New issue

Jump to bottom

# CVE-2021-27293: Fix NewDateRegex #1556

Closed b-c-ds opened this issue on Feb 16, 2021 · 13 comments · Fixed by #1557

b-c-ds commented on Feb 16, 2021 • edited Contributor

This advisory was emailed to the maintainer. Posting here as an issue as requested.

## Doyensec Vulnerability Advisory

- Regular Expression Denial of Service (REDoS) in RestSharp
- Affected Product: RestSharp (All released versions)
- Vendor: https://restsharp.dev
- Severity: Medium
- Vulnerability Class: Denial of Service
- Status: Open
- Author: Ben Caller (Doyensec)

## SUMMARY

The .NET library RestSharp uses a regular expression which is vulnerable to Regular Expression Denial of Service (REDoS) when converting strings into DateTimes. If a server responds with a malicious string, the client using RestSharp will be stuck processing it for an exceedingly long time. This allows the remote server to trigger a Denial of Service.

## TECHNICAL DESCRIPTION

The vulnerable regular expression is `NewDateRegex` in `RestSharp.Extensions.StringExtensions`:

RestSharp/src/RestSharp/Extensions/StringExtensions.cs
Line 28 in 0ed7b0a

```
28    static readonly Regex NewDateRegex = new Regex(@"newDate\((-?\d+)*\)");
```

It is used by the `ParseJsonDate` function when deserializing JSON responses into classes with `DateTime` properties.

Due to the `(-?\d+)*` part containing nested repeats, this regular expression has catastrophic backtracking when processing a long string of digits. The behaviour occurs as long as the digits are *not* followed immediately by a closing parenthesis ')'.

An example of a REDoS payload is `new Date(12345678901234567890123456789012345`.

The space between 'new' and 'Date' is required due to pre-processing in `ParseJsonDate`:

RestSharp/src/RestSharp/Extensions/StringExtensions.cs
Lines 124 to 126 in 0ed7b0a

```
124    if (input.Contains("new Date("))
125    {
126        input = input.Replace(" ", "");
```

The complexity is exponential: increasing the length of the malicious string of digits by one makes processing take about twice as long. On my laptop, 27 digits takes about 16 seconds to process and 28 digits takes about 32 seconds, so a string with 54 digits should take approximately 68 years to process.

The vulnerable regular expression was first introduced in commit `373a0a3`

## REPRODUCTION STEPS

The REDoS can be triggered by calling `RestSharp.Extensions.StringExtensions.ParseJsonDate` directly, or by deserializing JSON responses into a class with a property of type `DateTime`.

Example C# code to see the effect of the REDoS is attached below. Changing the length of the string of zeroes will change the processing time.

```
using System;
using System.Globalization;
using RestSharp;
using RestSharp.Extensions;
using RestSharp.Serialization.Json;

namespace DoyensecRestSharpRedosTest
{
    class Thing
    {
        public DateTime Time { get; set; }
    }

    class Program
    {
        static void ByParseJsonDate()
        {
            var redos = "new Date(" + new String('0', 30);
            DateTime dt = StringExtensions.ParseJsonDate(redos, CultureInfo.InvariantCulture);
        }

        static void ByDeserializer()
        {
            var redosJson = @"{""Time"": ""new Date(" + new String('0', 30) + @"""}";
            var thing = (new JsonDeserializer()).Deserialize<Thing>(new RestResponse {Content = redosJson});
            Console.WriteLine(thing.Time);
```

```
        }

        static void ByRequest()
        {
            // Server should return JSON {"Time": "new Date(0000000000000000000000000000000)"}
            Thing thing = new RestClient("https://localhost/thing.json").Execute<Thing>(new RestRequest(Method.GET)).Data;
            Console.WriteLine(thing.Time);
        }

        static void Main(string[] args)
        {
            ByDeserializer(); // or try ByParseJsonDate() or ByRequest()
        }
    }
}
```

## REMEDIATION

Fix `NewDateRegex`. We propose simply removing the asterisk:

```
    static readonly Regex NewDateRegex = new Regex(@"newDate\((-?\d+)\)");
```

.

Doyensec (www.doyensec.com) is an independent security research and development company focused on vulnerability discovery and remediation. We work at the intersection of software development and offensive engineering to help companies craft secure code.



