





Academy home

> All labs

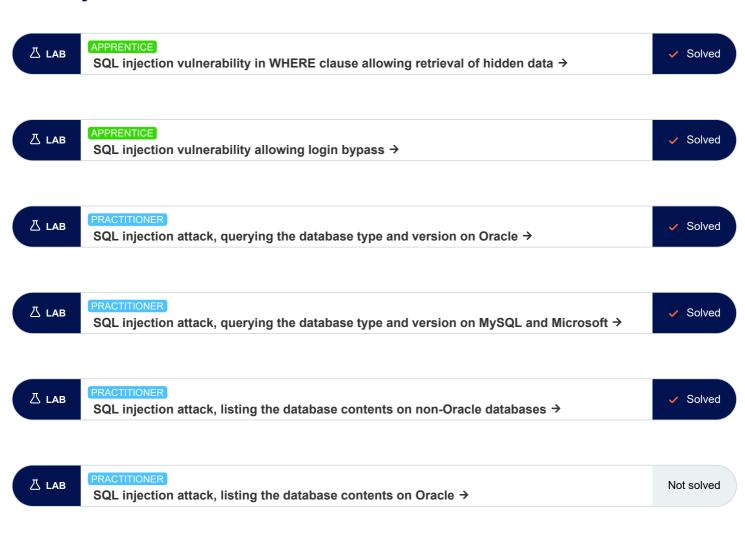
### **All labs**

#### Mystery lab challenge

Try solving a random lab with the title and description hidden. As you'll have no prior knowledge of the type of vulnerability that you need to find and exploit, this is great for practicing recon and analysis.

Take me to the mystery lab challenge →

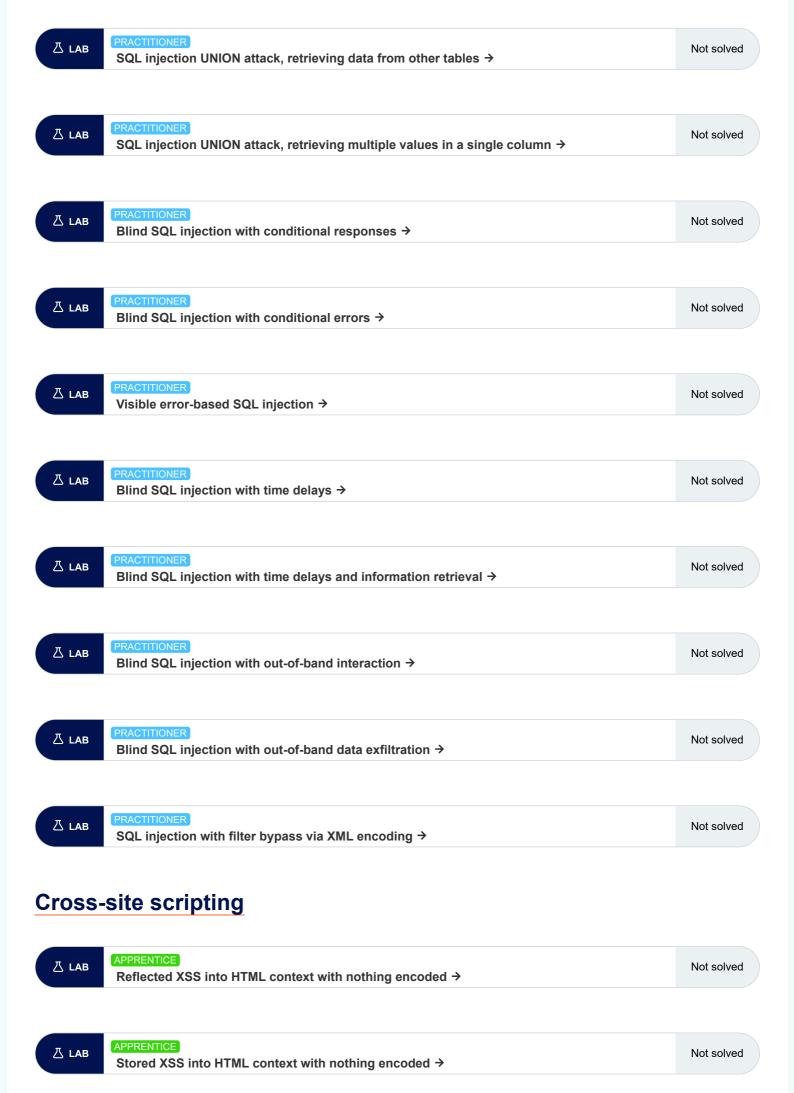
### SQL injection

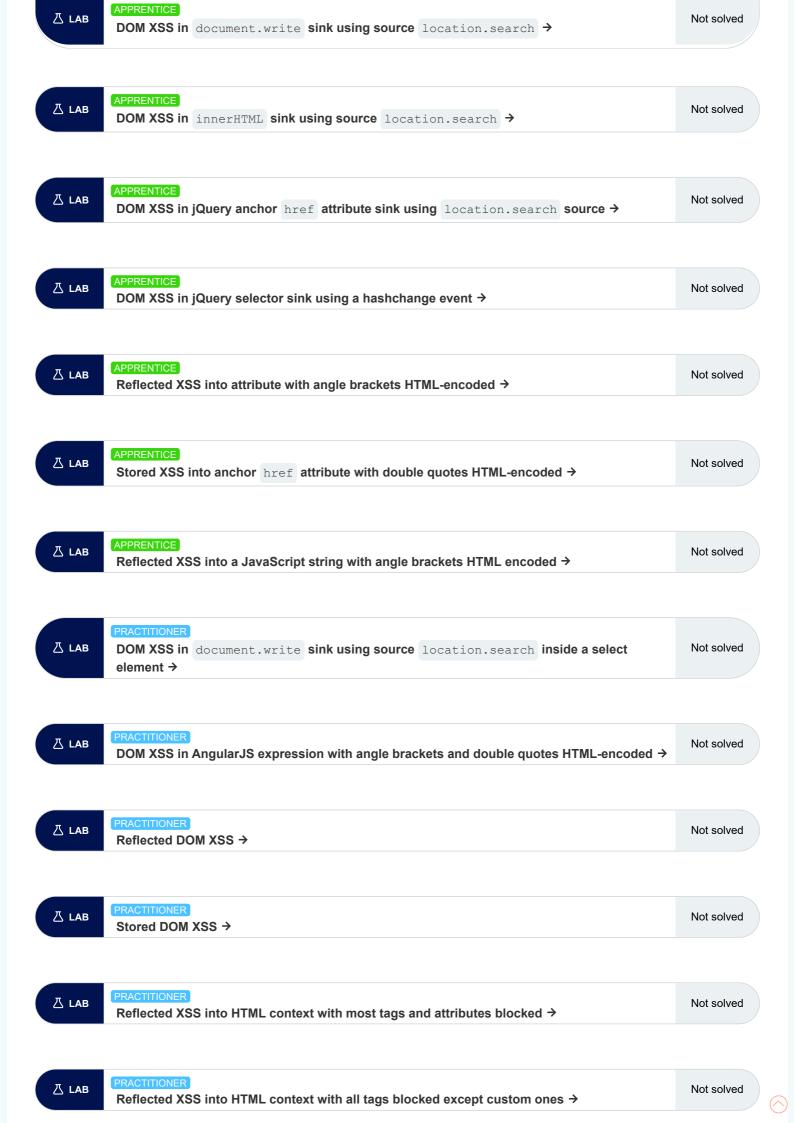


SQL injection UNION attack, determining the number of columns returned by the query →

