wallarm | ENTERPRISE API SECURITY

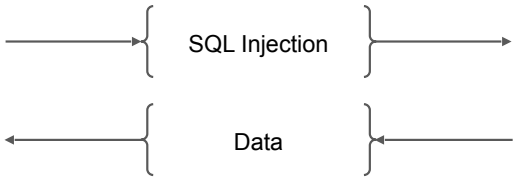
The OWASP Business Logic Abuse Top 10

Johanns Quiroz
Sr. Solutions Architect

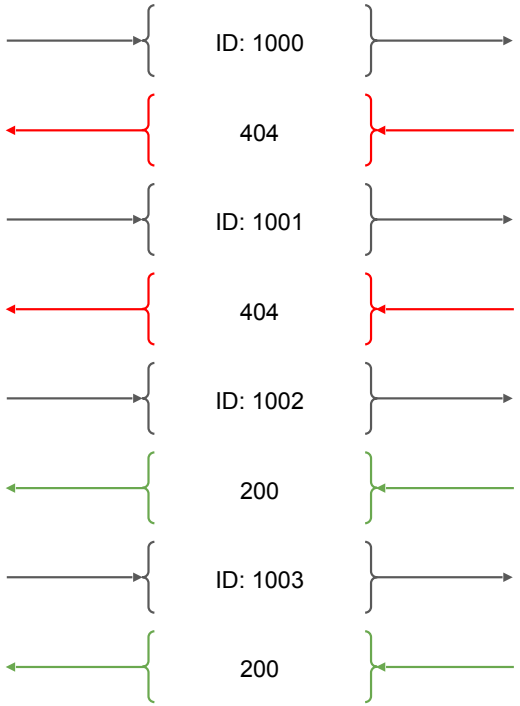
wallarm | ENTERPRISE API SECURITY

What is a business logic attack?

Stateless Attack



Stateful Attack



A pattern across multiple requests

Different Types of Flaws

Flaw in Code

Programming errors or insecure use of libraries.

Examples:

- SQL Injection (unsanitized inputs reaching a database).
- Cross-Site Scripting (unescaped user input rendered in HTML).
- Buffer overflows or memory corruption.
- Hardcoded secrets or misconfigured security headers.

Root Cause: Developer mistakes in syntax, validation, or API use.

Flaw in Business Logic

Application functions as designed, but the design enables abuse.

Examples:

- Redeeming a “one-time” coupon multiple times.
- Skipping steps in a multi-factor workflow.
- Exploiting expired session tokens that still grant access.
- Bypassing rate limits to drain resources.

Root Cause: Missing or flawed enforcement of business rules and state transitions.

low Common are Business Logic Attacks?

{ } +20%

API vulnerabilities

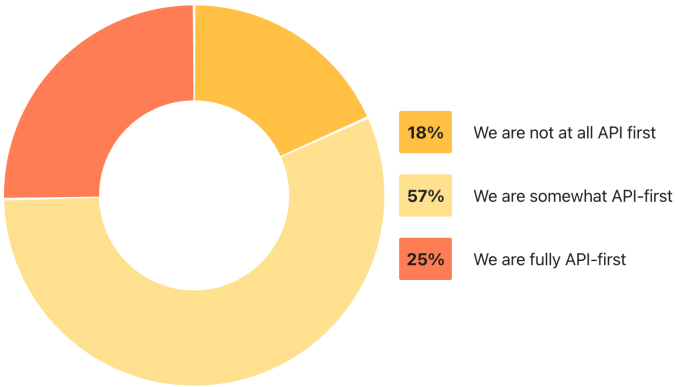
+30%

Business Logic Incidents

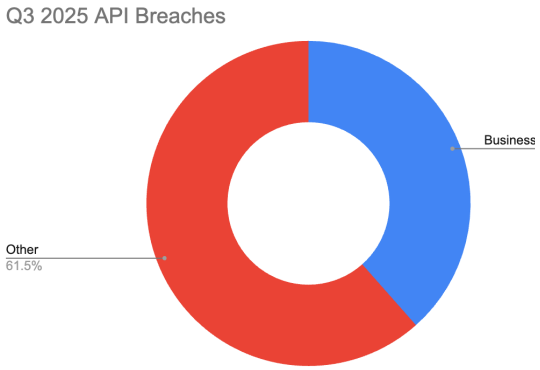
Q2 → Q3

Wallarm Q3 2025 API ThreatStats Report

82% of organizations now identify as API-first.



