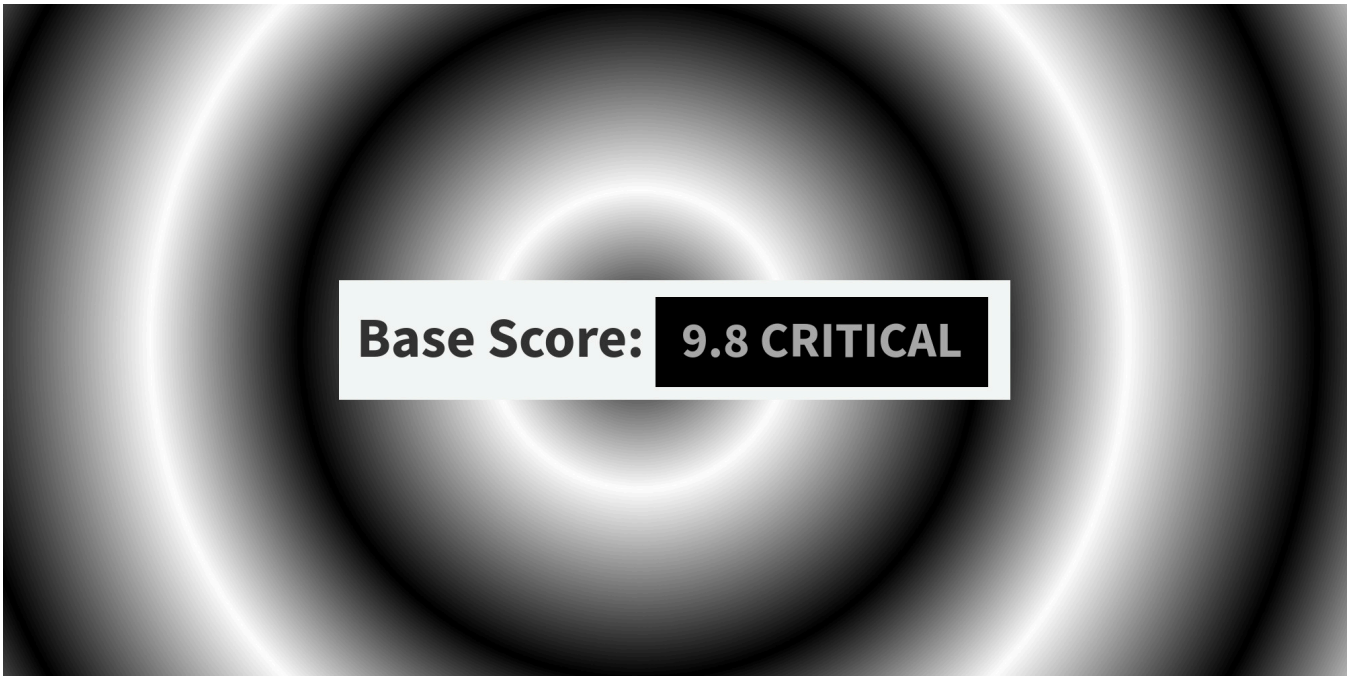

CURL AND LIBCURL

CVE-2020-19909 IS EVERYTHING THAT IS WRONG WITH CVES

AUGUST 26, 2023 | DANIEL STENBERG | 35 COMMENTS

This is a story consisting of several little building blocks and they occurred spread out in time and in different places. It is a story that shows with clarity how our current system with CVE Ids and lots of power given to NVD is a completely broken system.

CVE-2020-19909

On August 25 2023, we got an email to the curl-library mailing list from Samuel Henrique that informed us that "someone" had recently created a CVE, a security vulnerability identification number and report really, for a curl problem.

```
I wanted to let you know that there's a recent curl CVE published and it
doesn't look like it was acknowledged by the curl authors since it's not
```

```
mentioned in the curl website: CVE-2020-19909
```

We can't tell who filed it. We just know that it is now there.

We own our curl issues

In the curl project we work hard and fierce on security and we always work with security researchers who report problems. We file our own CVEs, we document them and we make sure to tell the world about them. We list over 140 of them with every imaginable detail about them provided. We aim at providing gold-level documentation for *everything* and that includes our past security vulnerabilities.