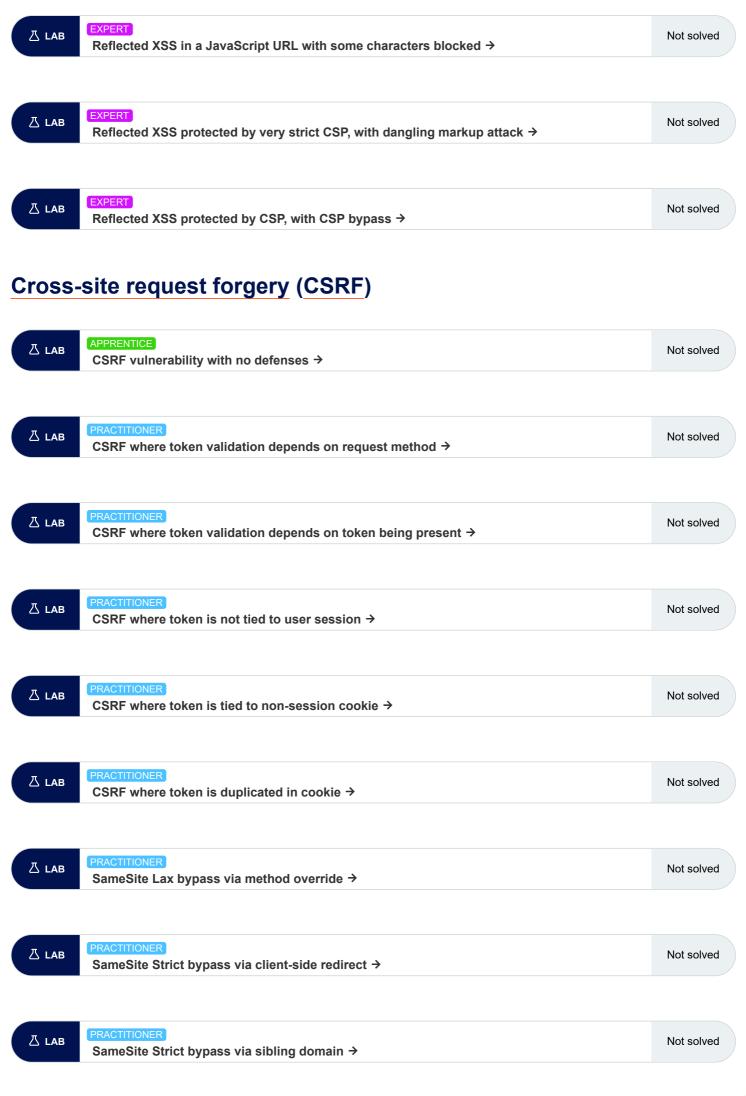
**∐ LAB** 

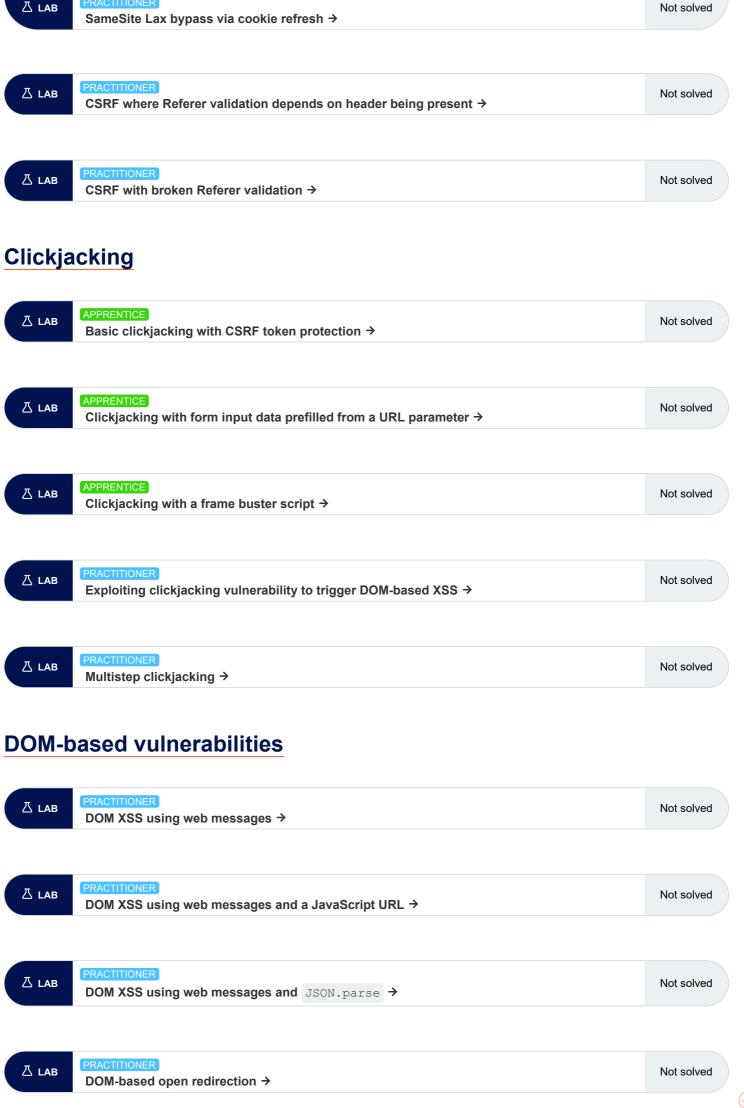
Not solved

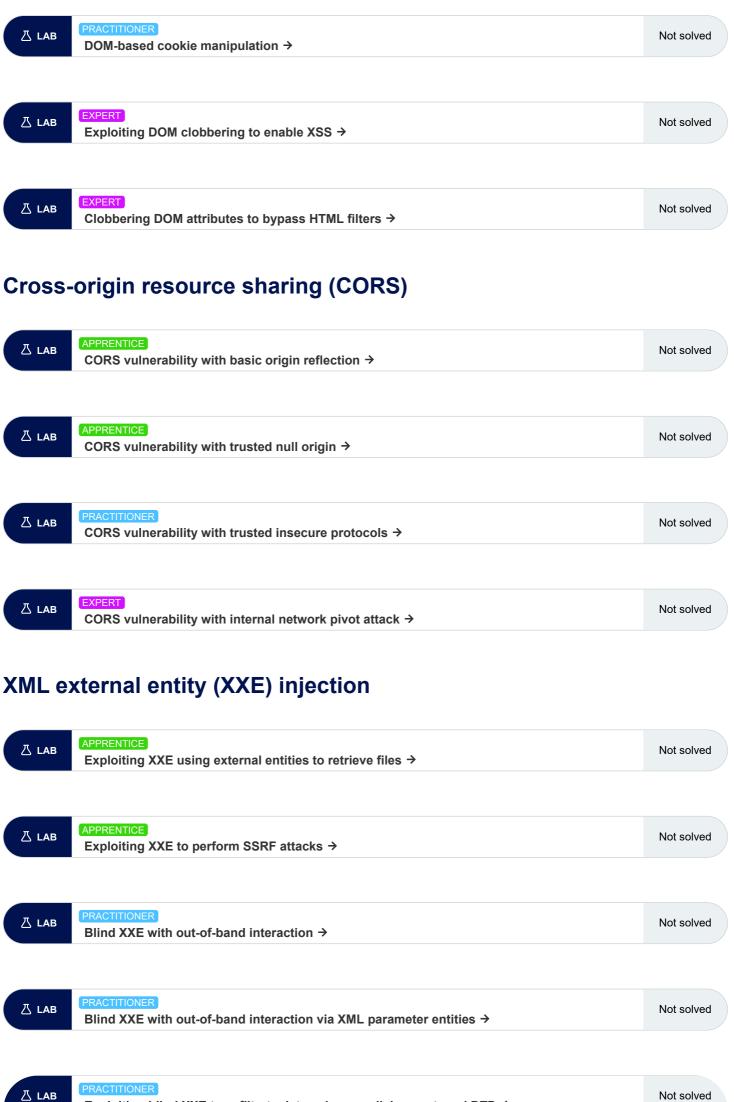


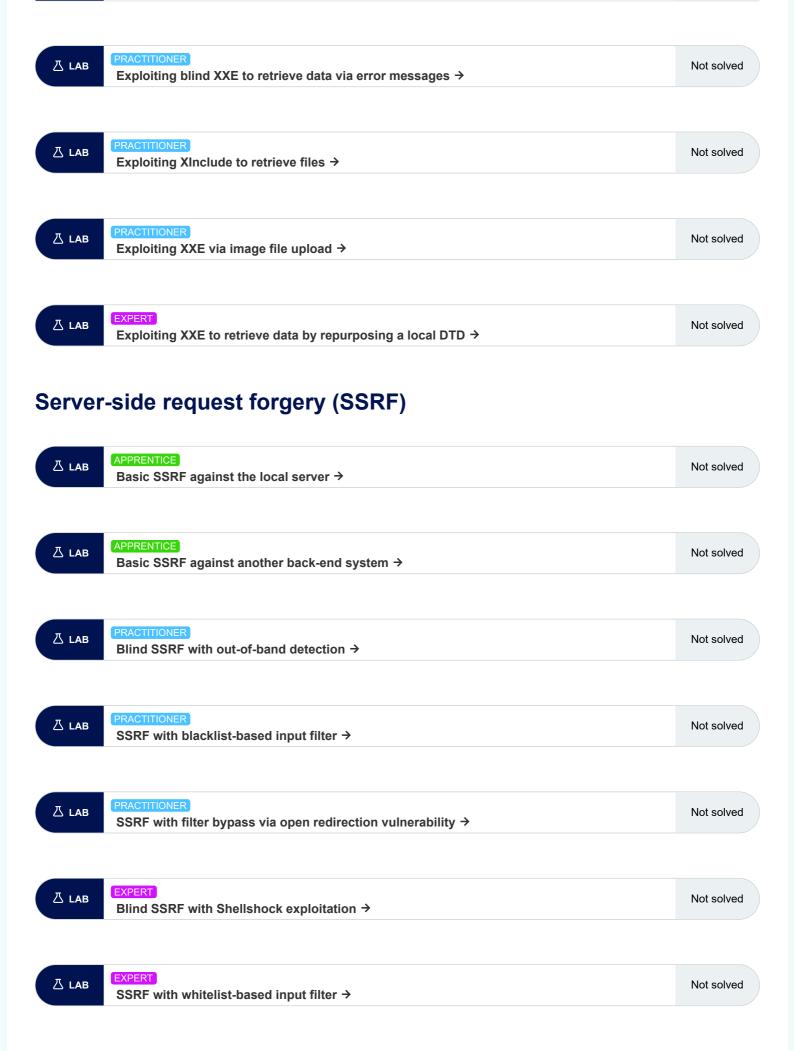


∐ LAB	PRACTITIONER  Reflected XSS with some SVG markup allowed →	Not solved
∐ LAB	PRACTITIONER	Not solved
	Reflected XSS in canonical link tag →	
	PRACTITIONER	
∐ LAB	Reflected XSS into a JavaScript string with single quote and backslash escaped →	Not solved
∐ LAB	PRACTITIONER	Not solved
△ LAB	Reflected XSS into a JavaScript string with angle brackets and double quotes HTML- encoded and single quotes escaped →	Not solved
	PRACTITIONER	
∐ LAB	Stored XSS into onclick event with angle brackets and double quotes HTML-encoded and single quotes and backslash escaped →	Not solved
	single quotes and backsiash escaped /	
	PRACTITIONER	
∐ LAB	Reflected XSS into a template literal with angle brackets, single, double quotes, backslash	Not solved
	and backticks Unicode-escaped →	
Пис	PRACTITIONER	Net estad
∐ LAB	Exploiting cross-site scripting to steal cookies →	Not solved
∐ LAB	PRACTITIONER  Exploiting cross-site scripting to capture passwords →	Not solved
	Exploiting cross-site scripting to capture passwords 7	
∐ LAB	PRACTITIONER	Not solved
△ LAB	Exploiting XSS to perform CSRF →	Not solved
∐ LAB	EXPERT  Reflected XSS with AngularJS sandbox escape without strings →	Not solved
∐ LAB	EXPERT	Not solved
(A. E.D.)	Reflected XSS with AngularJS sandbox escape and CSP →	1.01.001000
	(AVAPOV)	
∐ LAB	EXPERT  Reflected XSS with event handlers and href attributes blocked →	Not solved
	The state of the s	



