WAF bypass: ‘Severe’ OWASP ModSecurity Core Rule Set bug was present for several years

Jessica Haworth 05 July 2021 at 12:31 UTC
Updated: 06 July 2021 at 08:47 UTC

Industry News | Vulnerabilities | Open Source Software

High-scoring bug went unnoticed due to time and money constraints

A vulnerability in the OWASP ModSecurity Core Rule Set (CRS) project that could allow attackers to bypass security mechanisms was present for several years, the maintainers have admitted.

The bug – tracked as CVE-2021-35368 – bypasses the security protections offered by the in-built CRS web application firewall (WAF), an advisory warns.

This means that malicious request body payloads are able to pass the rule set without being inspected, due to a combination of two bugs in the CRS Drupal rule exclusion package.

The vulnerability is not limited to Drupal installations, however, but is present in every CRS installation that includes these rule exclusions – regardless of whether they are enabled or not.

Backend weaknesses exposed

The risk to websites depends on their ModSecurity configuration, a write-up details.

“If the backend is broken and configured with the correct trailing pathname information setting… then anything is possible. If the backend looks into the trailing path info as it should, then you are on the safe side,” it reads.

Christian Folini, co-lead of the volunteer-led Core Rule Set project, told The Daily Swig that the flaw has “been around for several years”.

“When we did the early rule exclusion packages in 2016 and 2017 we were not really used to the rule writing techniques that we had to employ,” Folini explained.

YOU MAY LIKE CVE board slams Distributed Weakness Filing project for publishing ‘unauthorized’ CVE records

“Also [there were] no coding guidelines. And as we [have come] to realize now, nobody has really reviewed this part of the code in the meantime, probably also thanks to the misconception that bugs within the rule exclusion packages would have limited impact since they are disabled by default.

“In retrospect, the failure is obvious. The important part now is learning our lesson and adopting our development and review process to avoid such failures in the future.”

A deeper look

The flaw was discovered and reported by Andrew Howe from Loadbalancer.org, who reviewed the ModSecurity engine last year, Folini said.

Howe reported the two bugs in the CRS in June. All known CRS installations that offer the predefined CRS rule exclusion packages are affected. This also applies to end-of-life CRS versions 3.0.x, 3.1.0, 3.1.1, as well as the currently supported versions 3.2.0 and 3.3.0.

Integrators and users are advised to upgrade to 3.1.2, 3.2.1, and 3.3.2, respectively.

Discussing the challenges faced by a volunteer-led project such as CRS, Folini pinpointed a lack of financial support as a key barrier.

Readmore of the latest open source software security news

Folini said: “Open source is not inherently more secure than closed source – it just means that people can look at the code. Yet the security advantage can only play out when people actually do look at the code, like Andrew Howe did.

Latest Posts

Deserialized web security roundup
Fortinet, Citrix bugs; another Uber breach; hacking NFTs at Black Hat

Critical IP spoofing bug patched in Cacti
‘Not that hard to execute if attacker has access to a monitoring platform running Cacti’

Casting a SpEL
Akamai WAF bypassed via Spring Boot to trigger RCE

"If we have these reviews, then the inherent transparency of an open source project will bring an advantage over traditional software, namely in the security domain where users really want to see what is going deep down in their software.

"Open source projects also tend to be more open about their shortcomings so they are often able to build up more trust and confidence with their user base. A commercial project is often tempted to avoid bad press by keeping a problem under the rug, or hiding a fix in the changelog."

MUST READ Moving security forward by looking back

Conducting these reviews is challenging, however, since the project does not have any funds to pay people for their time.

"You could say we're in a bit of Heartbleed situation," said Folini, referring to the 2014 discovery of an implementation flaw in the widely used OpenSSL cryptographic software library, which went unnoticed due (in part) to a lack of financial support.

"CRS is running on over 100 Tbit/s of traffic and most content delivery networks and cloud providers have a commercial offering where they charge for CRS, yet most of them fail to contribute back to our project so far," he explained.

Folini said that the project maintainers have contacted the organizations using CRS, and are calling for help from the "big players".

"Convincing a commercial company from sponsoring something they can get for free is a hard sell," he explained.

"Yet we have subscribed the first real sponsors in 2021 and I hope that the big players will join them.

"We have a lot of plans where to take our project, yet many of them are on hold until we find the money and resources to implement them."

YOU MAY ALSO LIKE Many Hats Club founder announces closure – spelling end for podcast, conference, Discord community

Industry News | Vulnerabilities | Open Source Software | Secure Development | DevSecOps | DevOps | TLS

Jessica Haworth

@JesscaHaworth

Related stories

Deserialized web security roundup

Fortinet, Citrix bugs; another Uber breach; hacking NFTs at Black Hat

16 December 2022

Critical IP spoofing bug patched in Cacti

15 December 2022

Casting a SpEL

Akamai WAF bypassed via Spring Boot to trigger RCE

14 December 2022

Cloud flaws brought to the fore as bug bounty vulnerabilities hit 65k in 2022

13 December 2022