That someone else suddenly has submitted a CVE for curl is a surprise. We have not been told about this and we would *really* have liked to. Now there is a new CVE out there reporting a curl issue and we have no details to say about it on the website. Not good.

I bet curl users soon would like to know the details about this.

Wait 2020?

The new CVE has an ID containing 2020 and that is weird. When you register a CVE you typically get it with the year you request it. Unless you get an ID for an *old* problem of the past. Is that what they did?

Sources seem to indicate that this was published just days ago.

What is this CVE?

Of course the top link when you search for this CVE is to NVD. Not the most reliable organization, but now we can't be too picky. On their site they explain this with very few details:

```
Integer overflow vulnerability in tool_operate.c in curl 7.65.2 via crafted
value as the retry delay.
```

And then the craziest statement of the year. They grade it a **9.8 CRITICAL** issue. With 10 as a maximum, this is close to the worst case possible, right?

The code

Let's pause NVD in their panic state for a moment because I immediately recognized this description. Brief as it is.

I spend a lot of time in the curl security team receiving reports, reviewing reports, reviewing source code, assessing claims and figuring out curl security issues. *I had seen this claim before!*

On July 27, 2019, a Jason Lee file an issue on hackerone, where he reported that there was an integer overflow problem in curl's `--retry-delay` command line option. The option accepts number of seconds and then internally converts to milliseconds by multiplying the value by 1000. The option sets how long time curl should wait until it makes a retry if the previous transfer failed with a transient error.

This means that on a 64 bit machine, if you write

```
curl --retry-delay 18446744073709552 ...
```

The number will overflow the math and instead of waiting until the end of the universe, it might retry again within the next few seconds. The above example apparently made it 384 seconds instead. On Windows, which uses 32 bit longs, you can get the problem already by asking for more than two million seconds (roughly 25 days).

A bug, sure. Security problem? No. I told Jason that in 2019 and then we closed the security report. I then filed a pull-request and fixed the bug. Credits to Jason for the report. We moved on. The fix was shipped in curl 7.66.0, released in September 2019.

Grading issues

In previous desperate attempts from me to reason with NVD and stop their scaremongering and their grossly inflating the severity level of issues, they have insisted that they take in all publicly available data about the problem and make an assessment.

It was obvious already before that NVD really does not try very hard to actually understand or figure out the problem they grade. In this case it is quite impossible for me to understand how they could come up with this severity level. It's like they saw "integer overflow" and figure that *wow, yeah that is the most horrible flaw we can imagine*, but clearly nobody at NVD engaged their brains nor looked at the "vulnerable" code or the patch that fixed the bug. Anyone that looks can see that this is not a security problem.

The issue listed by NVD even links to my pull request I mention above. There is no doubt that it is the exact same bug they refer to.

Spreading like a virus

NVD hosts a CVE database and there is an entire world and eco system now that pulls the records from them.

NVD now has this CVE-2020-19909 entry in there, rated 9.8 CRITICAL and now this disinformation spreads across the world. Now when we search for this CVE number we find numerous sites that repeats the same data. "This is a 9.8 CRITICAL problem in curl" – when it is not.

I will object

I learned about this slap in my face just a few hours ago (and I write this past Friday midnight), but I intend to do what I can to reject this CVE.

Update: I'm glad to see the Ubuntu took the lead and marked it as not-affected.

Update2: MITRE denied my request to reject the CVE. The full reason from them is now included in my description of CVE-2020-19909 on the curl site.

Update3: a follow-up post.

Discuss

Hacker news. Reddit. Lobste.rs.

CURL AND LIBCURL SECURITY

35 THOUGHTS ON "CVE-2020-19909 IS EVERYTHING THAT IS WRONG WITH CVES"

natsuki

AUGUST 26, 2023 AT 02:15

Come on.. you know who NVD really is……..

John

AUGUST 26, 2023 AT 09:36

How ridiculous! I thought only project maintainers could file CVEs!

Thomas Broyer

AUGUST 26, 2023 AT 12:01

If only project maintainers could file CVEs, no doubt many would have never been filed, vendors attempting to hide vulnerabilities rather than acknowledge them.