Postman 2025 State of the API Report



Wallarm Q3 2025 API ThreatStats Report


wallarm
he Universe of
WASP Top 10 Lists
Attack Surface Management
Low-Code/No-Code
Cloud-Native AppSec
Mobile
Non-Human Identities
LLMs
Desktop AppSec
erless
DevSecOps
Maritime
Client-Side Security
CI/CD
OWASP
Drone Security
Security
BLA
Smart Contract
Privacy Risks
OSS
MCP
Docker
Machine Learning
XR

Why does a business logic abuse Top 10 matter?

- Limited coverage of business logic as a concept, but …
  - Business logic abuse increasing in the real world.
  - Technology changes, but business logic abuse persists
  - Agentic AI and GenAI apps are collections of business logic
- Therefore …
  - It’s time to focus developers on business logic issues
  - It’s time to focus security on identifying and defending business logic issues

wallarm | ENTERPRISE API SECURITY

The Business Logic Abuse Top 10

'he BLA Top 10 Methodology

Grounded in Theory	Use of a Turing Machine model to guarantee that all possible business logic flaws can be represented in the taxonomy.
Grounded in Data	Validation against 76k CVEs and 25k GitHub issues ensures the categories are not academic abstractions.
Focused Scope	Filtered out other OWASP Top 10 categories; this Top 10 is truly about business logic abuse.
Iterative Refinement	The taxonomy wasn't defined only once. It was tested, clustered, re-tested, and tuned for both precision and coverage.

wallarm | ENTERPRISE API SECURITY

OWASP Business Logic Abuse Top 10

BLA1:2025 - Action Limit Overrun (ALO)	Re-using “one-time” actions (like coupons or refunds) again and again because the system doesn’t lock them after the first use.	BLA6:2025 - Missing Transition Validation (MTV)	Calling later workflow steps directly because the app doesn’t re-check that you satisfied the earlier requirements.
BLA2:2025 - Concurrent workflow order bypass (CWOB)	Skipping ahead in a multi-step process (e.g., finishing before the required earlier steps complete) by racing requests out of order.	BLA7:2025 - Resource Quota Violation (RQV)	Hammering a feature too fast or too often (vote spamming, heavy tasks) to get an edge or degrade the service.
BLA3:2025 - Object state manipulations (OSM)	Sending sneaky values so the app quietly flips hidden settings (like roles or status) it shouldn’t let you change.	BLA8:2025 - Internal State Disclosure (ISD)	Error messages, codes, or timing differences leak what’s happening inside, helping attackers plan targeted abuse.
BLA4:2025 - Malicious Logic Loop (MLL)	Triggering a process that never properly stops (or repeats too much) to chew up time, money, or system resources.	BLA9:2025 - Broken Access Control (BAC)	The app doesn’t properly check permissions in key business actions, so people do things they’re not allowed to do.
BLA5:2025 - Artifact Lifetime Exploitation (ALE)	Using “short-lived” things (tokens, sessions, temporary files) after they should have expired because the app didn’t actually retire them.	BLA10:2025 - Shadow Function Abuse (SFA)	Abusing forgotten or hidden functions (test utilities, internal endpoints) left available in production.

