imperva
a Thales company

How Imperva Protects Against the OWASP API Security Top Ten Risks

Introduction

In today's digital landscape, APIs (Application Programmable Interfaces) are indispensable for driving enterprise transformations, whether within cloud-native microservice architectures or legacy non-cloud environments. However, with this increasing reliance on APIs comes a heightened risk of cyber threats. Malicious actors are capitalizing on our growing reliance on APIs, constantly evolving their attack methods to exploit vulnerabilities with the intention of either manipulating the API's business logic to disrupt business or to exfiltrate data for malicious ends. As API usage continues to expand, robust security measures are imperative to protect both public and private APIs, along with the sensitive data they manage.

The **OWASP API Security Top 10**, revised in 2023, provides a comprehensive guide to the critical issues that organizations must tackle to ensure the robust security of their APIs. Among the vulnerabilities highlighted, Broken Object Level Authorization (BOLA) stands out as a top priority and a major challenge for security teams.

This eBook outlines the OWASP API Security Top 10 risks, their manifestations, and how Imperva API Security, when combined with our robust platform of Application Security products, effectively protects against the OWASP API Security Top 10.

The OWASP API Security Top 10

- **API1:2023 Broken Object Level Authorization**
  Sometimes also considered as Insecure Direct Object Reference (IDOR), BOLA arises from APIs exposing object identifiers through their endpoints, introducing significant Object Level Access Control risks.

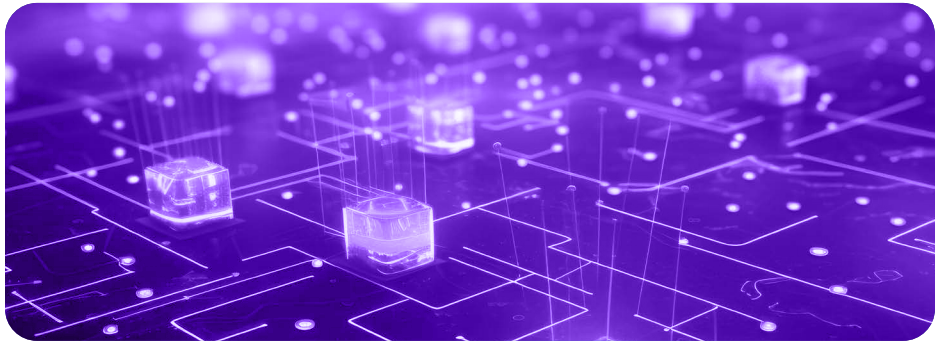
- **API2:2023 Broken Authentication**
  Vulnerabilities in authentication mechanisms that can lead to unauthorized access.

- **API3:202 Broken Object Property Level Authorization**
  Combining risks of Excessive Data Exposure and Mass Assignment, this vulnerability poses threats that can lead to the exposure and/or unauthorized manipulation of properties of important data objects.

- **API4:2023 - Unrestricted Resource Consumption**
  There are risks associated with APIs not imposing proper limitations on resource usage, leading to potential exploitation.

- **API5:2023 - Broken Function Level Authorization**
  Concerns related to inadequate authorization checks at the function level, enabling unauthorized access to functionalities.

- **API6:2023 - Unrestricted Access to Sensitive Business Flows**
  Vulnerabilities allowing unauthorized access to critical business processes normally running in the backend and shielded from external exposure.



- **API7:2023 - Server Side Request Forgery**
  The risk of attackers manipulating front-end services to post illegitimate requests to back-end services, exposing back-end services that are otherwise inaccessible.

- **API8:2023 - Security Misconfiguration**
  Issues arising from misconfigured security settings expose APIs to potential exploitation.

- **API9:2023 - Improper Inventory Management**
  Challenges related to inadequate tracking and management of API assets.