**b-c-ds** added a commit to b-c-ds/RestSharp that referenced this issue on Feb 16, 2021

Fix NewDateRegex in StringExtensions restsharp#1556 ··· c035d73

**b-c-ds** mentioned this issue on Feb 16, 2021

**Fix NewDateRegex in StringExtensions #1556** #1557

Merged

5 tasks

**b-c-ds** changed the title ~~Fix NewDateRegex~~ **CVE-2021-27293: Fix NewDateRegex** on Mar 5, 2021

**b-c-ds** commented on Mar 5, 2021 — Contributor, Author

Assigned CVE-2021-27293

**abhijeet490** commented on Apr 20, 2021

**@bcaller** I can see a fix to this vulnerability, but on which version can we get this update?

**b-c-ds** commented on Apr 20, 2021 — Contributor, Author

**@abhijeet490** The patch has not reviewed or merged. If you want the fix anytime soon you may need to make a fork.

**alexeyzimarev** closed this as completed in #1557 on May 5, 2021

**alexeyzimarev** pushed a commit that referenced this issue on May 5, 2021

Fix NewDateRegex in StringExtensions #1556 (#1557) ··· ✓ be39346

**neil-gok** commented on Jul 6, 2021 • edited

I see that this issue has been closed, but the vulnerability still exists in version 106.11.7. On what version will this fix be released?

👀 3

**b-c-ds** commented on Jul 9, 2021 — Contributor, Author

106.11.8-alpha.0.13 😆

**RajRele** commented on Jul 13, 2021

When could we expect a stable release which includes this fix?

👀 1

**gdoron** commented on Jul 15, 2021

I'm honestly asking, is it the common practice to share CVE publicly **before** the fix was released and adopted?

**gavinBurtonStore...** commented on Jul 15, 2021 • edited

> I'm honestly asking, is it the common practice to share CVE publicly **before** the fix was released and adopted?

CVE's are already public knowledge. But its disappointing that its been since February without a patch.

**@b-c-ds** This issue should not be closed. Can you please confirm whether an official release is planned?

**b-c-ds** commented on Jul 15, 2021 Contributor Author

**@gavinBurtonStoreFeeder** I'm not a maintainer. It won't let me reopen the issue. Sorry. I only reported this issue and contributed one pull request. I don't think this project is super active so I wouldn't hold my breath waiting for a non-alpha release.
Also, if you aren't deserialising DateTimes you should be able to ignore this vulnerability.

**gavinBurtonStore...** commented on Jul 15, 2021

> **@gavinBurtonStoreFeeder** I'm not a maintainer. It won't let me reopen the issue. Sorry. I only reported this issue and contributed one pull request. I don't think this project is super active so I wouldn't hold my breath waiting for a non-alpha release.
> Also, if you aren't deserialising DateTimes you should be able to ignore this vulnerability.

Fair enough. Thanks for raising the GH issue. There is a fix underway but I feel like restsharp is abandonware now.

❤️ 1

**Arslan-Ashfaq** commented on Oct 26, 2021

This issue still exist when this will be fixed.

**alexeyzimarev** commented on Oct 30, 2021 Member

> its been since February without a patch.

The PR was included to 106.12.0 release

> restsharp is abandonware now.

It is a typical ".NET community project", isn't it? 99M downloads, zero sponsors, little contributions from non-maintainers, but lots of complaints (like the comment just above).

Big kudos to **@b-c-ds** for reporting and fixing the issue, it's now released, the issue is closed.

**gavinBurtonStore...** commented on Nov 1, 2021

> > its been since February without a patch.
>
> The PR was included to 106.12.0 release
>
> > restsharp is abandonware now.
>
> It is a typical ".NET community project", isn't it? 99M downloads, zero sponsors, little contributions from non-maintainers, but lots of complaints (like the comment just above).
>
> Big kudos to **@b-c-ds** for reporting and fixing the issue, it's now released, the issue is closed.

You quoted me out of context by starting halfway through the sentence. And I made that comment in july, 5 months after the PR was created.

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

Successfully merging a pull request may close this issue.

**Fix NewDateRegex in StringExtensions #1556**
b-c-ds/RestSharp

8 participants