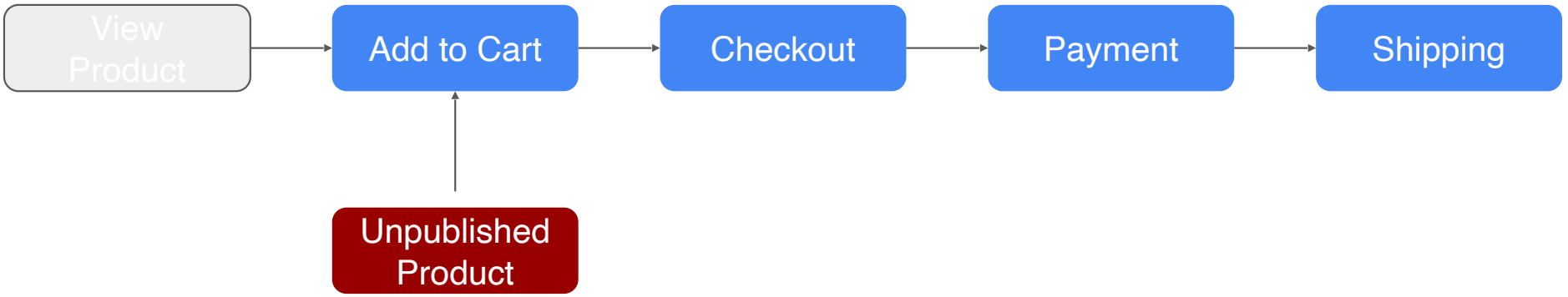
wallarm | ENTERPRISE API SECURITY

)WASP Business Logic Abuse Top 10

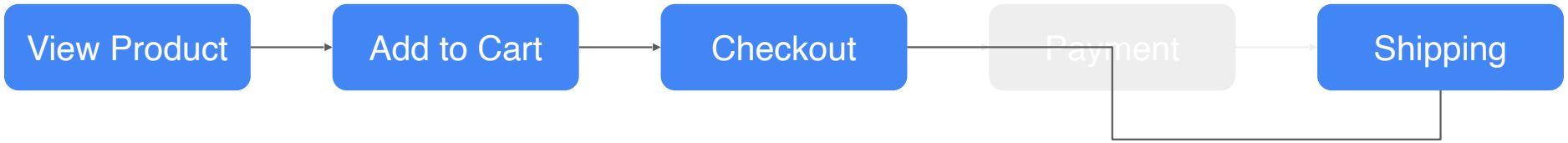
BLA1:2025 - Action Limit Overrun (ALO)	Re-using “one-time” actions (like coupons or refunds) again and again because the system doesn’t lock them after the first use.	BLA6:2025 - Missing Transition Validation (MTV)	Calling later workflow steps directly because the app doesn’t re-check that you satisfied the earlier requirements.
BLA2:2025 - Concurrent workflow order bypass (CWOB)	Skipping ahead in a multi-step process (e.g., finishing before the required earlier steps complete) by racing requests out of order.	BLA7:2025 - Resource Quota Violation (RQV)	Hammering a feature too fast or too often (vote spamming, heavy tasks) to get an edge or degrade the service.
BLA3:2025 - Object state manipulations (OSM)	Sending sneaky values so the app quietly flips hidden settings (like roles or status) it shouldn’t let you change.	BLA8:2025 - Internal State Disclosure (ISD)	Error messages, codes, or timing differences leak what’s happening inside, helping attackers plan targeted abuse.
BLA4:2025 - Malicious Logic Loop (MLL)	Triggering a process that never properly stops (or repeats too much) to chew up time, money, or system resources.	BLA9:2025 - Broken Access Control (BAC)	The app doesn’t properly check permissions in key business actions, so people do things they’re not allowed to do.
BLA5:2025 - Artifact Lifetime Exploitation (ALE)	Using “short-lived” things (tokens, sessions, temporary files) after they should have expired because the app didn’t actually retire them.	BLA10:2025 - Shadow Function Abuse (SFA)	Abusing forgotten or hidden functions (test utilities, internal endpoints) left available in production.

