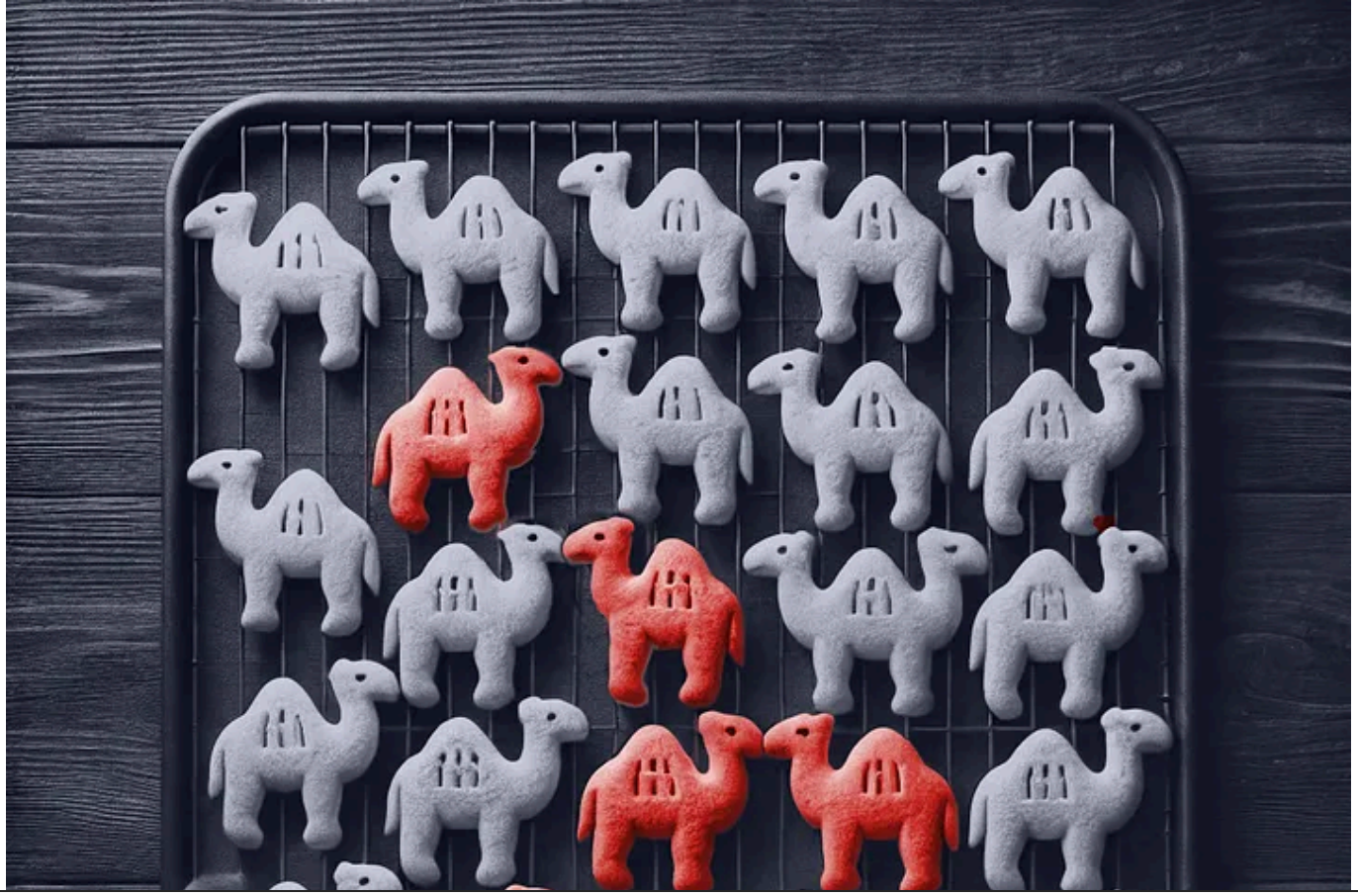
SecuRing

Baking Mojolicious Cookies revisited: a case study of solving security problems through security by obscurity

Known for its agility and simplicity, the Mojolicious framework is



Medium

Sign up to discover human stories that deepen your understanding of the world.

