Preview on OWASP Top 10 for API Security 2019

AIRLOCK®
SECURE ACCESS HUB

Airlock and the OWASP Top 10 API Security 2019 (Draft as of July 2019) The Ten Most Critical API Security Risks

The following table lists the ten most critical API security risks, as identified by OWASP in their draft of "OWASP Top 10 2019" as of July 2019. It explains how Airlock API addresses each of these risks to protect APIs from these types of attacks and which features are relevant.

Vulnerability	Description by OWASP	How Airlock API Gateway prevents an exploit	Relevant Airlock Features
A1 – Missing Object Level Access Control	APIs tend to expose endpoints that handle object identifiers, creating a wide attack surface Level Access Control issue. Object level authorization checks should be taken in mind in every function that accesses a data source using an input from the user.	Airlock provides comprehensive access management features for APIs. Therefore, API access in general is controlled by solid authentication and authorization policies. Citing from the OWASP document, the main problem occurs "because the server component usually does not fully track the client's state, and instead relies more on parameters like object IDs, that are sent from the client to decide which objects to access." Airlock uses its patented DyVE (Dynamic Value Endorsement) feature to do exactly what's missing: tracking the state of object IDs across client requests. Using DyVE, client requests are restricted to only use valid object IDs that have been offered by the API beforehand in the same client interaction. With this approach, attackers will not be able to inject stolen or crafted object IDs and get unauthorized access.	– API access management – User/client authentication and authorization – Dynamic Value Endorsement (DyVE)

About OWASP
The Open Web Application Security Project (OWASP) is an open community dedicated to enabling organisations to conceive, develop, acquire, operate and maintain applications that can be trusted. All of the OWASP tools, documents, forums and chapters are free and open to anyone interested in improving application security. For more information visit the homepage at www.owasp.org

About OWASP Top 10
OWASP just started a project for a new top 10 list for API Security raising awareness for this new topic, like it already does for application security. The 10 issues listed represent a broad consensus on what the most critical API security topics are at this time. For more information on the OWASP Top 10, visit https://www.owasp.org/index.php/OWASP_API_Security_Project

Preview on OWASP Top 10 for API Security 2019

AIRLOCK®
SECURE ACCESS HUB

Vulnerability	Description by OWASP	How Airlock API Gateway prevents an exploit	Relevant Airlock Features
A2 – Broken Authentication	Authentication mechanisms are often implemented incorrectly, allowing attackers to compromise authentication tokens or to exploit implementation flaws to assume other user's identities temporarily or permanently. Compromising system's ability to identify the client/user, compromises API overall security.	Airlock provides central access management for APIs, supporting all relevant standards such as OAuth 2 and OpenID Connect. Authentication of users and API clients is not restricted to API keys. A variety of integrated authentication means enable easy implementation of strong authentication policies also on APIs. API keys can be restricted for accessing only the authorized endpoints and subpaths with the correct HTTP verbs. Moreover, Airlock is compliant with PSD2 standards for API access management, such as NextGenPSD2 or STET. Roadmap 2020: Airlock API Gateway comes with fully integrated API Key management and usage plan enforcement.	– Policy Enforcement for Authentication and Authorization – Risk-based/adaptive authentication – Integration of many authentication factors – Secure Session Management – OAuth 2 support – OpenID Connect support – API Key Enforcement – API Key Management
A3 – Excessive Data Exposure	Looking forward to generic implementations developers tend to expose all object properties without considering their individual sensitivity, relying on clients to perform the data filtering before showing it to the user. Without controlling client's state, servers receive more and more filters which can be abused to gain access to sensitive data.	Acting as a reverse proxy, Airlock is able to rewrite responses from APIs. In particular, sensitive parts that should not be exposed to the client can be stripped away. By enforcing API specifications (e.g., OpenAPI), attackers can't send illegal filtering options on endpoints to trick APIs into delivering sensitive information. Error pages tend to leak sensitive information. Using Airlock, these can be replaced with safe standard pages.	– Reverse Proxy architecture – Content rewriting – Error page replacement – OpenAPI specification enforcement

