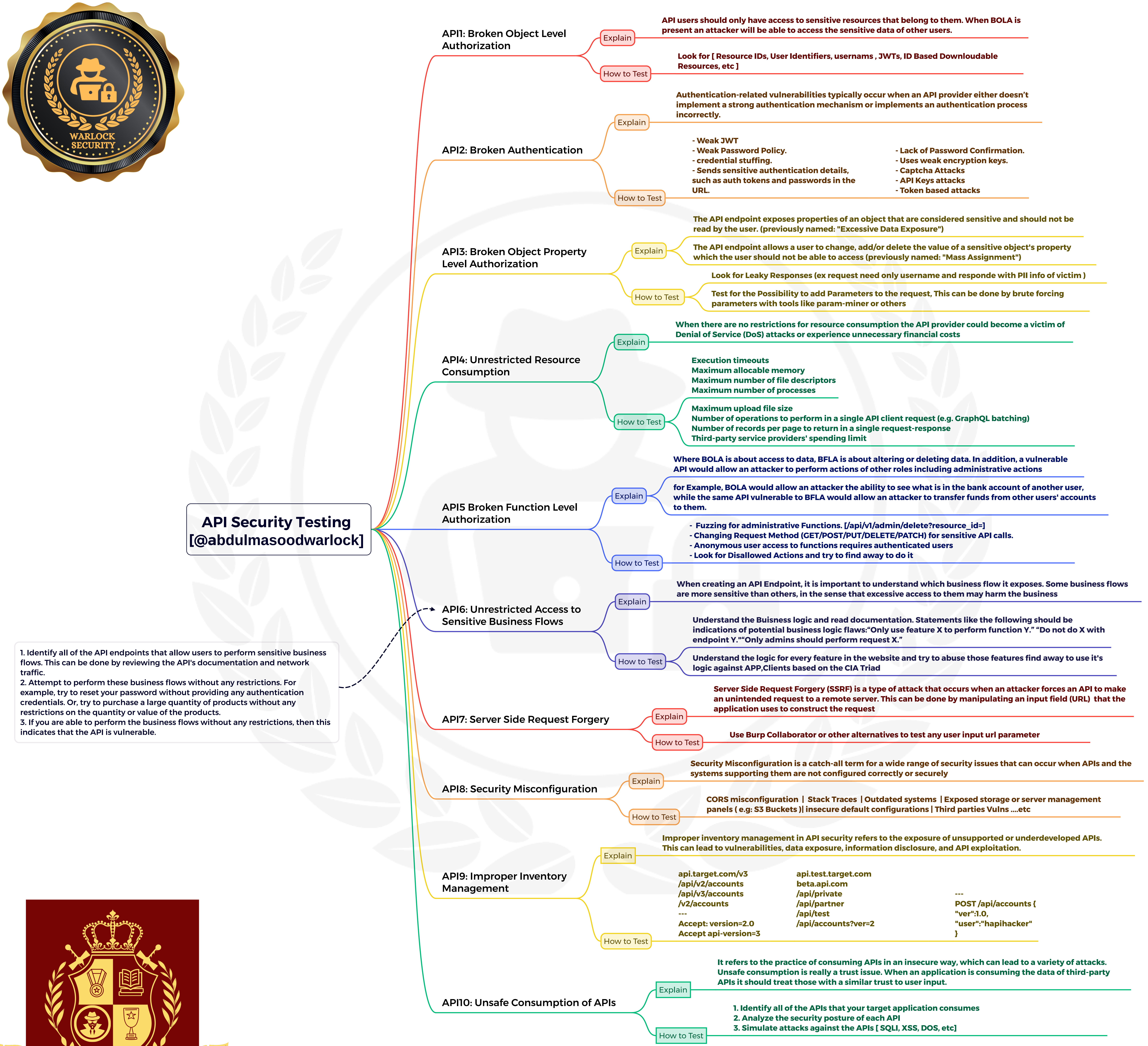
WARLOCK SECURITY



API Security Testing [@abdulmasoodwarlock]

API1: Broken Object Level Authorization

Explain: API users should only have access to sensitive resources that belong to them. When BOLA is present an attacker will be able to access the sensitive data of other users.

How to Test: Look for [Resource IDs, User Identifiers, usernams , JWTs, ID Based Downloudable Resources, etc]

API2: Broken Authentication

Explain: Authentication-related vulnerabilities typically occur when an API provider either doesn't implement a strong authentication mechanism or implements an authentication process incorrectly.

