Intigriti Bug Bytes #230 - November 2025

BY AYOUB · NOVEMBER 21, 2025

Hi hackers,

Welcome to the latest edition of Bug Bytes! In this month's issue, we'll be featuring:

- Finding an RCE using AI in GitHub
- CORS exploitation cheat sheet
- Scanning codebases with AI
- Bypassing paywalls
- SSTIs in AI models

And so much more! Let's dive in!

Company News

Intigriti wins 2025 UK IT Industry Awards

We are thrilled to announce that Intigriti has won *Security Innovation of the Year* at the UK IT Industry Awards 2025!

This award recognises Intigriti's breakthroughs and excellence in delivering cybersecurity services worldwide. We are immensely proud to mark this milestone and would like to thank our researcher community for making this possible.

Read all about this achievement in our **announcement post** or on our LinkedIn page.

ntigriti wins ‘Security Innovation of the Year’ at the 2025 UK IT Industry Awards

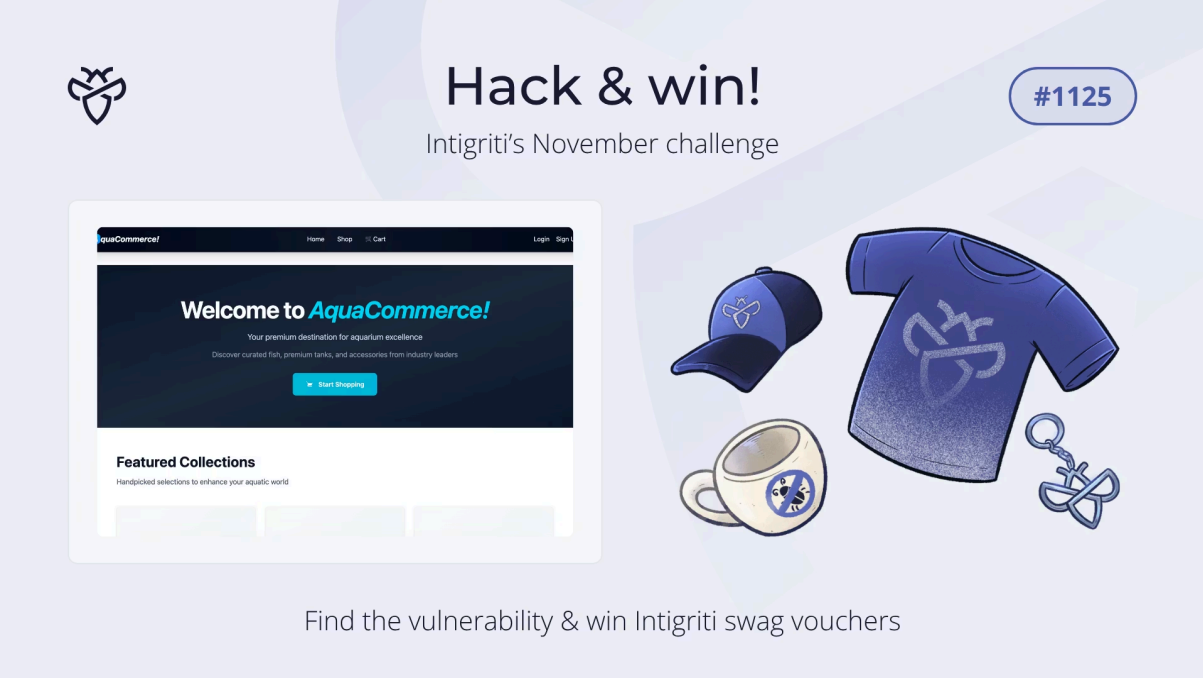
Learn more

Participate in INTIGRITI CTF 1125!

Looking for a fun activity to put your hacking skills to the test? This month’s CTF challenge requires you to pop a shell and find your way into the system to capture the flag.

With over 75+ solves so far, it joins the list of one of the most popular challenges to be featured on Intigriti.

Wish to still participate? Follow the instructions on the challenge page and make sure you submit your flag before Monday, the 24th, 11:59 PM UTC.



INTIGRITI 1125 CTF Challenge

[Participate now](#)

Blogs & videos

[Exploiting JWTs](#)

Exploiting JWT vulnerabilities: A complete guide Cover Image

JSON Web Tokens power authentication in millions of modern web applications... yet misconfigurations and improper validation create critical security flaws that can lead to complete account takeover. From

algorithm confusion attacks to key injection vulnerabilities, developers often introduce exploitable weaknesses.

