



## Bug 1642201 (CVE-2018-16839) - CVE-2018-16839 curl: Integer overflow leading to heap-based buffer overflow in Curl\_sasl\_create\_plain\_message()

**Keywords:** Security

**Status:** CLOSED ERRATA

**Alias:** CVE-2018-16839

**Product:** Security Response

**Component:** vulnerability

**Version:** unspecified

**Hardware:** All

**OS:** Linux

**Priority:** low

**Severity:** low

**Target Milestone:** ---

**Assignee:** Red Hat Product Security

**QA Contact:**

**Docs Contact:**

**URL:**

**Whiteboard:**

**Depends On:** [1644552](#) [1644553](#) [1644554](#)  
 [1652520](#)

**Blocks:** [1642204](#)

**TreeView+** [depends on](#) / [blocked](#)

**Reported:** 2018-10-24 01:44 UTC by Sam Fowler

**Modified:** 2021-12-10 18:02 UTC ([History](#))

**CC List:** 29 users ([show](#))

**Fixed In Version:** curl 7.62.0

**Clone Of:**

**Environment:**

**Last Closed:** 2021-10-25 09:51:32 UTC

**Embargoed:**

**Attachments** ([Terms of Use](#))

Sam Fowler 2018-10-24 01:44:16 UTC

[Description](#)

Curl versions 7.33.0 to 7.61.1 are vulnerable to a buffer overrun in the SASL authentication code.

The internal function ``Curl_auth_create_plain_message`` fails to correctly verify that the passed in lengths for name and password aren't too long, then calculates a buffer size to allocate.

On systems with a 32 bit ``size_t``, the math to calculate the buffer size triggers an integer overflow when the user name length exceeds

2GB ( $2^{31}$  bytes). This integer overflow usually causes a very small buffer to actually get allocated instead of the intended very huge one, making the use of that buffer end up in a heap buffer overflow.

Sam Fowler 2018-10-24 01:44:33 UTC

[Comment 1](#)

Acknowledgments:

Name: the Curl project  
Upstream: Harry Sintonen

Sam Fowler 2018-10-31 07:00:12 UTC

[Comment 2](#)

External Reference:

<https://curl.haxx.se/docs/CVE-2018-16839.html>

Upstream Patch:

<https://github.com/curl/curl/commit/f3a24d7916b9173c69a3e0ee790102993833d6c5>

Sam Fowler 2018-10-31 07:00:53 UTC

[Comment 3](#)

Created curl tracking bugs for this issue:

Affects: fedora-all [ [bug-1644552](#) ]

Created mingw-curl tracking bugs for this issue:

Affects: epel-7 [ [bug-1644553](#) ]

Tomas Hoger 2018-11-15 21:23:48 UTC

[Comment 5](#)

As noted above, this problem was introduced in Curl version 7.33.0 via the following commit:

<https://github.com/curl/curl/commit/c56f9797e7feb7c2dc>

Prior to that commit, the username and password lengths were limited to `MAX_CURL_USER_LENGTH` or `MAX_CURL_PASSWORD_LENGTH`, i.e. to 256 characters.

This did not affect curl packages in Red Hat Enterprise Linux 7 and earlier, which are based on upstream Curl versions prior to 7.33.0.

Tomas Hoger 2018-11-22 09:40:56 UTC

Comment 6

To trigger this overflow, user name and password with total length greater than `SIZE_T_MAX/2` must be provided - for example user name with length right below `SIZE_T_MAX` and short password. The problem is not inherently 32 bit, but providing inputs of sufficient size on 64 bit platforms is most likely outside of what computers today can do.

Also note that it does not seem to be possible to trigger this flaw on 32 bit platforms either, as multiple copies of user name and password are made in memory when using standard libcurl APIs, so an out-of-memory error would be hit before the problematic code in the `Curl_auth_create_plain_message()` function can be reached. Quoting the notes I added to the upstream commit:

<https://github.com/curl/curl/commit/f3a24d7916b9173c69a3e0ee790102993833d6c5#commitcomment-31315766>

""""

I'm also wondering if any PoC was provided for this, or a hint on a path to trigger this issue? To trigger the overflow, this requires input that is longer than `SIZE_T_MAX/2` (user name at or slightly below `SIZE_T_MAX/2`, and few bytes for password). The user name and password passed to `Curl_auth_create_plain_message` comes from `conndata` struct. Setting them there seems to be `set_login()`'s job. However, `set_login()` copies / `strdups` both of them, so at the end of that function user name and password needs to be in memory in two copies, which is not possible if single copy requires more than `SIZE_T_MAX/2` of space. Also what's passed to `set_login()` already is a copy of data also stored elsewhere in the memory (with copy created in `parseurlandfillconn()` or `override_login()`).

So it does not seem triggerable when using libcurl APIs.

""""

Note

You need to [log in](#) before you can comment on or make changes to this bug.