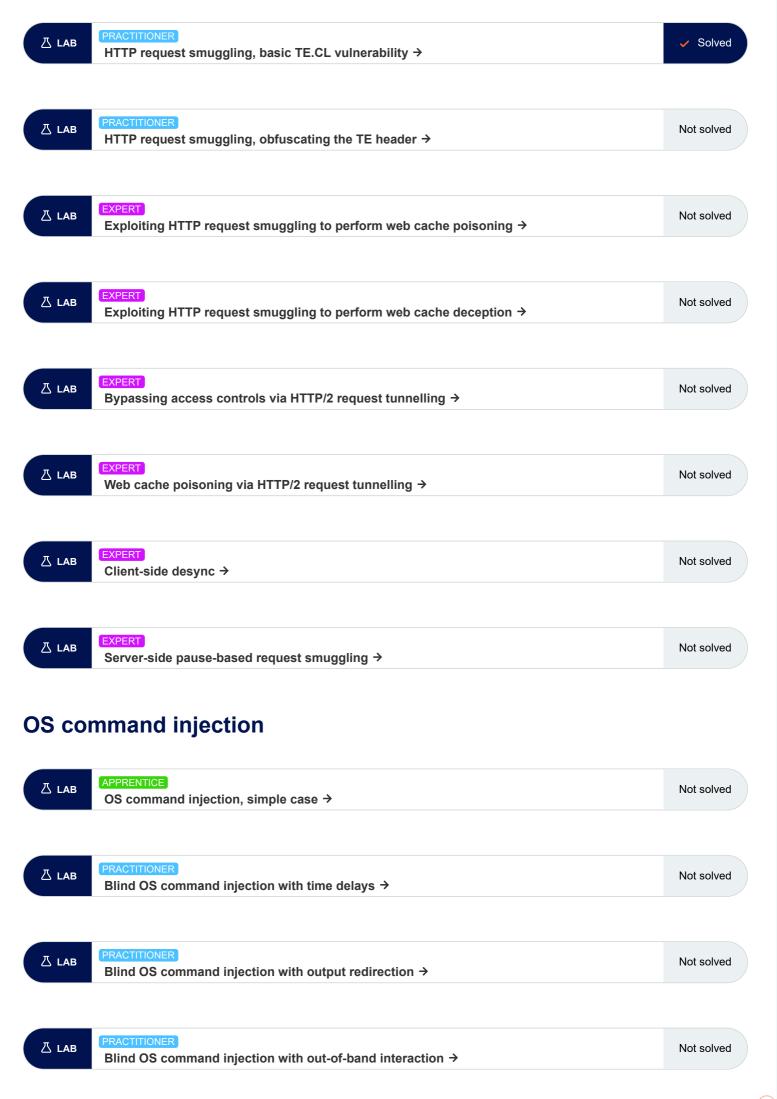






### **HTTP request smuggling**

∆ LAB	PRACTITIONER  HTTP request smuggling, confirming a CL.TE vulnerability via differential responses →	Not solved
∐ LAB	PRACTITIONER  HTTP request smuggling, confirming a TE.CL vulnerability via differential responses →	Not solved
∐ LAB	PRACTITIONER  Exploiting HTTP request smuggling to bypass front-end security controls, CL.TE vulnerability →	Not solved
∆ LAB	PRACTITIONER  Exploiting HTTP request smuggling to bypass front-end security controls, TE.CL vulnerability →	Not solved
∐ LAB	PRACTITIONER  Exploiting HTTP request smuggling to reveal front-end request rewriting →	Not solved
∆ LAB	PRACTITIONER  Exploiting HTTP request smuggling to capture other users' requests →	Not solved
∐ LAB	PRACTITIONER  Exploiting HTTP request smuggling to deliver reflected XSS →	Not solved
∐ LAB	PRACTITIONER  Response queue poisoning via H2.TE request smuggling →	Not solved
∐ LAB	PRACTITIONER  H2.CL request smuggling →	Not solved
∐ LAB	PRACTITIONER  HTTP/2 request smuggling via CRLF injection →	Not solved
∐ LAB	PRACTITIONER  HTTP/2 request splitting via CRLF injection →	Not solved
∐ LAB	PRACTITIONER  CL.0 request smuggling →	Not solved
∐ LAB	PRACTITIONER  HTTP request smuggling, basic CL.TE vulnerability →	Not solved



## Server-side template injection

 A LAB
 PRACTITIONER

 Basic server-side template injection (code context) →

Not solved

 A LAB
 PRACTITIONER

 Server-side template injection in an unknown language with a documented exploit →

Not solved

LAB PRACTITIONER
 Server-side template injection with information disclosure via user-supplied objects →
 Not solved.

 Image: LAB Server-side template injection in a sandboxed environment →
 Not solved