Willy

AUGUST 26, 2023 AT 14:16

Most developers (especially in opensource) are totally open with bugs. CVEs just add useless bureaucracy on a small fraction of important bugs and waste a lot of time to those trying to work on fixing the issues. Really. In haproxy all bugs and their impacts are clearly documented in commit messages, and there are probably way more issues documented there than what some could consider as representing a security risk, but at least all the info is available, without spending 3 days of work writing a stupid report that is then re-edited by people who know nothing about the product and decide to change affected versions as if it was fun to confuse users and distros.

Timm

AUGUST 28, 2023 AT 07:53

“Really. In haproxy all bugs and their impacts are clearly documented in commit messages.”

While that might be the case for this (and probably others) – how do you check if your current system setup is affected by any known vulnerabilities? Once you got more than 10 dependencies, you probably won’t be reading the commit message of every single one of them. If it’s done by a distro – fine; But not everything is packaged in distros (think log4j).

I think we actually do need a central database for this. Does the current setup have issues? Sure. Are there false (and sometimes fraudulent) reports? Sure. Is falling back to reading commit messages of a 3 to 4-digit number of dependencies of your project the right way? Certainly not.

I think there is a lot to be done in the ecosystem of “vulnerability management” (and I think “don’t manage, fix!” would be better) – and other things like GHSA and OSSI are also there by now — but NVD is what started it all.

So what’s the alternative to replace NVD – and where should we go?

Willy

AUGUST 30, 2023 AT 08:43

@Timm:
> how do you check if your current system setup is affected by any known vulnerabilities

Right now you already don't know, that's even worse. You can only know if it's affected by the rare vulnerabilities that get a CVE assigned. And worse, many distros will only ship these fixes and not all the other important ones because they trust a bogus metric. Some might even *introduce* vulnerabilities by backporting fixes where they were not needed (already seen in the kernel component in some distros in the past).

> I think we actually do need a central database for this.

I think not because most of the time nobody can decide what really is a vulnerability. I've seen distros reject some haproxy fixes in the past while they could result in trivial DoSes for some users. The current bug here is a perfect example of this. The only thing that matters is to be *up to date* and to insist on projects to maintain stable versions and cooperate with operating system vendors to provide fixes in a timely manner. As long as projets are not ultra-careful about their fixes, distros and vendors will be wary of taking all the fixes, and will instead cherry-pick random fixes. And as long as they'll do this, end-users will remain vulnerable in field and will continue to trust the bogus CVE metric.

Espie Marc

AUGUST 31, 2023 AT 01:18

I'm old enough to remember when CVE actually fixed a problem. Back at the beginning of disclosures, downstream vendors all issued individual notices. It was a bit difficult to figure out which were duplicates, because all of HP, IBM, Redhat, Sun, SGI … would report the same problem in wildly different ways with varying degrees of details.

At that point, the CVE process made it simpler to figure out whether it was already a problem you knew about, regardless of vendor. This was a time-saver for every system administrator juggling with multiple systems at the time (and we generally knew enough about software code to assert criticity by ourselves)

The problem with severity grades came much later, when in some shady part of the security community, recognition became associated with having found "vulnerabilities".

dk

AUGUST 26, 2023 AT 12:55

That's horrible, fun fact it's not the first time !
I found myself in the same situation in the PAST when searching for vulnerabilities on apache.

droidmonkey

AUGUST 26, 2023 AT 13:32

KeePassXC Team feels your pain. Sorry that this happened to you too!

helio

AUGUST 26, 2023 AT 14:10

Hello there.
Out of curiosity, you say:
"Of course the top link when you search for this CVE is to NVD. Not the most reliable organization…"
What would be the best source to consult CVEs?
Thank you.

★ Daniel Stenberg

AUGUST 26, 2023 AT 19:12

The sad truth is that there just is no best single source for CVE information. In the example of curl, there is no reliable source for curl related CVEs beside our own records on curl.se. I bet that's similar for lots of other projects/products as well. It does not scale at all. It is quite terrible.

Mark Cox

AUGUST 30, 2023 AT 08:13

You could make CVE a reliable source for curl related CVEs by submitting them to the project and looking after them yourself. We do this for OpenSSL and Apache and this problem as stated in the blog then goes away. Joining the program is simple, lightweight, without contract and smaller projects can join a CVE root if they prefer to get help and advice with it. See https://www.cve.org/ProgramOrganization/CNAs

★ Daniel Stenberg

AUGUST 30, 2023 AT 08:41

@Mark: yes thanks, so I've been told and I have initiated that plan. It is still unfortunate. It does not seem like a system that scales for the entire world.