Vulnerability	Description by OWASP	How Airlock API Gateway prevents an exploit	Relevant Airlock Features
A4 – Lack of Resources & Rate Limiting	Quite often APIs do not impose any restrictions on the size or number of resources that can be requested by the client/user. Not only this can impact the API server performance, leading to Denial of Service (DoS), but also leaves the door open to authentication flaws such as brute force.	Airlock has rate limiting features and enforces various limits on payload and JSON objects, such as the maximum nesting level, maximum name lengths, or number of elements and payload size. For attacks against authentication credentials, specific detection and prevention features are in place. Roadmap 2020: Airlock API Gateway comes with flexible usage plans on an API key level.	– Rate limiting – Limits enforcement – DOS prevention – Authentication credential brute-forcing detection
A5 – Missing Function/Resources Level Access Control	Complex access control policies with different hierarchies, groups and roles and a not so clear separation between administrative and regular functions tend to lead to authorization flaws. Exploiting these issues, attackers gain access to other users resources and/or administrative functions.	Airlock brings comprehensive access management and control features. By default, access to new APIs is turned off and must be enabled. Authorization of authenticated requests can be controlled on a fine-grained level including API endpoints, subpaths (e.g. /admin), HTTP verbs (GET vs. DELETE). In addition, access history and risk scoring inform access control decisions and allows implementation of risk-based authentication schemes.	– API access management and control – HTTP verbs and subpath filtering and authorization – Risk-based authentication – History of successful authentication contexts for cross-checking

Vulnerabilities	Description by OWASP	How Airlock API Gateway prevents an exploit	Relevant Airlock Features
A6 – Mass Assignment	Binding client provided data (e.g. JSON) to data models without proper properties filtering based on a whitelist usually lead to Mass Assignment. Either guessing objects properties, exploring other API endpoints or reading the documentation, providing additional object properties in request payloads, allow attackers to modify object properties they are not supposed to.	Airlock validates API calls against formal specifications such as OpenAPI. Access to unpublished endpoints, "creative" object syntax and tampered element formats are stopped at the gateway already. On top of formal specifications, policy learning allows to enhance object property rules, in case the specification is not tight enough. For example, just requiring an element to be a "string" opens the door for injection attacks. Using policy learning with actual traffic, Airlock may suggest enforcing an element type UUID, which is not susceptible to injection attacks. Furthermore, all elements are checked against blacklist rules and web attack signatures. Using DyVE (Dynamic Value Endorsement), object IDs and other element values may be restricted to be valid only within a single API interaction (e.g., identified by the same access token). For example, after a user has been authenticated, the "user_name" attribute of subsequent calls should be considered to be read-only by the client and must not be changed.	– OpenAPI specification support – Policy learning – Web security signatures and blacklists – DyVE (Dynamic Value Endorsement)
A7 – Security Misconfiguration	Security misconfiguration is commonly a result of insecure default configurations, incomplete or ad hoc configurations, open cloud storage, misconfigured HTTP headers, unnecessary HTTP methods, permissive Cross-Origin resource sharing (CORS) and verbose error messages containing sensitive information.	Airlock contains default rules which are regularly updated. The mapping-oriented configuration guides the administrator to configure access only to known endpoints, preventing forceful browsing. Error messages are rewritten or replaced to eliminate exposure of stack traces and the like. Typical errors such as excessively permissive CORS headers or missing "secure" attributes in cookies are recognized and corrected. Configuration validators are checking the Airlock configuration and warn about common misconfigurations (Log only mode, certificate mismatches, etc.). The smart policy learning automatically generates meaningful and well-balanced configuration suggestions for easily handling detected issues. This helps guiding administrators along best practices and prevents overreaction in stressful situations. If formal specifications of APIs are available, e.g. an OpenAPI file, Airlock can automatically enforce API calls to be compliant with the specification and prevent misconfiguration. Moreover, Airlock provides an up-to-date TLS configuration which is centrally enforced. HTTP security headers are set by default and can be customized.	– Secure default configuration – built-in filters – TLS termination and secure TLS configuration – Always up-to-date system with maintenance releases and hot-fixes – Policy learning supports integrators – Default CORS rules – Content rewriting – Error page replacement – Header rewriting – Configuration validation – Open API specification support