In our technical article, we've outlined 7 methods to test and exploit JWT vulnerabilities, including real-world code examples and proof-of-concepts.

- **Cross-site scripting (XSS) vulnerabilities continue to haunt web applications despite decades of awareness...** and are unlikely to disappear anytime soon. Many researchers struggle to move beyond basic payloads when filters and WAFs block their attempts. In our complete XSS exploitation guide, we've broken down a proven 3-step methodology to systematically identify reflected XSS, from mapping reflection points to crafting context-aware payloads that evade common filters.

- **As applications shift logic to the client-side, DOM-based XSS vulnerabilities have become increasingly prevalent...** yet they remain one of the hardest vulnerability types to detect and exploit. Unlike traditional XSS, malicious input flows from a DOM source to a DOM sink without ever appearing in the HTTP response. In our comprehensive guide, we've documented proven methodologies for identifying and exploiting DOM-based XSS.

Tools & resourcesTools

Metis AI-powered security code review

```
$ metis --codebase-path ../demo --backend postgres --project-schema demo
Metis CLI. Type 'help' for usage, 'exit' to quit.
> index
# Indexing codebase...
Indexing completed successfully.
# Reviewing file ../demo/src/allocators.cpp...

File: src/allocators.cpp
  Identified issue 1: Fallback after posix_memalign failure in debug_alloc returns an uninitialized pointer
    Snippet: if (alignment > 8)
    {
        void* ptr;
        int retval = posix_memalign(&ptr, alignment, size);
        if (...
    Why: In debug_alloc, when posix_memalign fails (i.e. retval != 0), the code logs a warning but still returns 'ptr' without assigning
it a safe value. This leaves the possibility that an uninitialized or invalid pointer is returned, potentially causing undefined behavior
or memory corruption when the pointer is used. The fallback path does not actually perform a safe memory allocation such as calling
malloc().
    Mitigation: Modify the fallback behavior to ensure a safe allocation. For example, if posix_memalign fails, call malloc(size) (while
possibly also checking that the alignment requirements are met) or handle the error appropriately by returning nullptr and signaling an
error to the caller.
    Confidence: 1.0

  Identified issue 2: Insufficient fallback in debug_realloc leading to assertion failure and potential production issues
    Snippet: void* debug_realloc(void* pUserData, void* pOriginal, size_t size, size_t alignment, VkSystemAllocato...
    Why: In debug_realloc, if posix_memalign fails, the function asserts false and returns nullptr in production builds where assertions
may be disabled, this can lead to a nullptr being returned unexpectedly during a reallocation, which might lead to use-after-free errors
or crashes if the caller does not handle a failed reallocation properly.
    Mitigation: Improve error handling in debug_realloc by providing a safe fallback (for example, falling back to a standard
realloc-like behavior or explicitly handling the error condition) rather than relying on assert. Ensure that the caller of realloc is
prepared to handle a nullptr return value in a safe manner.
    Confidence: 0.9

Results saved to results/review_file_20250703_093901.json
```

Metis AI-powered security code review

Scanning codebases for vulnerabilities is now possible with AI. Arm Metis is an open-source, AI-driven tool for deep security code review, capable of identifying a wide range of anomalies in codebases.