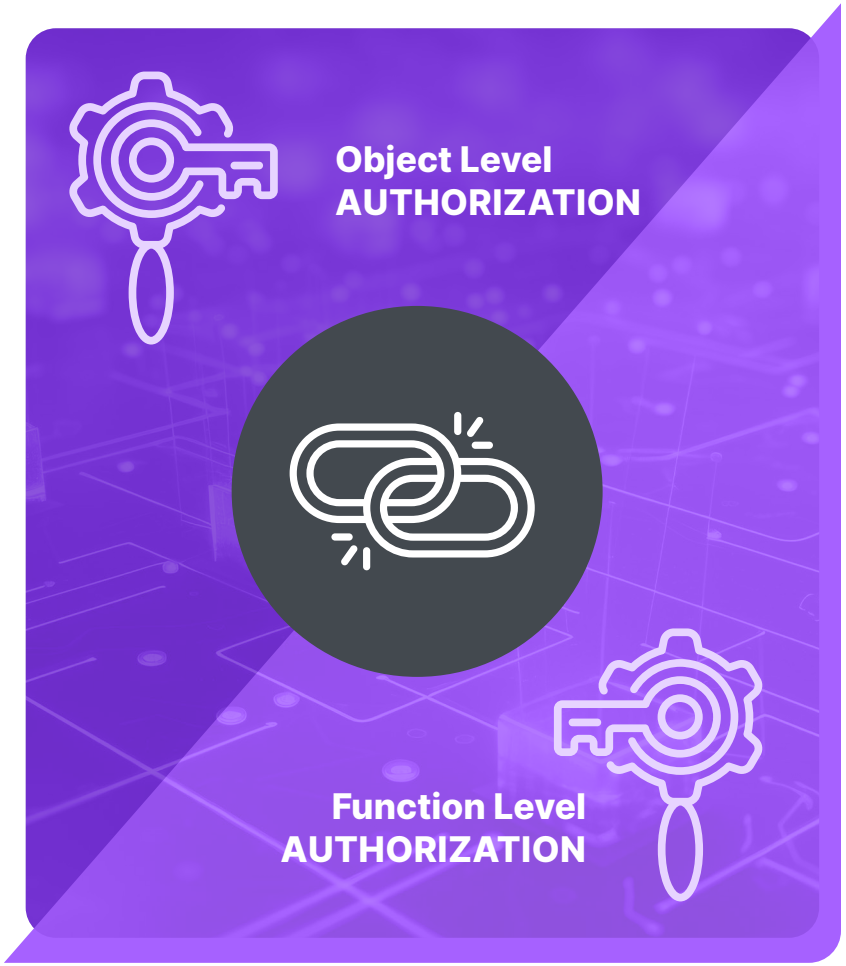
- **API10:2023 - Unsafe Consumption of APIs**
  Risks associated with improper utilization and handling of APIs, leading to potential vulnerabilities.

BOLA/BOFLA Vulnerabilities (A1, A5)

Broken Object Level Authorization/Broken Function Level Authorization (BOLA/BFLA) takes advantage of the lack of proper authorization in an API. Bad actors often use bots to scan for BOLA/BFLA vulnerabilities. BOLA/BFLA abuses targeting API business logic implementation vulnerabilities are often very difficult to detect as they are application-specific. Without well-defined attack patterns, abusive requests are indistinguishable from normal requests.

API Security risk assessment, which is done automatically as part of the API Discovery process, tracks API data exchange patterns to identify API endpoints potentially at risk of BOLA abuse. The risk assessment is automated and with reasonable levels of accuracy.

Our research shows that actual BOLA/BFLA abuse attempts often start scanning for vulnerable endpoints using automated attacks. Applying an anti-bot policy to the risk API endpoint will protect it from potential abuse.



Broken Authentication (A2)

OWASP API Security Broken Authentication Risks

Broken authentication can occur when API tokens are generated from a stolen account due to an Account Takeover (ATO) incident. Additionally, misuse of API tokens directly can also pose authentication risks.

Imperva API Security identifies these risks through its Risk Assessment feature, which runs as part of the API Discovery process. Unauthenticated API endpoints are reported as a potential vulnerability.

To further mitigate these threats, Imperva offers Account Takeover Protection (ATO), safeguarding against unauthorized access attempts and account compromises that can lead to the generation of API tokens that, in turn, allow bad actors access to the API. The combined solution of API Security and ATO Protection safeguards API endpoints from Broken Authentication threats.

