```
From 56b0509dfc6b559cd7555ea81ee62e3622069255 Mon Sep 17 00:00:00 2001
From: =?UTF-8?q?Petr=20P=C3=ADsa=C5=99?= <ppisar@redhat.com>
Date: Thu, 13 Dec 2018 13:05:07 +0100
Subject: [PATCH] Fix a buffer overwrite in parse_stream()

The parse_stream() function allocates BUFSIZE-byte long output buffer. Then it
reads a string using PerlIO's read() with a maximal string length tsiz=BUFSIZE
characters into a temporary buffer. And then it retrieves a length of the string
in the temporary buffer in bytes and copies the strings from the temporary
buffer to the output buffer.

While it works for byte-stream file handles, when using UTF-8 handles, length
in bytes can be greater than length in characters, thus the temporary buffer
can contain more bytes than the size of the output buffer and we have a buffer
overwrite. This corrupts memory, especially metadata for libc memory
management and subsequent free() aborts with "free(): invalid next size
(normal)".

Minimal reproducer: Execute this code with an UTF-8 encoded file with non-ASCII
charcters on the standard input:

use XML::XPath;
use open ':std', ':encoding(UTF-8)';
my $xpath = XML::XPath->new(ioref => \*STDIN);
$xpath->find('/');

https://bugzilla.redhat.com/show_bug.cgi?id=1473368
https://bugzilla.redhat.com/show_bug.cgi?id=1658512
---
 Expat/Expat.xs | 10 ++++++----
 1 file changed, 6 insertions(+), 4 deletions(-)

diff --git a/Expat/Expat.xs b/Expat/Expat.xs
index ed66531..dbad380 100644
--- a/Expat/Expat.xs
+++ b/Expat/Expat.xs
@@ -343,8 +343,8 @@ parse_stream(XML_Parser parser, SV * ioref)
   }
   else {
     tbuff = newSV(0);
-    tsiz = newSViv(BUFSIZE);
-    buffsize = BUFSIZE;
+    tsiz = newSViv(BUFSIZE); /* in UTF-8 characters */
+    buffsize = BUFSIZE * 6; /* in bytes that encode an UTF-8 string */
   }

   while (! done)
@@ -386,9 +386,11 @@ parse_stream(XML_Parser parser, SV * ioref)
          croak("read error");

        tb = SvPV(tbuff, br);
-       if (br > 0)
+       if (br > 0) {
+         if (br > buffsize)
+           croak("The input buffer is not large enough for read UTF-8 decoded string");
          Copy(tb, buffer, br, char);
-       else
+       } else
          done = 1;

        PUTBACK ;
```