Preview on OWASP Top 10 for API Security 2019

AIRLOCK®
SECURE ACCESS HUB

Vulnerabilities	Description by OWASP	How Airlock API Gateway prevents an exploit	Relevant Airlock Features
A8 – Injection	Injection flaws, such as SQL, NoSQL, Command Injection, occur when untrusted data is sent to an interpreter as part of a command or query. The attacker's hostile data can trick the interpreter into executing unintended commands or accessing data without proper authorization.	Requests containing injections like SQL, XSS, HTML, LDAP, OS commands and others are detected and blocked by combining blacklisting and dynamic whitelisting. Real-time threat intelligence feeds are used to block known malicious clients. If formal specifications of APIs are available, e.g. an OpenAPI file, Airlock can automatically enforce API calls to be compliant with the specification and prevent injections for attributes with a tight format description. Dynamic Value Endorsement tracks object IDs in API interactions and prevents injections of crafted IDs. Attacks in headers or cookies are prevented by filtering and a cookie store. Airlock protects itself against overflow and OS injection attacks by using strict security domain separation, SELinux to implement least privilege, ASLR, No-execute and strong stack protection. The ICAP interface allows checking content either with Airlock add-on modules such as SOAP/ XML-filters or third-party malware scanners (AV). Other types of injections and protocol violations are prevented by the protocol termination and regeneration.	– Built-in blacklist filters – OpenAPI specification support – Dynamic Value Endorsement (DyVE) – Whitelist parameter Learning – Cookie Store – HTTP Protocol termination and regeneration – Security domain separation – Principle of least privilege – Address-Layout-Randomization (ASLR) – No-Execute (NX) – Stack-protection (SSP) and Stack Clash Protection
A9 – Improper Assets Management	APIs tend to expose more endpoints than traditional web applications, what makes proper and updated documentation highly important. Proper hosts and deployed API versions inventory also play an important role to mitigate issues such as deprecated API versions and exposed debug endpoints.	Airlock reads formal specifications such as OpenAPI files. Access is then restricted to endpoints and versions that have actually been published, i.e., are listed in the specification. In addition, proper access management is enforced on published endpoints. All other endpoints remain internal. Internal hosts and APIs are not accessible to the public by default due to the reverse-proxy architecture of Airlock. OpenAPI files registered on Airlock may also be served publicly as an always up-to-date API documentation for API client developers.	– OpenAPI specification support – Selectively publishing endpoints – Access management to pulished endpoints – Reverse-proxy architecture – Serving of API documentation

AIRLOCK®
SECURE ACCESS HUB

Vulnerabilities	Description by OWASP	How Airlock API Gateway prevents an exploit	Relevant Airlock Features
A10 – Insufficient Logging & Monitoring	Insufficient logging and monitoring, coupled with missing or ineffective integration with incident response, allows attackers to further attack systems, maintain persistence, pivot to more systems and tamper, extract, or destroy data. Most breach studies show time to detect a breach is over 200 days, typically detected by external parties rather than internal processes or monitoring.	Airlock is a highly specialized security component dealing with detection and prevention of various types of API attacks. Any resource request and incident is persisted into a database. Detected attacks are visible in the onboard reporting with many default dashboards and a dynamic query language for zooming in on relevant details. When attacks occur, events are generated and responsible contacts may be notified instantaneously. Attack information can be forwarded to a SIEM system using formats like CEF or JSON for correlation and incident response. Virtual patches against attacks are quickly applicable, centrally and for all protected applications at once.	– Detection of ongoing attacks – Logging of attacks – Blocking of attacks – Event generation for notification and timely incident response – Real-time graphical reporting dashboards – Linked log entries with reports for quick root cause analysis – SIEM Integration (CEF log format, Airlock App for Splunk)

Ergon Informatik AG
Merkurstrasse 43
CH-8032 Zurich

Phone +41 44 268 89 00
www.ergon.ch

ergon