Broken Object Property Level Authorization (A3)

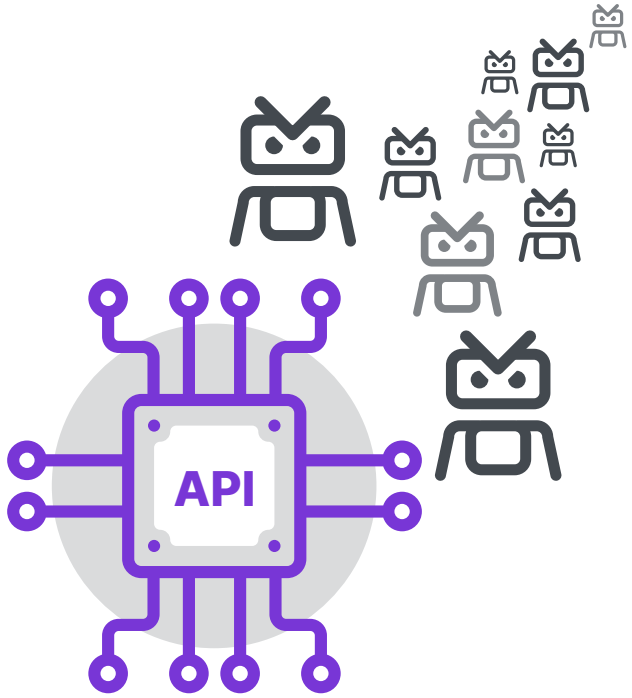
Broken Object Property Level Authorization (BOPLA) includes threats that were recognized in the Excessive Data Exposure and Mass Assignment categories in the previous OWASP API Security Top 10 list. Excessive Data Exposure threats refer to anomalies where an API call returns an excessive amount of sensitive data.

Mass Assignment threats usually manifest themselves in API calls with parameters that are not normally used.

Imperva API Security supports proactive enforcement to alert or block unexpected new parameters.

Unrestricted Resource Consumption (A4)

Unrestricted Resource Consumption (A4) can manifest through various channels, notably excessive bot traffic inundating the API. Such traffic can lead to resource exhaustion, diminishing the availability and performance of the API, thereby disrupting legitimate user access and potentially causing service outages. To counter this threat, Imperva employs **Advanced Bot Protection (ABP)**, a sophisticated defense mechanism designed to identify and mitigate malicious bot activity. Imperva integrates API Security with ABP, enabling the mapping of 'at-risk' endpoints to ABP policies tailored to detect and block suspicious bot traffic. By leveraging a combination of ABP and API Security capabilities, Imperva supports uninterrupted service availability and safeguards against the adverse effects of unrestricted resource consumption.



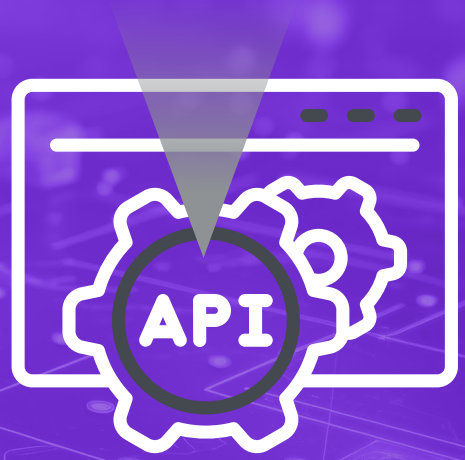
Excessive Bot Traffic

Unrestricted Access to Sensitive Business Flows (A6)

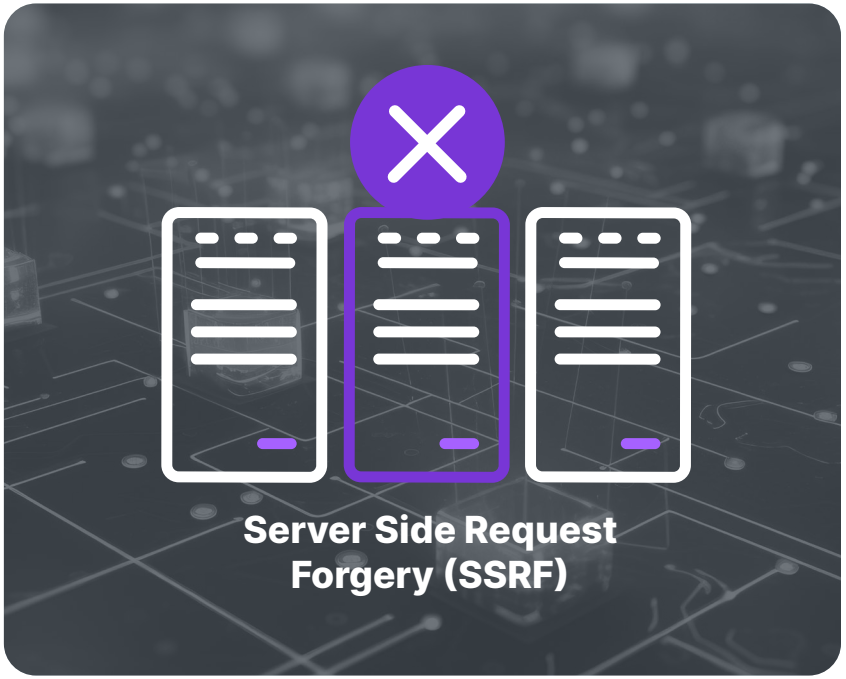
Unrestricted Access to Sensitive Business Flows can manifest in various ways within the realm of application security, particularly concerning the handling of sensitive user data and transactions. One common manifestation is through unauthorized access to critical business processes or functionalities, allowing malicious actors to exploit vulnerabilities and manipulate sensitive data for fraudulent activities. Additionally, insufficient access controls and improper authentication mechanisms can exacerbate the risk, enabling unauthorized users to bypass security measures and gain unrestricted access to sensitive business flows. Such breaches not only compromise the integrity and confidentiality of user data but also undermine trust in the application's security posture.