EXPERT

Server-side template injection with a custom exploit →

Not solved

### Path traversal

△ LAB

△ LAB

 Z LAB
 PRACTITIONER

 File path traversal, traversal sequences blocked with absolute path bypass →

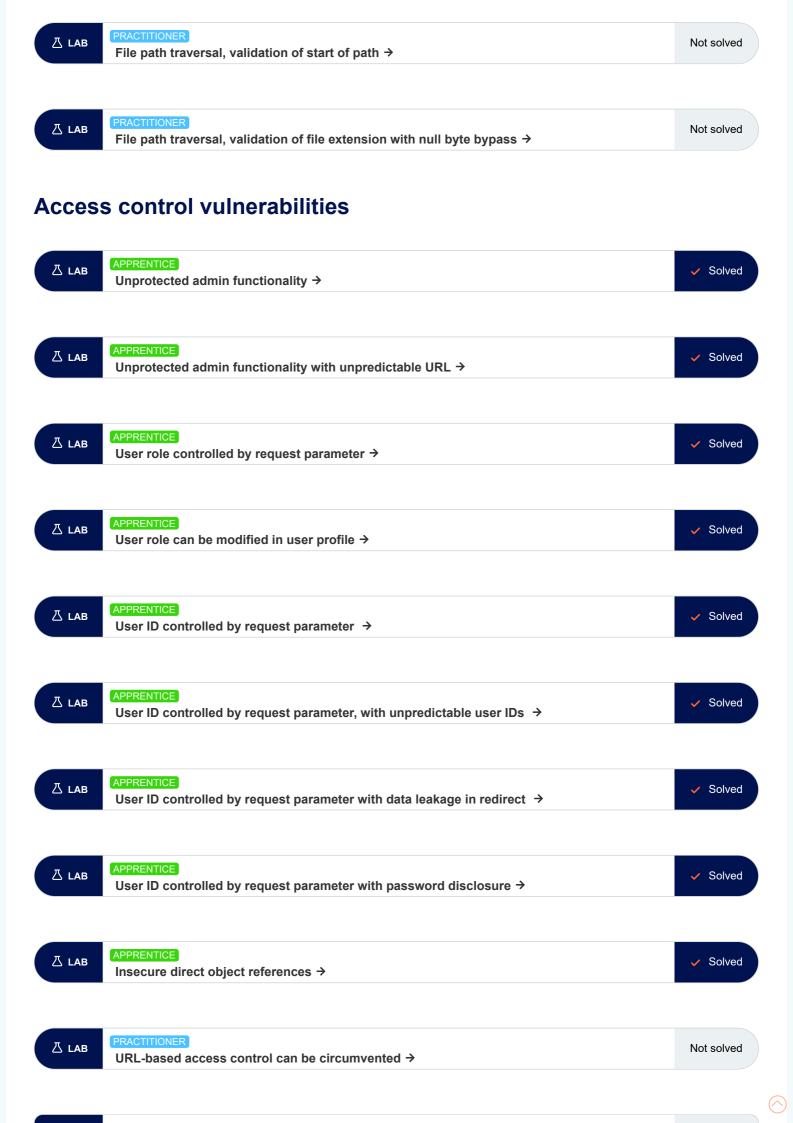
Not solved

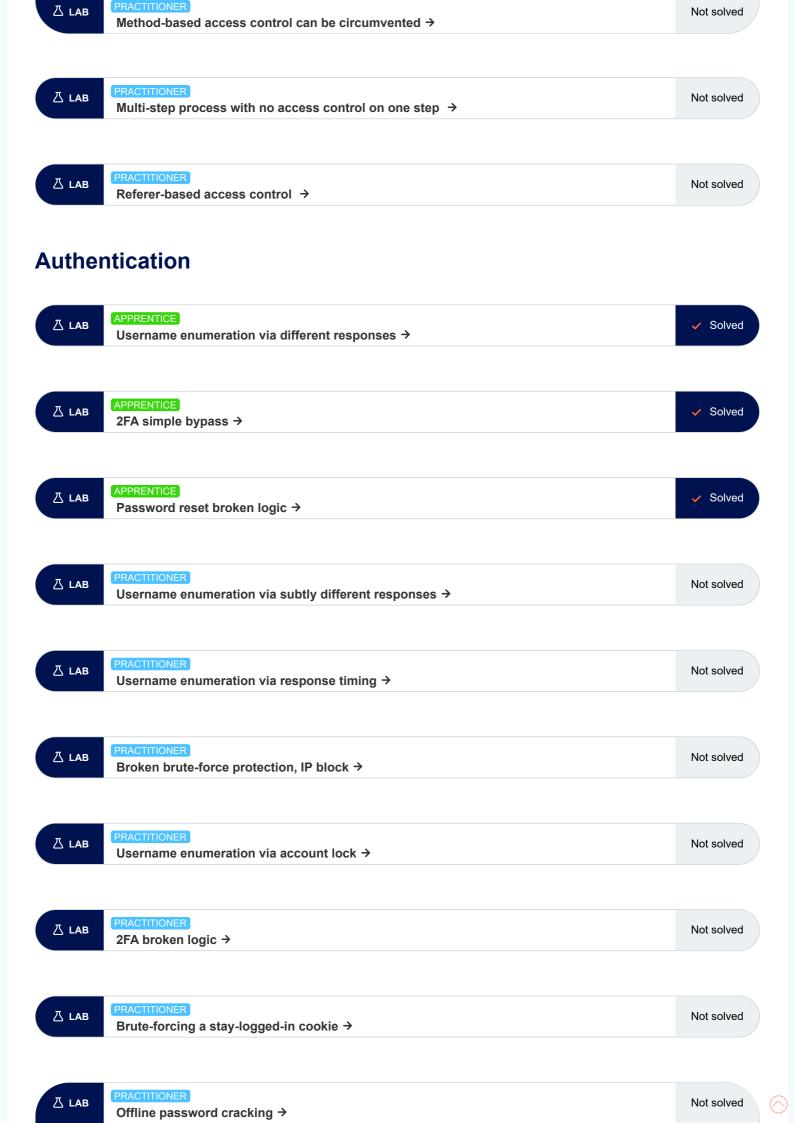
LAB PRACTITIONER
 File path traversal, traversal sequences stripped non-recursively →
 Not solved.