Joe

AUGUST 27, 2023 AT 00:21

Not sure how reliable they are compared to NIST's NVD, but I usually use https://www.cvedetails.com/ or https://cve.mitre.org/

Drew

AUGUST 28, 2023 AT 02:24

They both use the data from NIST's NVD, they're not separate databases.

Luis

AUGUST 27, 2023 AT 01:43

I use attackerkb.com. It doesn't have information on all the CVEs, but the ones they do have are often real security issues with great analysis and insight.

Louis

AUGUST 28, 2023 AT 16:25

Lately I have been using https://ess.coalitioninc.com/ it's really clean and has all the info in a single place!

Willy

AUGUST 26, 2023 AT 14:12

I feel sad for you to discover this state of affairs that late. I've been through this as well, even with mitre publishing all gory details on twitter just after I filed the entry myself while it was still under embargo! That was my last one by the way. Too much hassle and too much trouble caused to users in the end.

For being on the linux kernel security list I can see that there are reporters who are just seeking a CVE to add a line on their CV, which is why I'm calling CVE "Curriculum Vitae Enhancer". Whatever tiny bug, strange behavior or documentation weakness will eventually be turned into a CVE if you wait long enough, because there is a huge business behind this from companies selling as a premium service to patch, virtual patch, or at least notify CISOs. They're even willing to reward junkies who need a collection of CVEs as trophees to show on their CV and brag with friends.

Nowadays my estimate is that maybe 1/4 to 1/3 of CVEs do really concern security. The rest is advertising either for the reporter or for the product itself (some do report dummy CVEs themselves in hope that it will spread the word about their product). The vast majority of security do not have a CVE assigned since most of them are fixed without even noticing that a bug could have a security impact.

I encourage you to watch Greg KH's presentation about "CVE are dead", it even enlights another bad practise showing that developers sometimes have to pretend a bug is a security issue to be allowed by their manager to backport the fix. Ah it's here: https://www.youtube.com/watch?v=HeeoTE9jLjM

★ Daniel Stenberg

AUGUST 26, 2023 AT 19:14

@Willy: unfortunately, the Linux kernel is a unique project and they can do whatever they want. I don't feel that I have that wiggle room for projects I run. I feel there is an expectation that I need to conform at least mostly to in order to get treated and accepted properly.

Willy

AUGUST 26, 2023 AT 20:38

I know what you mean, but your project is well reputed and your professional reputation doesn't have to be proven anymore. As long as you let interested parties deal with filing CVEs for you when they *think* it's deserved, it can save you some time and pain, and let them deal with that usually with good contacts to fix the annoying stuff. Nowadays the CVEs in haproxy are filed by debian/SuSE/RedHat security teams essentially, and they're pre-filled correctly. The real problem is when everyone knows a CVE makes zero sense for something trivial and the reporter nevertheless wants it. For us till now, simply crediting them for the finding in commit messages was sufficient to give them the recognition they were possibly looking for. But that's true that you can never bet certain it will always suffice.

★ Daniel Stenberg

AUGUST 26, 2023 AT 20:55

@Willy: Thanks for all valuable feedback. I am going to reconsider my general take on all this going forward. Something is going to have to change.

Jeff Gonzalez

AUGUST 26, 2023 AT 18:16

Douglas Hubbard outlines in his “How to Measure Anything in Cybersecurity “ book how ridiculous the CVSS scoring stuff is in the context of risk. A lot of cybersecurity is just theatre at this point.

smsm42

AUGUST 26, 2023 AT 20:15

You may want to register as CNA at https://www.cve.org/PartnerInformation/ListofPartners

★ Daniel Stenberg

AUGUST 26, 2023 AT 20:34

@smsm42: right, I am looking into this.

Charlie Van Noland

AUGUST 27, 2023 AT 13:45

What about software that’s no longer maintained? That the developers have abandoned, or have passed away from this plane of existence? 3rd parties should still be able to let the world know that the software they’re using is vulnerable, *especially* if developers are no longer maintaining it for whatever reason.

Simon Geard

AUGUST 27, 2023 AT 21:33

Yes, I figured out pretty quickly when I started doing security work that you need to assess any CVE with a critical eye. They’re mostly good, but I have a few suppressed in our reporting tools because they’re stupid reports which are disputed by the projects for reasons similar to what you described. It’s a pain, because it can be quite time consuming to verify that something like this really isn’t a threat ..