Missing Transition Validation (MTV)

Missing Transition Validation (MTV)



Missing Transition Validation (MTV)

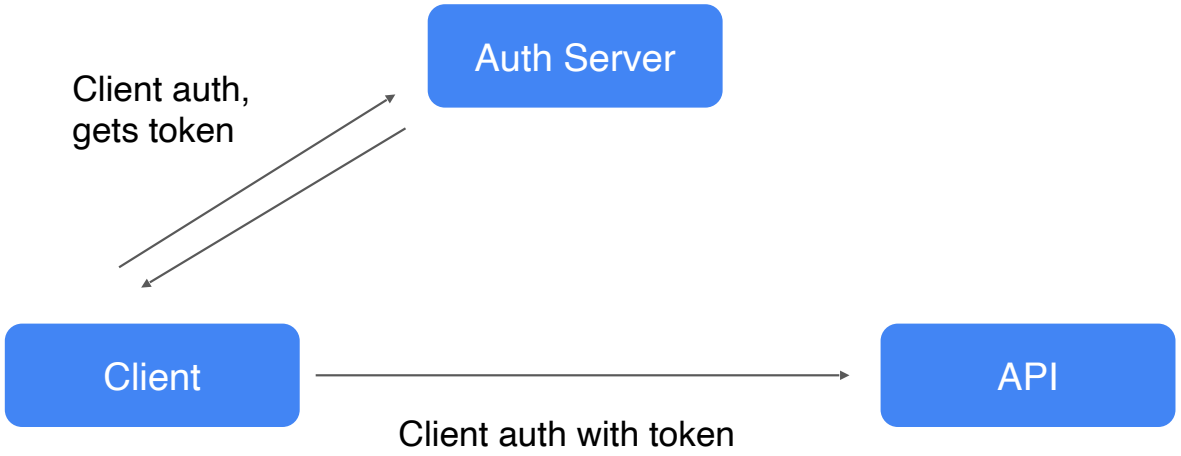


wallarm | ENTERPRISE API SECURITY

)WASP Business Logic Abuse Top 10

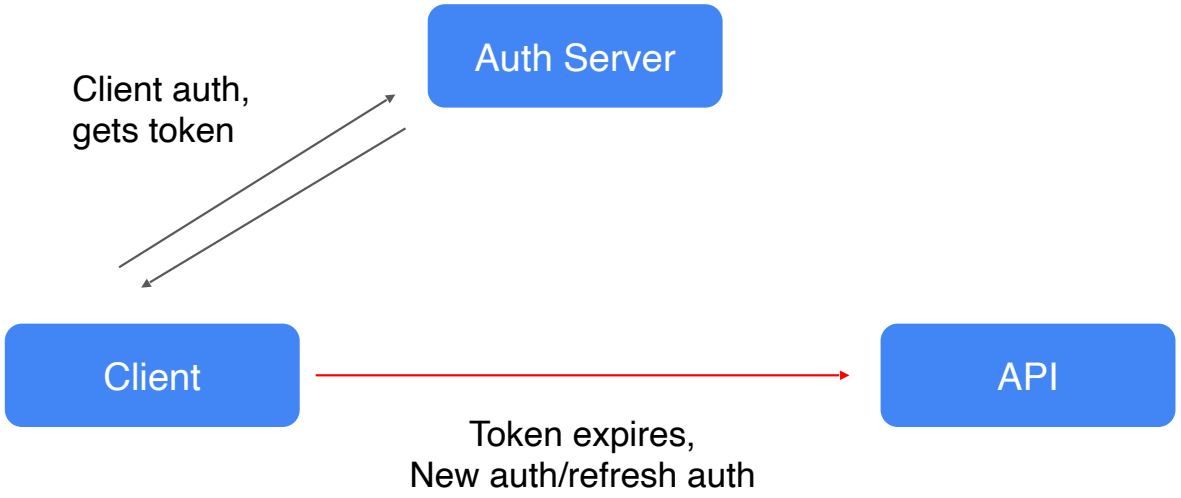
BLA1:2025 - Action Limit Overrun (ALO)	Re-using “one-time” actions (like coupons or refunds) again and again because the system doesn’t lock them after the first use.	BLA6:2025 - Missing Transition Validation (MTV)	Calling later workflow steps directly because the app doesn’t re-check that you satisfied the earlier requirements.
BLA2:2025 - Concurrent workflow order bypass (CWOB)	Skipping ahead in a multi-step process (e.g., finishing before the required earlier steps complete) by racing requests out of order.	BLA7:2025 - Resource Quota Violation (RQV)	Hammering a feature too fast or too often (vote spamming, heavy tasks) to get an edge or degrade the service.
BLA3:2025 - Object state manipulations (OSM)	Sending sneaky values so the app quietly flips hidden settings (like roles or status) it shouldn’t let you change.	BLA8:2025 - Internal State Disclosure (ISD)	Error messages, codes, or timing differences leak what’s happening inside, helping attackers plan targeted abuse.
BLA4:2025 - Malicious Logic Loop (MLL)	Triggering a process that never properly stops (or repeats too much) to chew up time, money, or system resources.	BLA9:2025 - Broken Access Control (BAC)	The app doesn’t properly check permissions in key business actions, so people do things they’re not allowed to do.
BLA5:2025 - Artifact Lifetime Exploitation (ALE)	Using “short-lived” things (tokens, sessions, temporary files) after they should have expired because the app didn’t actually retire them.	BLA10:2025 - Shadow Function Abuse (SFA)	Abusing forgotten or hidden functions (test utilities, internal endpoints) left available in production.

rtifact Lifetime Exploitation (ALE)