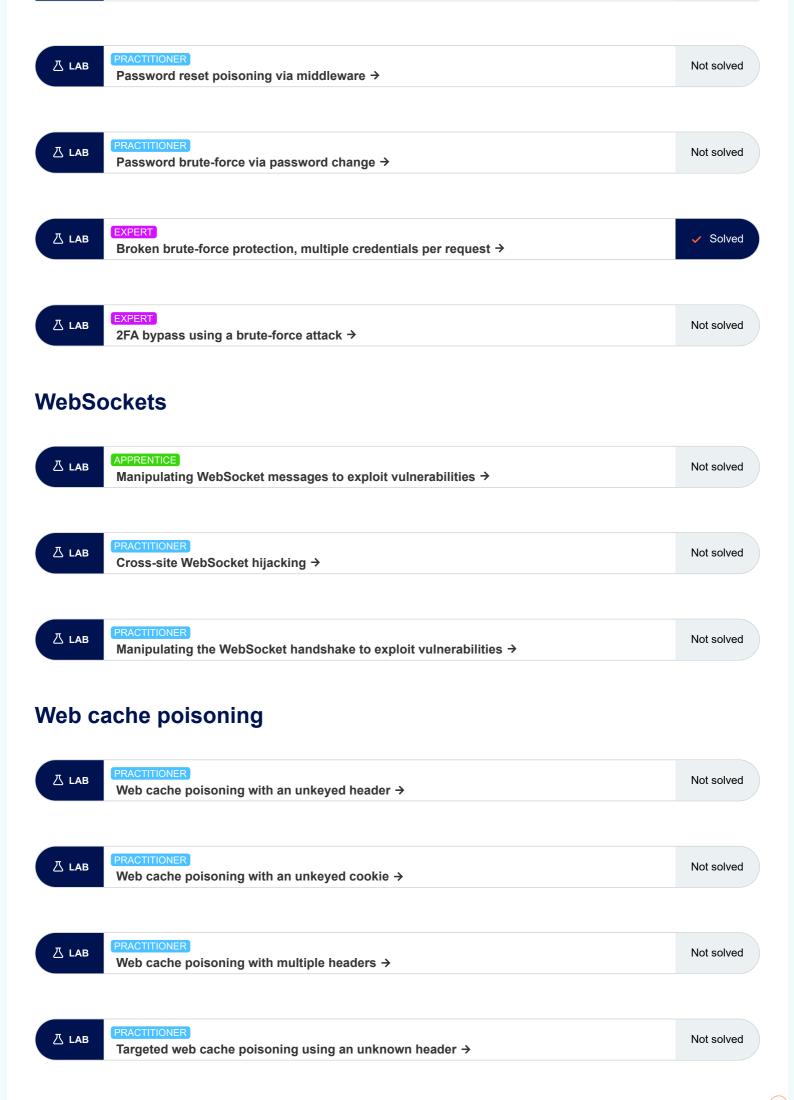
PRACTITIONER

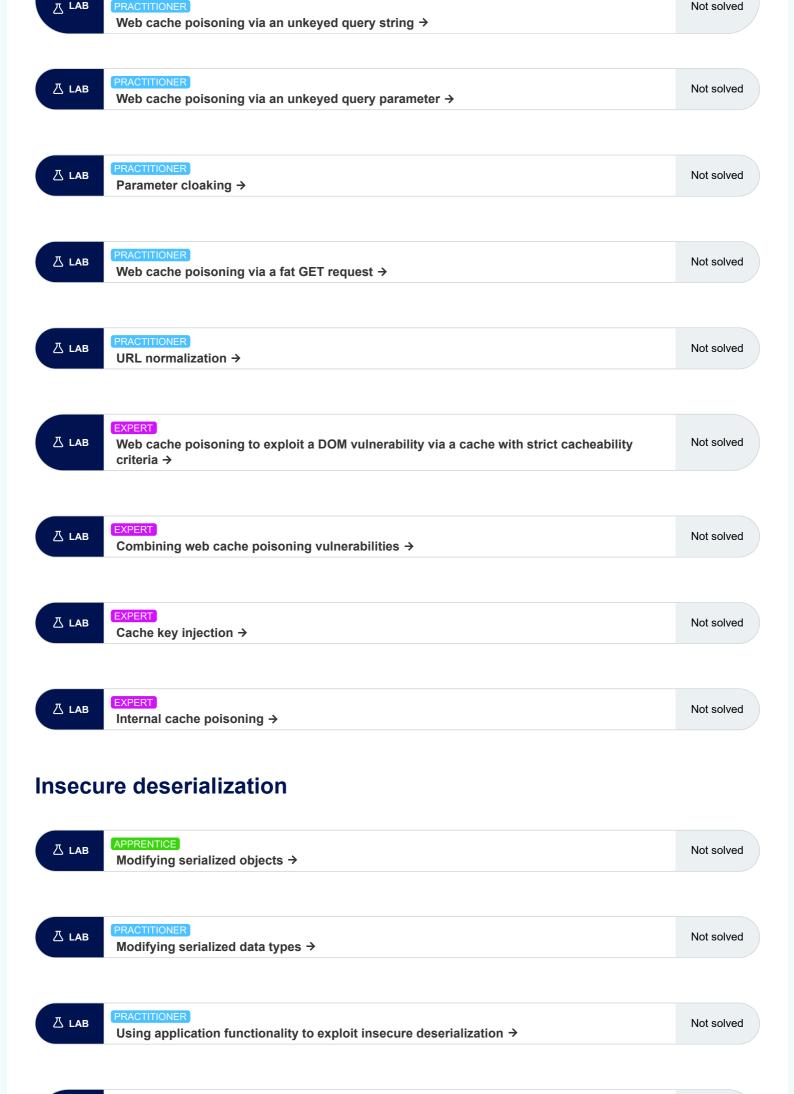
File path traversal, traversal sequences stripped with superfluous URL-decode →