With Imperva's **Online Fraud Prevention Solutions (OFP)** capabilities, organizations can proactively detect and mitigate potential threats to sensitive business flows. OFP employs advanced fraud detection algorithms and real-time monitoring to identify suspicious activities, enabling organizations to swiftly intervene and prevent fraudulent transactions.

Imperva also offers an extensive data security solution, in conjunction with user tracking capabilities leveraging a Runtime Application Self Protection (RASP) agent. These solutions further enhance an organization's capability to track and protect sensitive data flows, especially those in the backend.



Server Side Request Forgery (SSRF) (A7)



Recognizing Server Side Request Forgery (SSRF)

Server Side Request Forgery (SSRF) presents a significant threat that manifests when attackers manipulate the application server to issue server-side requests to the backend, exploiting vulnerabilities to access sensitive data or perform unauthorized actions. Recognizing SSRF can be challenging as the attacks often occur within the application's internal network, making them difficult to detect using traditional security measures. However, indicators such as unexpected or unauthorized network requests originating from the application server can signal the presence of SSRF attacks. Additionally, anomalies in server logs or unusual patterns of data access may also indicate malicious activity, prompting further investigation to confirm the presence of SSRF vulnerabilities.

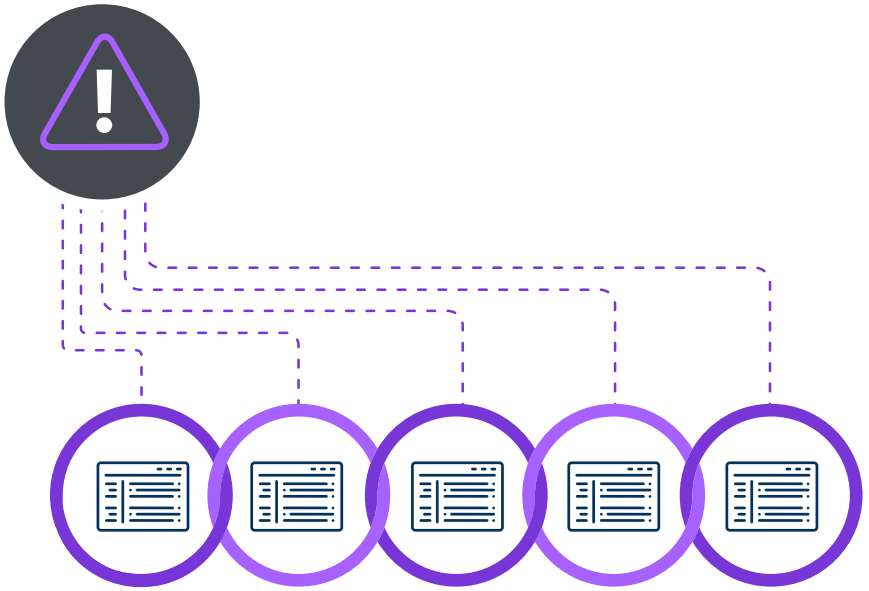
To protect against the risk of SSRF, Imperva offers robust security solutions designed to safeguard applications from malicious attacks. Imperva Cloud WAF, in particular, provides advanced threat detection and prevention capabilities, including the ability to block SSRF attacks in real-time. By leveraging sophisticated algorithms and behavioral analysis, Imperva Cloud WAF can identify and mitigate SSRF attempts, preventing attackers from exploiting vulnerabilities and accessing sensitive backend resources.