Patrick Batenburg

AUGUST 28, 2023 AT 16:00

I am one of those people that pulls the NVD regularly (at least weekly) and was very surprised to see this one show up. CVE-2020, mentioning version 7.65, very high score, odd to say the least. Had not been able to investigate further yet but the SANS daily mentioned your blog today. Thanks for the clarification!

Art

AUGUST 28, 2023 AT 18:59

I'd argue this should not get a CVE ID assignment, self-pwn is not a security policy violation, IOW, 'rm' is much more dangerous than 'curl –retry-delay'. I see no way for an attacker to set or inflence the value of –retry-delay.

I suspect NVD mistakently considered the library use of curl/libcurl, maybe assuming that an application using libcurl could be exploited to execute arbitrary code, using this integer overflow as a starting point. AFAICT libcurl does not implement any retry options, the application would have to do this, so again, no vulnerability in curl/libcurl.

Stewart Smith

AUGUST 28, 2023 AT 23:30

Amazon Linux is currently rating as Moderate / CVSSv3 4.0 over at
https://explore.alas.aws.amazon.com/CVE-2020-19909.html

Considering when this was fixed, the "Fix Pending" for AL1 and AL2 will be that we list is creating an Advisory on https://alas.aws.amazon.com/ listing it and having the metadata in the yum repositories updated with this metadata.

Kazuki Omo

SEPTEMBER 3, 2023 AT 04:48

Hi, I totally agree with you and disappointed the situation, but
> The new CVE has an ID containing 2020 and that is weird.
Unfortunately, it's happen always.
When you see CVE-ID assign process in below;
https://www.cve.org/About/Process

It says
"The "Year" portion is the year that the CVE ID was reserved or the year the vulnerability was made public. The year portion is not used to indicate when the vulnerability was discovered. "

So someone already reserved it on 2020, or someone find and assinged "the vulnerability was discovered and published on 2020 and fixed it already", MITRE(or CNA) assing CVE-2020-XXX.

Zach Bloomquist

SEPTEMBER 4, 2023 AT 17:38

Looks like cURL and SQLite have the same woes: https://www.sqlite.org/cves.html

Previously I worked on an open source project that pulled in many third party libraries. Users would run their corpo vulnerability scanners on the project and find dependencies with open CVEs and demand fixes, not understanding that in our usage of the libraries, the vulnerability is not exposed.

I think in 4 years, we had users open roughly 50 issues like this, which corresponded to exactly 0 real world exploitable issues.

A central vuln DB makes sense for sysadmins, but too many make it the end-all-be-all.

David

SEPTEMBER 7, 2023 AT 17:00

if (CVE.find('overflow')!=-1) severity = 9+random.random();

Yann Vernier

SEPTEMBER 9, 2023 AT 10:22

I think the lower bound value for this particular overflowing multiply of 64-bit 2's complement long is actually 384 milliseconds, not seconds (and 704ms for 32-bit). The CVE is still a waste of time and the 9.8 scored rating was fraudulent. It's also ridiculous how some focus on how the rating translates to a score instead of how the rates were fabrications.

Clinton Kerrison

SEPTEMBER 13, 2023 AT 00:33

Arguably, rather than outright scrapping the issue, there is a case for changing the rating.
The network vector doesn't make sense, as this requires manipulating the commandline arguments, and should be a physical vector. Also based on the mitre response, only the availability impact should be applied, and probably as a low. This would change the rating to: Low 2.4 (https://www.first.org/cvss/calculator/3.1#CVSS:3.1/AV:P/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:L)

wh

SEPTEMBER 15, 2023 AT 15:46

I am so relieved to have stumbled upon this topic!

Recently I had to evaluate a software system which had ~2000 active CVE-s associated with its software components.

Obviously when doing this I had to do prioritise the work, and I did that based on attack vector and overall score. I have found that in many cases similar to: “user opens a forged email from which the client sw can crash” named as a network vulnerability.

For me as a user of the NVD, I found that certain claims are inflated and it creates noise in which real issues are harder to find.

I thought that the inflated claims of severity come from researchers who want to show how impactful their finds are..

COMMENTS ARE CLOSED.