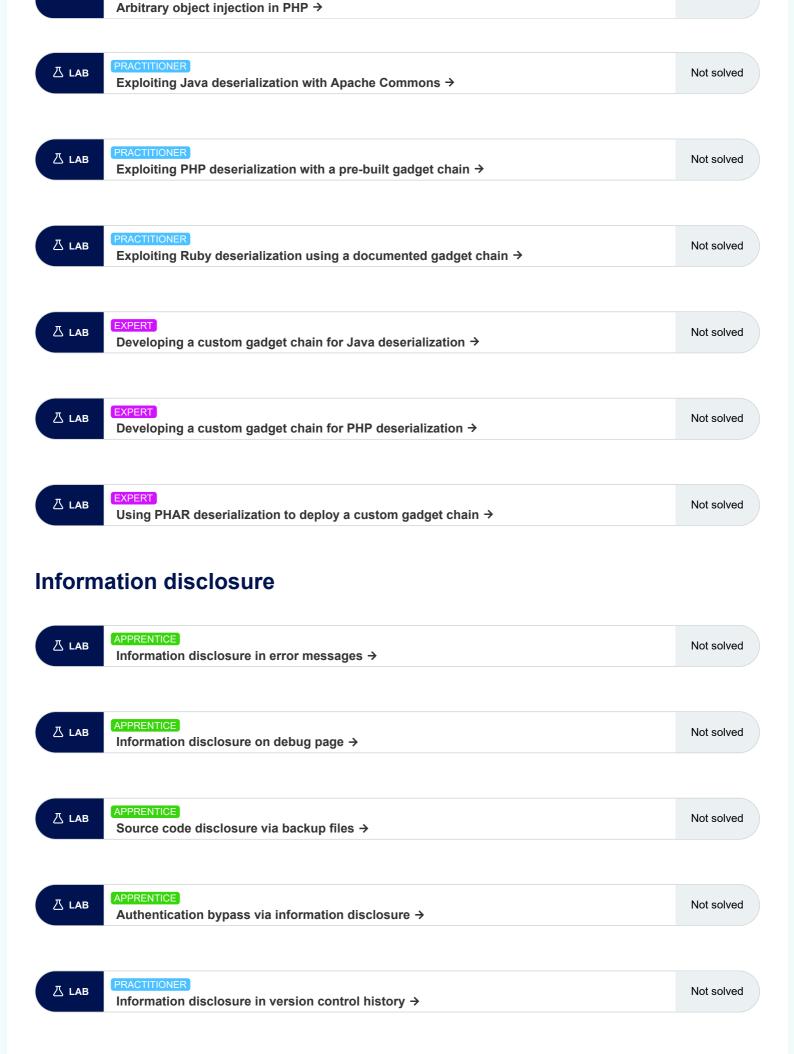
Not solved



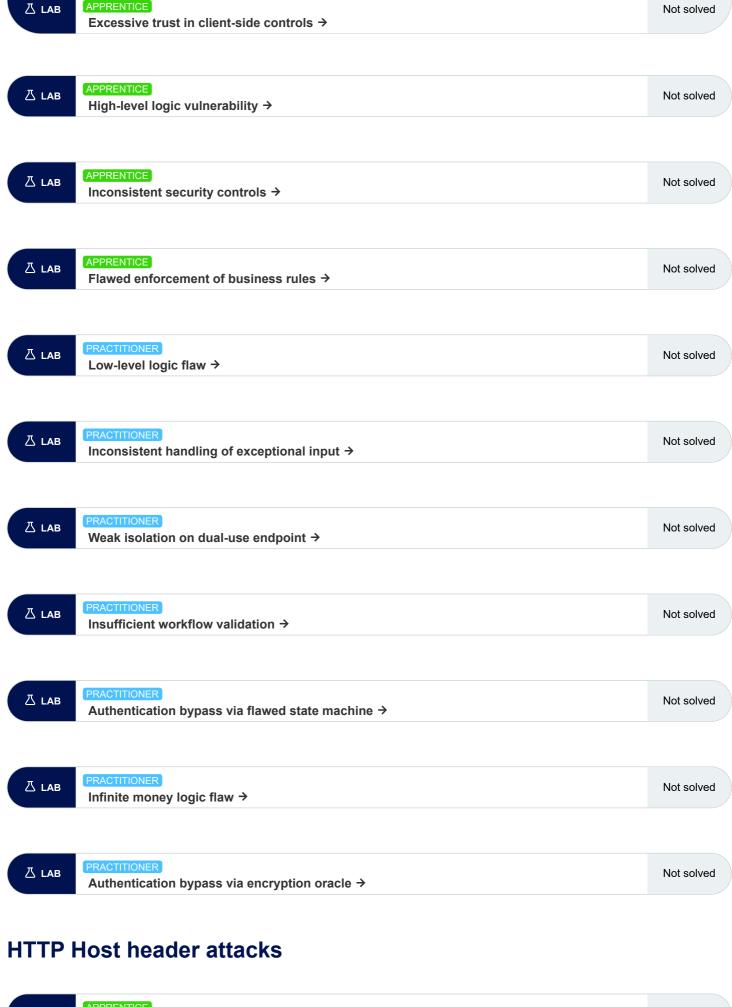




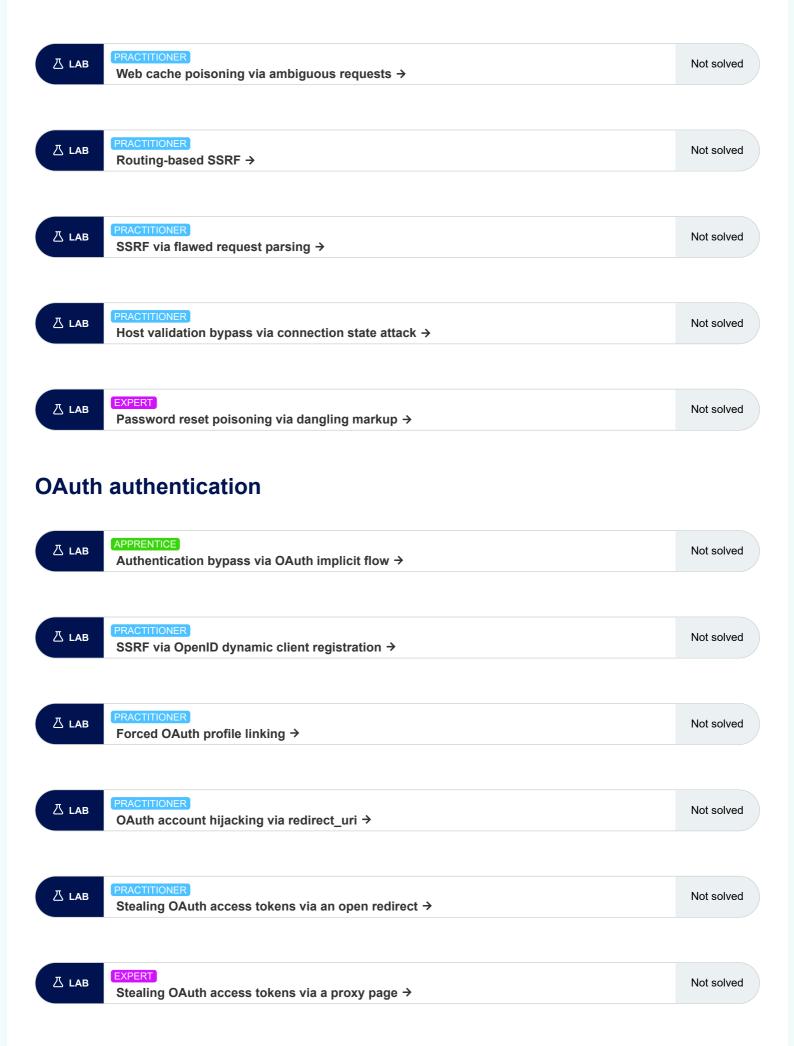




# **Business logic vulnerabilities**

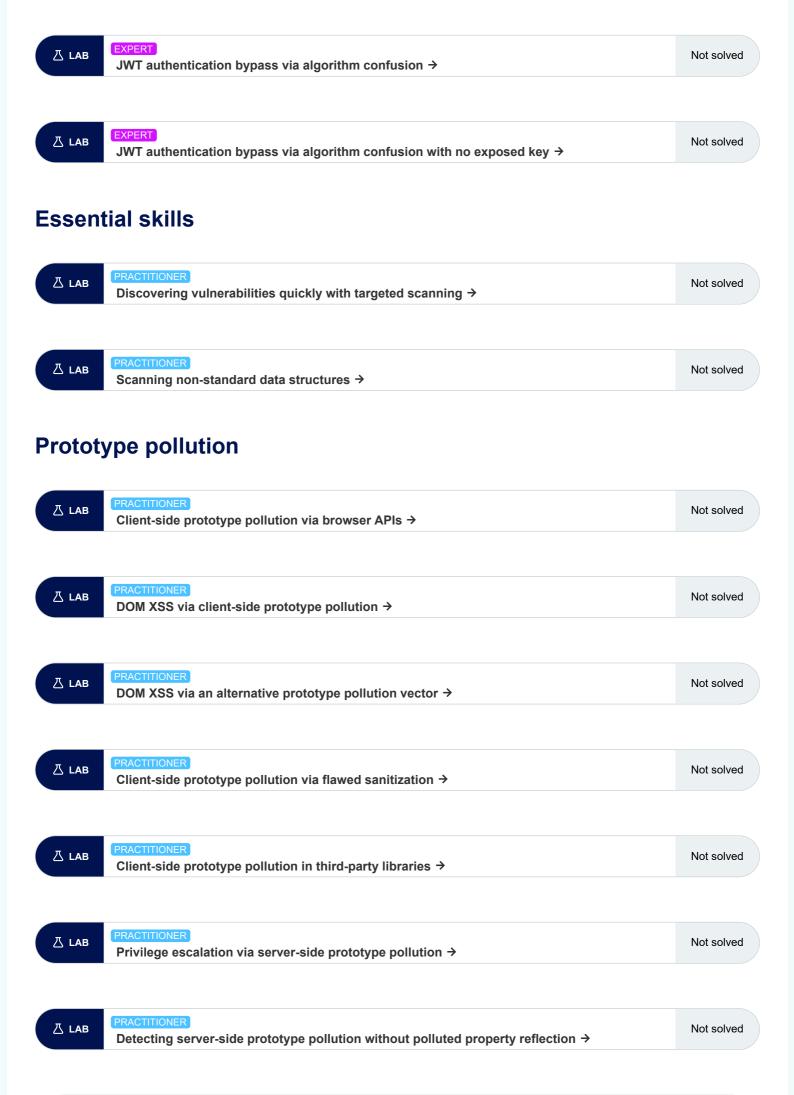


∐ LAB	APPRENTICE  Basic password reset poisoning →	Not solved
∐ LAB	APPRENTICE  Host header authentication bypass →	Not solved



