🕓 2 MIN

DATE
2 NOVEMBER 2020

SERVICESTACK JWT EXPLOIT BAC

AUTHOR
ZI0BLACK

I'm zi0black, Security Researcher and Penetration Tester at Shielder. I love to turn IoT devices in expensive paperweights.

Re-discovering a JWT Authentication Bypass in ServiceStack

TL;DR

ServiceStack before version 5.9.2 failed to properly verify JWT signatures, allowing to forge arbitrary tokens and bypass authentication/authorization mechanisms.
The vulnerability was discovered and patched by the ServiceStack team without highlighting the actual impact, so we chose to publish this blog post along with an advisory.

Routine checks –> Auth bypass

During a Web Application Penetration Test for one of our customers, I noticed that after the login process through a 3rd-party Oauth service the web application used JWT tokens to track sessions and privileges.

Every time I see JWT tokens, I have kind of a routine of tests I do to check for common JWT libraries vulnerabilities.
At some point, during the test, I tried to **remove the signature**, without changing the header and with my big surprise the authenticated API I was testing answered with a "200 OK".

My curiosity was over the top and I had to find a way to **read the source code** to understand what was going on. Unfortunately, it was a BlackBox test 😩 but fortunately, it didn't take much time to achieve **RCE** 😏.

A deep dive into the source code

My first guess was that the customer tried to parse the JWT token manually and failed to implement a proper check for `NULL` signatures.

After I went crazy reading a ton of lines of decompiled `C#` code I finally realized that the customer was using a 3rd-party library called ServiceStack. It was a matter to said it loudly and my fast typer mate Paupu already Google'd it and a GitHub result showed up.

Me and Paupu doing the happy dance

~~0-day~~

By searching in the source code, we made a sad discovery: the developers were already aware of the issue and fixed it in version 5.9.2, just a month before our test. ಥ_ಥ

At this point I was super confused, our customer had no other dependency with known vulnerabilities and it seemed to pay attention to security advisories in the used 3rd-party libraries. Maybe the vendor didn't provide precise information about the vulnerability impact?

Reading the v5.9.2 release notes confirmed my assumption, the only information about the patch is:
If you're using JWT Auth please upgrade to v5.9.2 when possible to resolve a JWT signature verification issue.

So what's the point here?

During one of our weekly team meetings, I shared this finding with my co-workers and we agreed that probably most of the developers were unaware of the actual risk of using ServiceStack before version 5.9.2 because no precise information about the vulnerability impact is publically disclosed. This was also confirmed by our customer during our final report presentation.

I then chose to write this blog post, to ask the MITRE to assign a CVE (CVE-2020-28042) to this vulnerability and to publish an advisory which explains its technical aspects.

We always encourage releasing advisories about vulnerabilities in widely used libraries to help developers keeping their code secure and updated.

Did you develop a web application using JWT to handle sessions/privileges? Contact us for a Penetration Test!

Previous post
Sometimes they come back: exfiltration through MySQL and CVE-2020-11579

Next post

Hunting for bugs in Telegram's animated stickers remote attack surface

INFO

Shielder S.r.l.

P.I. 11435310013

REA TO - 1213132

Registered Capital: 81.000,00 €

Via Palestro, 1/C
10064 Pinerolo (TO) Italy

CONTACTS

info@shielder.com

Landline: (+39) 0121 - 39 36 42

Commercial: (+39) 345 - 30 31 983

Technical: (+39) 393 - 16 66 814

SITEMAP

Home

Company

Services

Advisories

Blog

Careers

Contacts

Disclosure policy

Privacy policy