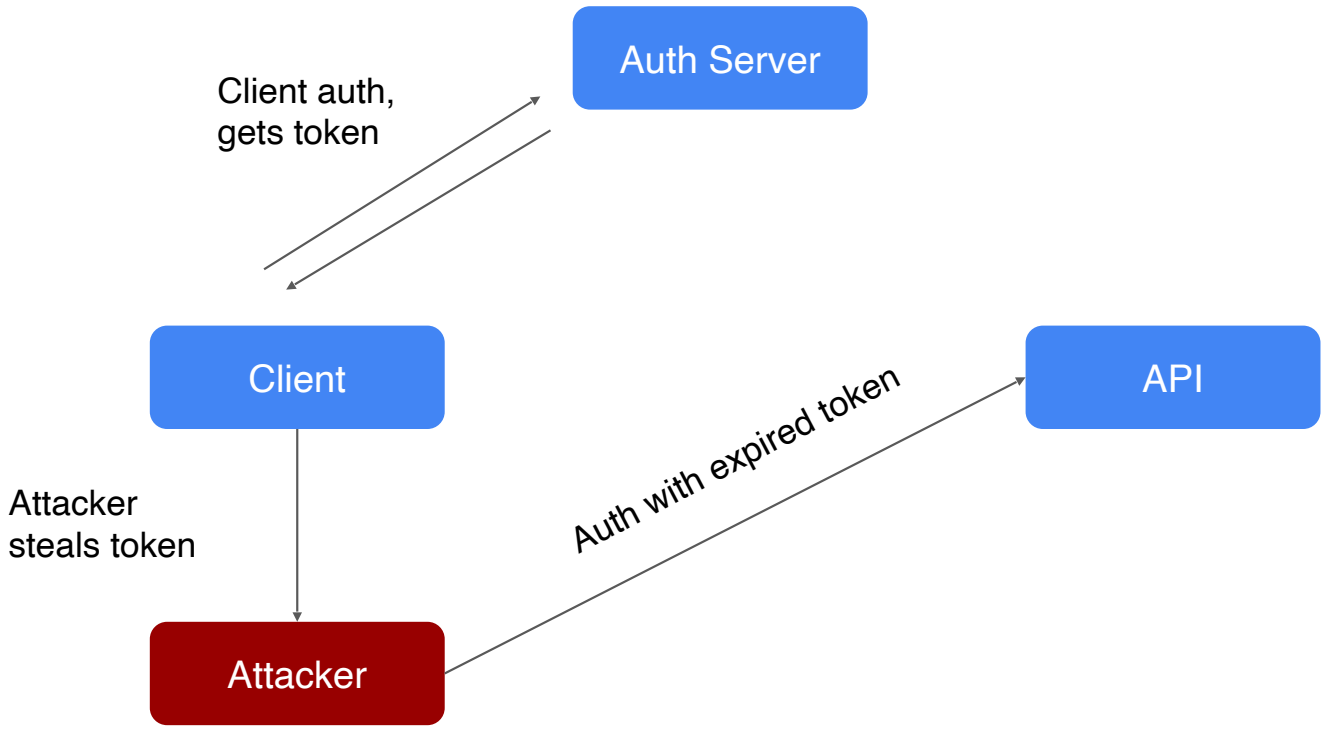



A FEW
MOMENTS LATER

rtifact Lifetime Exploitation (ALE)



rtifact Lifetime Exploitation (ALE)



wallarm | ENTERPRISE API SECURITY

stripe

ril 2025 – A sophisticated web skimming campaign has been targeting e-commerce platforms like WooCommerce, ordPress, and PrestaShop, injecting malicious scripts that replace legitimate checkout forms with fake replicas. tive since August 2024, the attackers exploit a deprecated Stripe API to validate stolen card data before exfiltrating improving efficiency and evasion. The campaign also impersonates Square payment forms and integrates yptocurrency options such as Bitcoin and Ethereum to further deceive users.

BLA10:2025 - Shadow Function Abuse (SFA)	Abusing forgotten or hidden functions (test utilities, internal endpoints) left available in production.

wallarm | ENTERPRISE API SECURITY

Salesloft.

gust 2025 – An attacker exploited stolen OAuth tokens stolen from the Salesloft Drift integration (an AI chat ent that connects with Salesforce and other enterprise tools) to systematically exfiltrate sensitive data from ndreds of Salesforce customer instances. The attackers retrieved high-value data such as AWS access keys, owflake tokens, passwords, Cases, Accounts, Contacts, and Opportunities. They also deleted job logs to ver their tracks.

BLA5:2025 - Artifact Lifetime Exploitation (ALE)

Using “short-lived” things (tokens, sessions, temporary files) after they should have expired because the app didn’t actually retire them.

wallarm | ENTERPRISE API SECURITY

ticketmaster®

going – The U.S. Federal Trade Commission sued ticket reseller Key Investment Group for evading rchasing limits to buy up thousands of tickets to live events including Taylor Swift's Eras tour and resell em at a markup, according to a complaint filed in Maryland federal court on Monday.

e Baltimore, Maryland-based company, which operates ticket resale sites including TotalTickets.com, ed thousands of Ticketmaster accounts, including fake or purchased accounts, the FTC said.

BLA7:2025 - Resource Quota Violation (RQV)

Hammering a feature too fast or too often (vote spamming, heavy tasks) to get an edge or degrade the service.