Free

- ✓ Distraction-free reading. No ads.
- ✓ Organize your knowledge with lists and highlights.
- ✓ Tell your story. Find your audience.

✦ Membership

- ✓ Read member-only stories
- ✓ Support writers you read most
- ✓ Earn money for your writing
- ✓ Listen to audio narrations
- ✓ Read offline with the Medium app

Information about the sites and companies using Mojolicious isn't widely documented in easily verifiable public sources. It may or may not be used for many public-facing websites or relied upon for backend services and high-performance internal apps. However, it has a dedicated user base, especially in industries that have historically used Perl (if there were any).

One aspect that caught our attention was how the framework implements sessions, particularly how session data is stored and protected. Mojolicious implements sessions as signed cookies stored on the client side, consisting of the Base64-encoded JSON with session-related information, some padding, and the HMAC-SHA256 signature to prevent tampering.

Sounds just like JWT, right? Remember that:

---

significant discrepancy: the signature algorithm in our case did not match what Antoine had described — it was no longer HMAC-SHA1. Additionally, when I attempted to use popular brute-force tools like *hashcat* and *john* (which are often employed for cracking weak HMAC signatures), I encountered unexpected input validation errors related to the cookies' length. Something was clearly different about this application and was definitely related to the length of session cookies.

Developers' Response to Brute Force Attacks in Mojolicious

As I dug deeper into the issue, I discovered that shortly after Antoine had published his findings, the Mojolicious team made a tiny change to how session cookies are handled. Specifically, they introduced a modification in the framework that pads session values to 1025 bytes by default. This update can be traced back to the pull request #1791 (pad session values to 1025 bytes

There are other examples of long, HMAC-SHA256-signed client-side stored session cookies in use — one of the most prominent ones being JWT (JSON Web Token). It is a popular standard, widely used across many modern web applications as a compact, URL-safe means of representing claims between parties.

Recognizing the similarities between Mojolicious session cookies and JWTs, I decided to simply try combining the capabilities of the hashcat's JWT cracking module that is highly compute optimized and GPU accelerated. It also contains simplified input validation mechanisms used in the HMAC-SHA256 cracking process, previously used for for Mojolicious secrets.

It worked! Here you can find this new module on a merge request: Pull Request #4090 · hashcat/hashcat.

enough may simply show that the application or test environment prepared for the security audit is misconfigured.

Perl | Security | Pentesting | Mojolicious | Security Testing

Published in SecuRing

211 Followers · Last published Apr 24, 2025

Follow

We help to achieve the appropriate level of application security

Written by Jakub Kramarz

Follow

In SecuRing by Jakub Kramarz

Security of External Dependencies in CI/CD Workflows

Use of external dependencies in build processes brings common security risks...

Oct 10, 2024

In SecuRing by Natalia Trojanowska-Korepta

Which Single Sign-On (SSO) is for you? SAML vs OAuth vs OIDC

Comprehensive overview of the most common questions about Single Sign-On...

Apr 28, 2022 12

Recommended from Medium

Sharon Brizinov

How I made $64k from deleted files —a bug bounty story

In InfoSec Write-ups by Iski

🍔 Secret Sauce in Robots.txt 🤫 — How I Found Hidden Admin Panel...

Medium

Sign up to discover human stories that deepen your understanding of the world.

Free

- ✓ Distraction-free reading. No ads.
- ✓ Organize your knowledge with lists and highlights.
- ✓ Tell your story. Find your audience.

✦ Membership

- ✓ Read member-only stories
- ✓ Support writers you read most
- ✓ Earn money for your writing
- ✓ Listen to audio narrations
- ✓ Read offline with the Medium app

loyalonlytoday

Automated Shodan Recon (bug bounty)

Another great tool for bug bounty hunters

Apr 21 92 1

Arrhenius Paelongan

SQL Injection Leads to $$$ Bounty: How I Found a Critical Bug

Bug hunting is a bit like treasure hunting—except instead of digging in dirt, you're...

Apr 14 4 4