to third in the 2021 community survey, highlighting its increasing importance.
Testing for logging and monitoring issues can be challenging, often requiring interviews or confirmation of attack detection during penetration tests. Although there is limited CVE/CVSS data for this category, the ability to detect and respond to breaches is crucial. Effective logging and monitoring are essential for accountability, visibility, incident alerting, and forensic analysis.

Security logging and monitoring failures arise from insufficient or ineffective practices, undermining the ability to detect and respond to incidents. Common issues include insufficient log generation, limited coverage, ineffective aggregation and storage, neglected analysis and alerting, lack of integration with incident response, and inadequate third-party monitoring. These failures can lead to undetected breaches, stolen data, compliance issues, reputational damage, and financial losses. To mitigate these risks, organizations should develop a comprehensive plan, establish centralized log management, implement automated event correlation, deploy real-time monitoring and alerting, ensure secure log storage, and conduct regular log analysis. Addressing these issues can significantly improve security posture and response capabilities.

How Codesealer Helps

Codesealer effectively mitigates security logging and monitoring failures by providing robust and comprehensive logging and monitoring capabilities.

Codesealer ensures comprehensive logging of all requests, including detection of blocked or suspicious activity. By forwarding relevant data to the organization's SIEM for monitoring, Codesealer enables prompt detection and response to security incidents, bolstering overall security posture and response capabilities.

Ranks high

due to its significant impact

Logging and monitoring are vital for detecting security incidents, investigating breaches, and ensuring compliance with regulations. They provide real-time insights into system activities, enabling rapid response to threats and minimizing the impact of cyberattacks on business operations.

A10:2021 – Server-Side Request Forgery (SSRF)

Server-Side Request Forgery (SSRF) flaws occur when a web application fetches a remote resource without properly validating the user-supplied URL. This vulnerability allows attackers to coerce the application into sending crafted requests to unexpected destinations, bypassing firewall or network access controls.
As modern web applications offer convenient features for end-users, the incidence of SSRF is on the rise. Additionally, the severity of SSRF is increasing due to the widespread use of cloud services and the complexity of architectural designs.

The 2019 Capital One breach exposed sensitive personal and financial data of millions of customers, affecting 100 million Americans and 6 million Canadians. Orchestrated by a former AWS engineer, the breach exploited misconfigured AWS accounts to access Capital One's systems. This resulted in the theft of 80,000 bank account numbers and 140,000 US social security numbers. Capital One faced significant fines and settlements, including an $80 million fine from the US OCC and a $190 million settlement for customer lawsuits. The breach was made possible by misconfigured firewalls and involved a server-side request forgery (SSRF) attack on AWS infrastructure.

How Codesealer Helps

Codesealer addresses the risk of Server-Side Request Forgery (SSRF) by removing the APIs from the attack surface. To further bolster defenses against SSRF attacks, developers can implement a multi-layered approach.

100,000

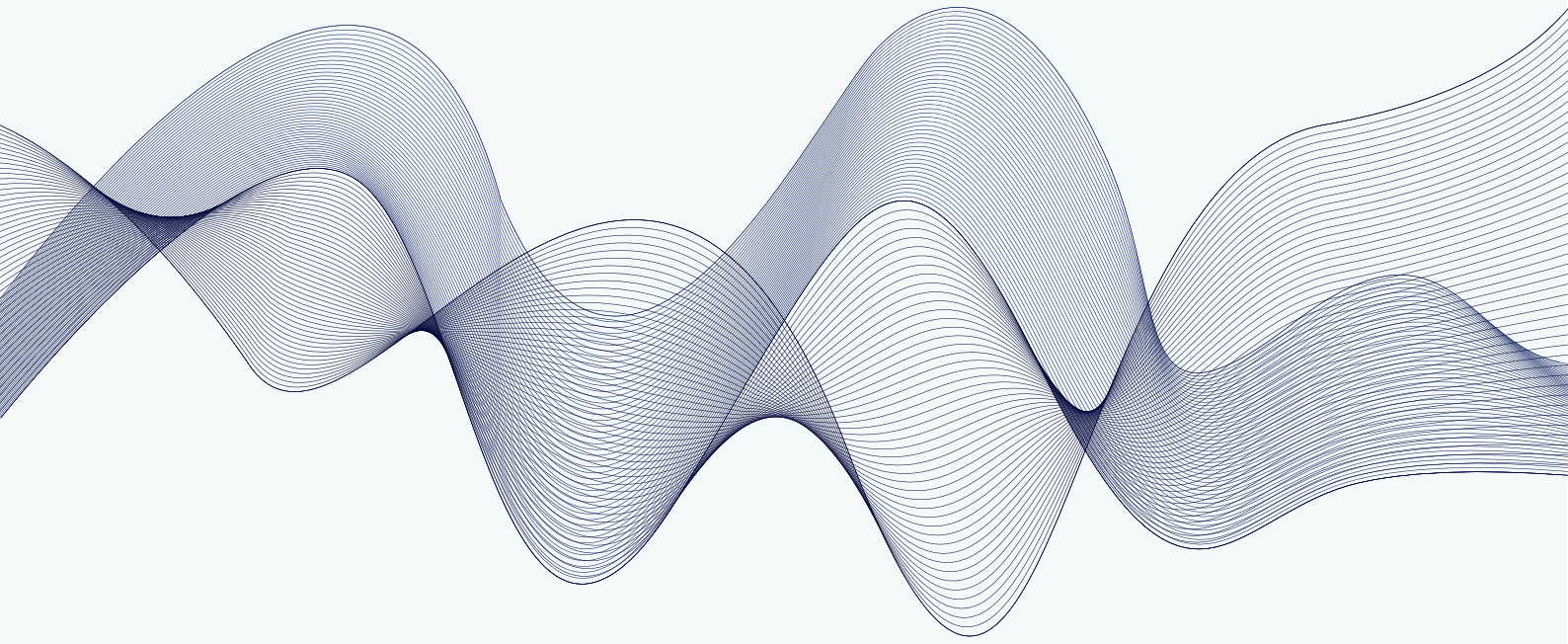
businesses are hit by SSRF attacks every 3 months

Implementing strict URL validation and disabling HTTP redirections at the application layer can effectively prevent SSRF exploitation. However, it's important not to rely solely on deny lists or regular expressions for SSRF mitigation, as attackers can often find ways to bypass these measures.

Ready to seal your API?

Reach out to our team today to learn more about Codesealer's API protection features and discover how we can fortify your web applications against evolving cyber threats. Schedule a consultation or request a demo to witness the transformative impact of Codesealer firsthand.

info@codesealer.com
Codesealer A/S
Njalsgade 76, 3rd Floor
2300 Copenhagen S
CVR 39228920




Codesealer