How to Test:

- Weak JWT
- Weak Password Policy.
- credential stuffing.
- Sends sensitive authentication details, such as auth tokens and passwords in the URL.
- Lack of Password Confirmation.
- Uses weak encryption keys.
- Captcha Attacks
- API Keys attacks
- Token based attacks

API3: Broken Object Property Level Authorization

Explain:

The API endpoint exposes properties of an object that are considered sensitive and should not be read by the user. (previously named: "Excessive Data Exposure")

The API endpoint allows a user to change, add/or delete the value of a sensitive object's property which the user should not be able to access (previously named: "Mass Assignment")

How to Test:

Look for Leaky Responses (ex request need only username and responde with PII info of victim)

Test for the Possibility to add Parameters to the request, This can be done by brute forcing parameters with tools like param-miner or others

API4: Unrestricted Resource Consumption

Explain: When there are no restrictions for resource consumption the API provider could become a victim of Denial of Service (DoS) attacks or experience unnecessary financial costs

How to Test:

Execution timeouts
Maximum allocable memory
Maximum number of file descriptors
Maximum number of processes

Maximum upload file size
Number of operations to perform in a single API client request (e.g. GraphQL batching)
Number of records per page to return in a single request-response
Third-party service providers' spending limit

API5 Broken Function Level Authorization

Explain:

Where BOLA is about access to data, BFLA is about altering or deleting data. In addition, a vulnerable API would allow an attacker to perform actions of other roles including administrative actions

for Example, BOLA would allow an attacker the ability to see what is in the bank account of another user, while the same API vulnerable to BFLA would allow an attacker to transfer funds from other users' accounts to them.

How to Test:

- Fuzzing for administrative Functions. [/api/v1/admin/delete?resource_id=]
- Changing Request Method (GET/POST/PUT/DELETE/PATCH) for sensitive API calls.
- Anonymous user access to functions requires authenticated users
- Look for Disallowed Actions and try to find away to do it

API6: Unrestricted Access to Sensitive Business Flows

Explain: When creating an API Endpoint, it is important to understand which business flow it exposes. Some business flows are more sensitive than others, in the sense that excessive access to them may harm the business

How to Test:

Understand the Buisness logic and read documentation. Statements like the following should be indications of potential business logic flaws:"Only use feature X to perform function Y." "Do not do X with endpoint Y.""Only admins should perform request X."

Understand the logic for every feature in the website and try to abuse those features find away to use it's logic against APP,Clients based on the CIA Triad

1. Identify all of the API endpoints that allow users to perform sensitive business flows. This can be done by reviewing the API's documentation and network traffic.
2. Attempt to perform these business flows without any restrictions. For example, try to reset your password without providing any authentication credentials. Or, try to purchase a large quantity of products without any restrictions on the quantity or value of the products.
3. If you are able to perform the business flows without any restrictions, then this indicates that the API is vulnerable.

API7: Server Side Request Forgery

Explain: Server Side Request Forgery (SSRF) is a type of attack that occurs when an attacker forces an API to make an unintended request to a remote server. This can be done by manipulating an input field (URL) that the application uses to construct the request

How to Test: Use Burp Collaborator or other alternatives to test any user input url parameter

API8: Security Misconfiguration

Explain: Security Misconfiguration is a catch-all term for a wide range of security issues that can occur when APIs and the systems supporting them are not configured correctly or securely

How to Test: CORS misconfiguration | Stack Traces | Outdated systems | Exposed storage or server management panels (e.g: S3 Buckets)| insecure default configurations | Third parties Vulns ....etc

API9: Improper Inventory Management

Explain: Improper inventory management in API security refers to the exposure of unsupported or underdeveloped APIs. This can lead to vulnerabilities, data exposure, information disclosure, and API exploitation.

How to Test:

```
api.target.com/v3        api.test.target.com
/api/v2/accounts         beta.api.com
/api/v3/accounts         /api/private              ---
/v2/accounts             /api/partner              POST /api/accounts {
---                      /api/test                 "ver":1.0,
Accept: version=2.0      /api/accounts?ver=2       "user":"hapihacker"
Accept api-version=3                               }
```

API10: Unsafe Consumption of APIs

Explain: It refers to the practice of consuming APIs in an insecure way, which can lead to a variety of attacks. Unsafe consumption is really a trust issue. When an application is consuming the data of third-party APIs it should treat those with a similar trust to user input.

How to Test:

1. Identify all of the APIs that your target application consumes
2. Analyze the security posture of each API
3. Simulate attacks against the APIs [SQLI, XSS, DOS, etc]

W S T C
WARLOCK SECURITY TRAINING CENTER

Presented by Abdul Masood Founder of Warlock Security