Security Misconfiguration (A8)

Security Misconfiguration can manifest in several ways, posing significant threats to the integrity and confidentiality of sensitive data. Common indicators of security misconfiguration include issues such as security tunnel misconfigurations and flawed design in the API specification. Security tunnel misconfigurations may result in unintended access to restricted resources or improper handling of sensitive data, while bad security design in the API specification can create vulnerabilities that expose the application to exploitation by malicious actors. Recognizing these risks requires thorough auditing of the application's configuration settings, API design, and network architecture to identify any misconfigurations or design flaws that may compromise security.

Imperva offers robust security solutions to mitigate the risks of Security Misconfiguration, safeguarding applications against potential vulnerabilities and exploitation. Imperva Web Application Firewall (WAF) can ensure proper security configurations of the original server by enforcing proper certificate management both for servers and for clients.. Additionally, Imperva's **API Verification** tool provides advanced security risk assessment capabilities, enabling organizations to analyze API specifications and identify potential misconfigurations or design flaws that may pose security risks. As API specifications are often used to configure an API server or gateway, such API specification security audits can significantly reduce the chance of server-side misconfigurations for APIs.

Improper Asset Management (A9)



Improper Asset Management presents a critical challenge concerning the management and lifecycle of APIs. This risk often manifests through various indicators, such as the resurfacing of deprecated APIs known as shadow APIs following their deprecation. Additionally, poorly designed APIs that allow for variable paths and parameter keys can exacerbate the risk, complicating asset management and increasing the likelihood of security vulnerabilities. Recognizing the presence of improper asset management requires thorough monitoring and auditing of API endpoints, configurations, and usage patterns to identify any instances of shadow APIs or design flaws that may compromise security.

Imperva offers comprehensive solutions to mitigate the risks associated with Improper Asset Management, safeguarding applications against potential vulnerabilities and unauthorized access. Leveraging **Imperva API Discovery**, organizations can proactively identify and assess the risk of resurfacing deprecated API endpoints and design flaws within their API infrastructure. By alerting organizations to the presence of shadow APIs and design vulnerabilities, Imperva enables proactive measures to be taken, such as endpoint deprecation or redesign, to mitigate the risk effectively. Through continuous monitoring and analysis of API usage patterns, Imperva helps ensure that necessary security measures are in place to protect against the risks of improper asset management, preserving the integrity and confidentiality of sensitive data while bolstering overall security posture.

Unsafe Consumption of APIs (A10)

Modern applications often make calls to third-party APIs to perform certain functions. These third-party APIs are sometimes out of the control of the organization. Unchecked use of APIs outside of an organization's control can be a security risk especially when sensitive data is exchanged.

Imperva API Security supports an "Anywhere" option for API sensors to be deployed to inspect a customer's internal APIs. For API calls that are inspected by the Anywhere API sensor, Imperva API security can apply the same discovery and risk assessments to track the consumption of APIs.

Conclusion and Best Practices

As the digital landscape continues to evolve, securing APIs is critical in safeguarding sensitive data and maintaining secure operations for customers and stakeholders. Recognizing the risks highlighted by the OWASP API Security Top 10 enables organizations to add resilience to their API security posture by implementing effective mitigation strategies essential to protect against potential exploits and data breaches.

Best Practices For Ensuring The Robust Security of Your APIs

1. Leverage Imperva API Security to Discover and classify all APIs in your inventory.
2. Use continuous discovery to maintain an always up-to-date API inventory.
3. Identify and protect sensitive and high-risk APIs Leveraging advanced threat detection and prevention solutions, such as Imperva Cloud WAF and Advanced Bot Protection (ABP).
4. Perform risk assessments on vulnerable API endpoints.
5. Establish a robust monitoring system to analyze API usage patterns and detect suspicious activity.
6. Adopt a comprehensive approach to API Security leveraging the strength of your full application security.
7. Educate development teams and stakeholders on best practices for secure API design, implementation, and management.
8. Stay informed about emerging threats and security vulnerabilities, and proactively update security measures to address evolving risks.

By making API security a top priority and taking proactive steps to tackle risks head-on, organizations can ramp up their overall security and keep potential API abuse at bay, safeguarding sensitive data and keeping unauthorized intruders out.

Taking a holistic approach to API security not only boosts your overall resilience against threats but also puts a stop to bad actors trying to abuse critical API business logic or steal data.

Imperva is an analyst-recognized, **cybersecurity** leader championing the fight to **secure data and applications** wherever they reside.

imperva.com
+1.866.926.4678