## File upload vulnerabilities

∑ LAB	APPRENTICE	Not solved
	Remote code execution via web shell upload →	
∐ LAB	APPRENTICE  Web shell upload via Content-Type restriction bypass →	Not solved
∆ LAB	PRACTITIONER  Web shell upload via path traversal →	Not solved
∆ LAB	PRACTITIONER  Web shell upload via extension blacklist bypass →	Not solved
∆ LAB	PRACTITIONER  Web shell upload via obfuscated file extension →	Not solved
∆ LAB	PRACTITIONER  Remote code execution via polyglot web shell upload →	Not solved
∆ LAB	EXPERT  Web shell upload via race condition →	Not solved
JWT		
∆ LAB	APPRENTICE  JWT authentication bypass via unverified signature →	Not solved
∆ LAB	APPRENTICE  JWT authentication bypass via flawed signature verification →	Not solved
∆ LAB	PRACTITIONER  JWT authentication bypass via weak signing key →	Not solved
∐ LAB	PRACTITIONER  JWT authentication bypass via jwk header injection →	Not solved
∐ LAB	PRACTITIONER  JWT authentication bypass via jku header injection →	Not solved
∐ LAB	PRACTITIONER  JWT authentication bypass via kid header path traversal →	Not solved

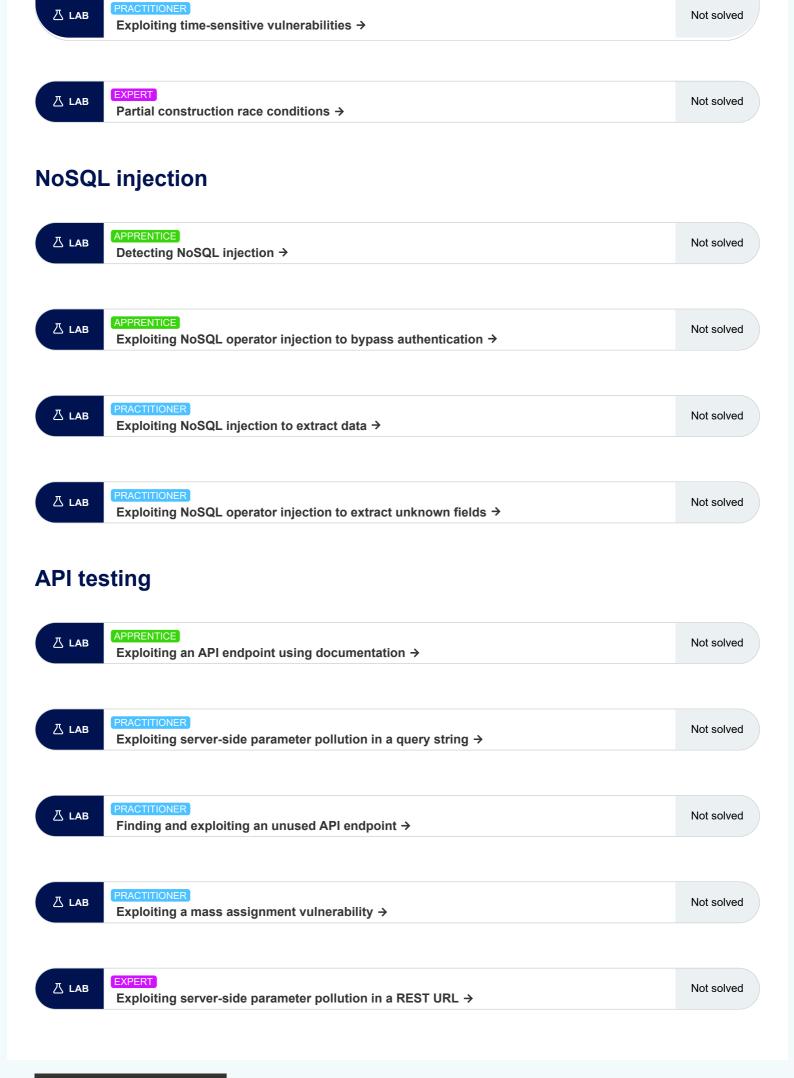


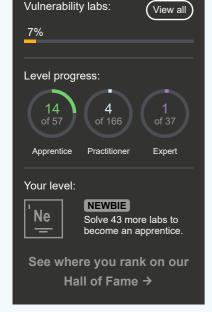
LAB

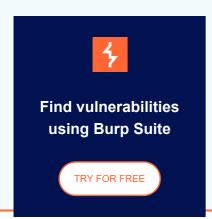
∐ LAB	PRACTITIONER  Remote code execution via server-side prototype pollution →	Not solved	
∐ LAB	EXPERT  Exfiltrating sensitive data via server-side prototype pollution →	Not solved	
GraphQL API vulnerabilities			
∆ LAB	APPRENTICE  Accessing private GraphQL posts →	Not solved	
∐ LAB	PRACTITIONER  Accidental exposure of private GraphQL fields →	Not solved	
∐ LAB	PRACTITIONER  Finding a hidden GraphQL endpoint →	Not solved	
∆ LAB	PRACTITIONER  Bypassing GraphQL brute force protections →	Not solved	
	PRACTITIONER		
∆ LAB	Performing CSRF exploits over GraphQL →	Not solved	
Race conditions			
∆ LAB	APPRENTICE  Limit overrun race conditions →	Not solved	
∐ LAB	PRACTITIONER  Bypassing rate limits via race conditions →	Not solved	
∐ LAB	PRACTITIONER  Multi-endpoint race conditions →	Not solved	
∐ LAB	PRACTITIONER  Single-endpoint race conditions →	Not solved	

Bypassing flawed input filters for server-side prototype pollution →

囚







# Burp Suite

Web vulnerability scanner Burp Suite Editions Release Notes

#### Vulnerabilities

Cross-site scripting (XSS) SQL injection Cross-site request forgery XML external entity injection Directory traversal Server-side request forgery

#### Customers

Organizations Developers

#### Company

About Careers Contact Legal Privacy Notice

### Insights

Web Security Academy Blog Research





© 2024 PortSwigger Ltd.