ake Action

Development

- Make BLA vulnerabilities a priority for developers
- Educate developers about BLA issues
- Architect applications to make these conditions impossible to exploit

ake Action

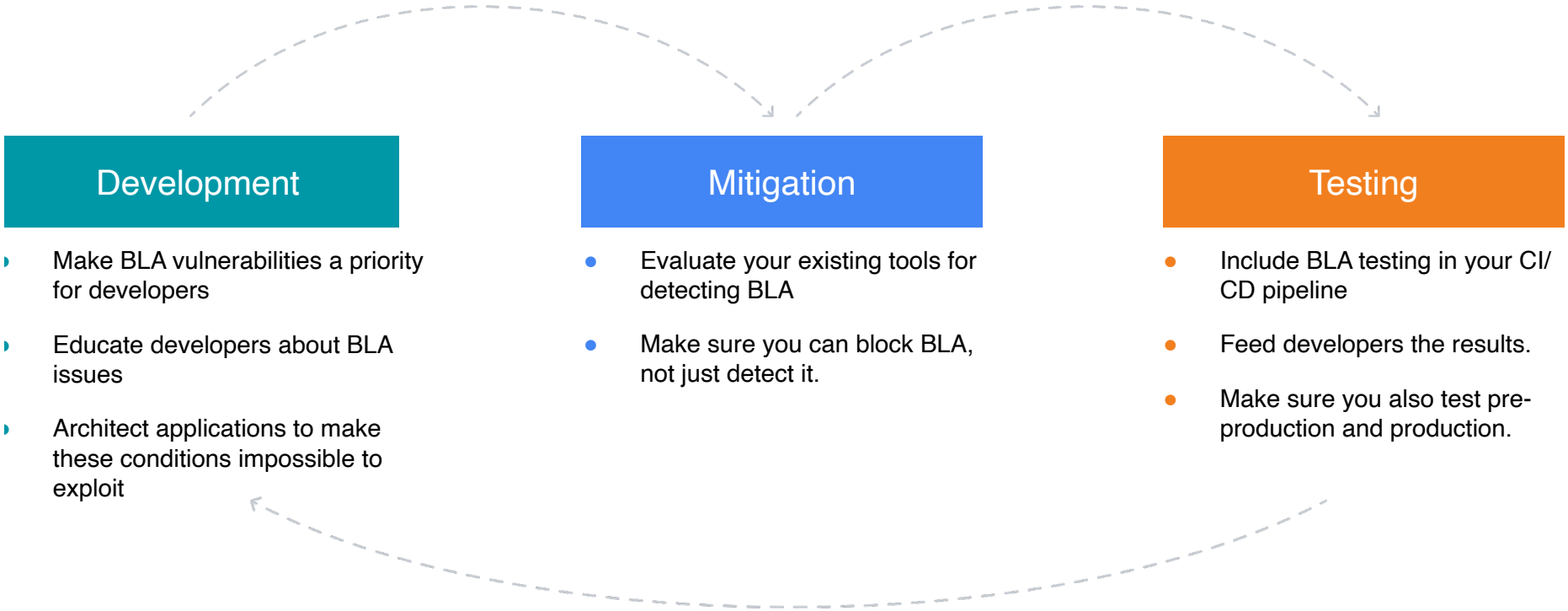
Development

- Make BLA vulnerabilities a priority for developers
- Educate developers about BLA issues
- Architect applications to make these conditions impossible to exploit

Mitigation

- Evaluate your existing tools for detecting BLA
- Make sure you can block BLA, not just detect it.

ake Action



Development

- Make BLA vulnerabilities a priority for developers
- Educate developers about BLA issues
- Architect applications to make these conditions impossible to exploit

Mitigation

- Evaluate your existing tools for detecting BLA
- Make sure you can block BLA, not just detect it.

Testing

- Include BLA testing in your CI/CD pipeline
- Feed developers the results.
- Make sure you also test pre-production and production.

wallarm | ENTERPRISE API SECURITY

API Certification!

https://www.wallarm.com/api-security-certification

wallarm | ENTERPRISE API SECURITY

Get the Whitepaper!

Wallarm Protection for the OWASP Top 10 Business Logic Abuse

Learn how Wallarm protects your organization from the OWASP Top 10 For Business Logic Abuse.

https://www.wallarm.com/resources/wallarm-protects-against-the-owasp-business-logic-abuse-top-10