Are you sure your APIs are secure?

Bas Dijkstra
bas@ontestautomation.com
www.ontestautomation.com

Why API security testing?

Explosive Growth

83% of all internet traffic is from APIs

Akamai

Major Attack Target

2022: APIs "most frequent attack vector"

Gartner

High Profile Breaches

High-profile API breaches announced weekly

PELOTON coinbase

Regulatory Compliance

Regulations mandate privacy, vulnerability detection, testing

PCI GDPR.EU

But isn't API security testing something best left to the experts?

You can do a lot with some basic tools and a healthy dose of curiosity and creativity

Let's have a look at some examples

Vulnerability:

injection

eBay

In late 2015 and early 2016, eBay had a severe XSS vulnerability. The website used a “url” parameter that redirected users to different pages on the platform, but the value of the parameter was not validated. This allowed attackers to inject malicious code into a page.

The vulnerability enabled attackers to gain full access to eBay seller accounts, sell products at a discount, and steal payment details. It was actively used by attackers to manipulate eBay listings of high value products such as vehicles. eBay eventually remediated the vulnerability, but follow-on attacks continued until 2017.

Where to perform input validation / sanitizing?

Frontend, backend or both?

Why?

The OWASP API security top 10

https://owasp.org/API-Security/editions/2023/en/0x00-header/

Example

Vulnerability:

Broken Object Level Authorization (BOLA)

2023 OWASP API security top 10: #1

Predictable or findable resource IDs

Insufficient or lack of rate limiting

Look out for …

https://owasp.org/API-Security/editions/2023/en/0xa1-broken-object-level-authorization/

Also covered: vulnerability:

Unrestricted Resource Consumption

2023 OWASP API security top 10: #4

coinbase

KrebsonSecurity

In-depth security news and investigation

HOME ABOUT THE AUTHOR ADVERTISING/SPEAKING

USPS Site Exposed Data on 60 Million Users

November 21, 2018 54 Comments

Many of the API's features accepted "wildcard" search parameters, meaning *they could be made to return all records for a given data set without the need to search for specific terms.* No special hacking tools were needed to pull this data, other than knowledge of how to view and modify data elements processed by a regular Web browser like Chrome or Firefox.

How about
altering data?

Vulnerability:

Broken Function Level Authorization (BFLA)

2023 OWASP API security top 10: #5

Access to admin endpoints by regular users

Also look out for …

https://owasp.org/API-Security/editions/2023/en/0xa5-broken-function-level-authorization/

Where BOLA is about accessing data…

… BFLA is about the ability to alter or delete data

BFLA and BOLA

So, if you happen upon a BOLA vulnerability…

… it might be a good idea to check for BFLA, too

Another example

Vulnerability:

Unrestricted Access to Sensitive Business Flows

2023 OWASP API security top 10: #6

First identify sensitive business flows…

… then take prevention measures

Prevention

https://owasp.org/API-Security/editions/2023/en/0xa6-unrestricted-access-to-sensitive-business-flows/

Some of the protection mechanisms are more simple while others are more difficult to implement. The following methods are used to slow down automated threats:

- Device fingerprinting: denying service to unexpected client devices (e.g headless browsers) tends to make threat actors use more sophisticated solutions, thus more costly for them
- Human detection: using either captcha or more advanced biometric solutions (e.g. typing patterns)
- Non-human patterns: analyze the user flow to detect non-human patterns (e.g. the user accessed the "add to cart" and "complete purchase" functions in less than one second)
- Consider blocking IP addresses of Tor exit nodes and well-known proxies

Secure and limit access to APIs that are consumed directly by machines (such as developer and B2B APIs). They tend to be an easy target for attackers because they often don't implement all the required protection mechanisms.

Yet another example

Create Password

* PASSWORD

••••••••••

* CONFIRM PASSWORD

••••••••

Passwords should match.

- ✓ Password Strength: Strong
- ✓ At least 8 characters
- ✓ Contains a digit
- ✓ Contains a lowercase letter
- ✓ Contains an uppercase letter
- ✓ Contains a symbol

Vulnerability:

Broken Authentication

2023 OWASP API security top 10: #2

Vulnerability:

Improper Inventory Management

2023 OWASP API security top 10: #9

We covered many potential security issues today!

The 2023 OWASP API security top 10 entries we missed

#1 Broken Object Level Authorization
#2 Broken Authentication
#3 Broken Object Property Level Authorization
#4 Unrestricted Resource Consumption
#5 Broken Function Level Authorization
#6 Unrestricted Access to Sensitive Business Flows
#7 Server Side Request Forgery
#8 Security Misconfiguration
#9 Improper Inventory Management
#10 Unsafe Consumption of APIs

Server Side Request Forgery: an example

That's a lot you can do!

(and there's more…)

https://www.apisecuniversity.com

Contact

_Email: bas@ontestautomation.com

_Website: https://www.ontestautomation.com

_LinkedIn: https://www.linkedin.com/in/basdijkstra