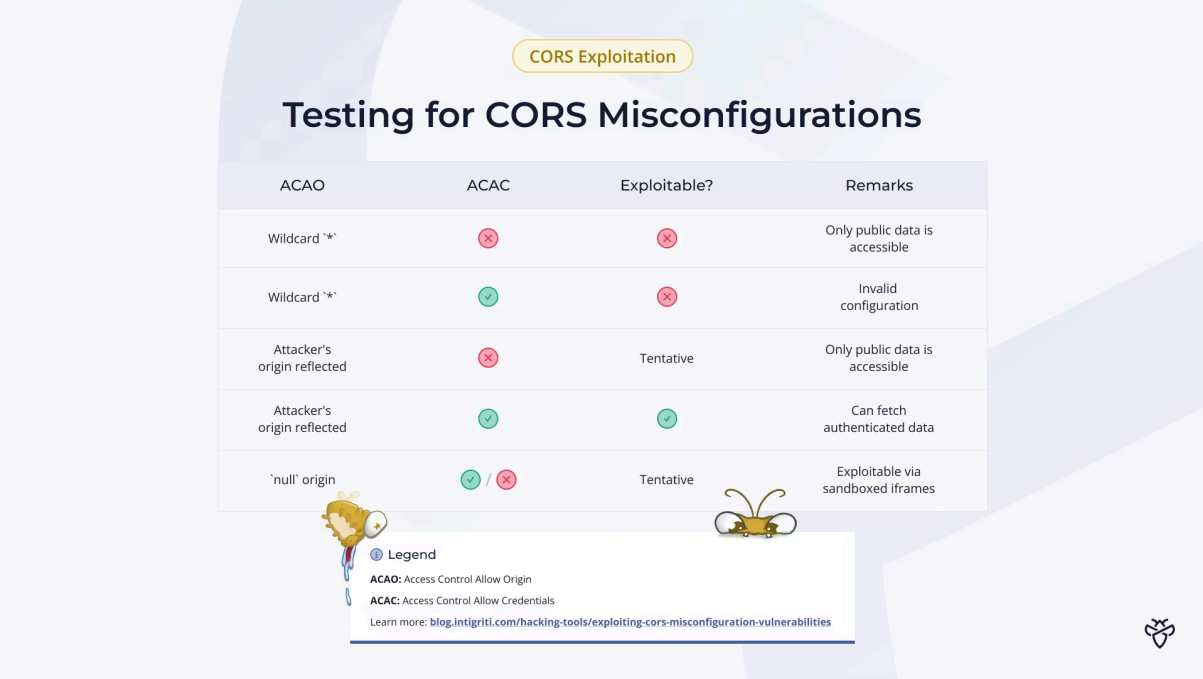
- **Finding misconfigurations in JWT implementations can be as simple as running jwt_tool.** This tool can help you spot all sorts of JWT attacks (including CVEs) to forge your own tokens.

- **Have you found a possible XSS injection point?** And is your target's WAF still getting in the way? Check out JS-DOMestify, a simple tool that converts any JS code to browser-runnable code using only ASCII characters and basic, non-intrusive symbols.

- **Developers commonly set up third-party tools and services incorrectly,** sometimes leaving them wide open for vulnerabilities to arise. Misconfig Mapper is a simple, template-based tool that can help you easily check your list of targets for possible third-party security misconfigurations.

Resources

Exploiting CORS Misconfigurations



Testing for CORS misconfigurations cheat sheet

CORS misconfigurations can result in sensitive data leaks. However they can be unexploitable due to browser restrictions. We've created a small cheat sheet to help you determine the exploitability of any CORS misconfiguration.

If you wish to delve deeper into CORS exploitation, be sure to read our comprehensive article.

- AI tools are being used more in web app hacking. This researcher shares how he scored a critical RCE worth $20K in GitHub using Claude AI.

- Content security policies can be bypassed in various ways. @xssdoctor shares one of the lesser-known methods to evade CSP with a PDF file.

- This latest CVE in Entr'ouvert Lasso covers an interesting SAML-based RCE triggered when you send a malformed request.

- Paywall bypasses can lead to financial losses to companies. @medusa_0xf shares several interesting techniques to bypass payment gateways in her video.

- Everyone likes to find RCEs. This security researcher shares his methodology of finding his first remote code execution vulnerability, earning him a 4-digit bounty.

- Sometimes, simple bugs like IDORs can cause a major impact on the affected organization. Discover how this researcher gained access to the PII of 6.4 million users.
- Bug bounties can be challenging, especially as you're starting. @furkan0x01 shares his experience of how he made 6 figures in his first year as a full-time bug bounty hunter.
- Authentication bypasses are still out there. This bug bounty hunter documents how he found an authentication bypass via an OAuth misconfiguration.
- As development continues to evolve, so do the injection bugs. This researcher found a particularly interesting server-side template injection vulnerability in an AI model.

Feedback & suggestions

Before you click away: Do you have feedback, or would you like your technical content to get featured in the next Bug Bytes issue? We want to hear from you. Feel free to send us an email at support@intigriti.com or DM us on X/Twitter, and we'll take it from there.

Did you like this Bug Bytes issue? Consider sharing it with your friends and tagging us along on X/Twitter, Instagram, or LinkedIn.

Wishing you a bountiful month ahead,

Keep on rocking!

AUTHOR

Ayoub

Senior security content developer