Bug 26224 (CVE-2020-27618) - iconv hangs when converting some invalid inputs from several IBM character sets (CVE-2020-27618)

Status:	RESOLVED FIXED	**Reported:**	2020-07-09 20:21 UTC by Arjun Shankar
		Modified:	2020-11-06 02:19 UTC (History)
Alias:	CVE-2020-27618	**CC List:**	2 users (show)
Product:	glibc	**See Also:**	~~CVE-2016-10228~~
Component:	locale (show other bugs)	**Host:**	
Version:	unspecified	**Target:**	
		Build:	
Importance:	P2 normal	**Last reconfirmed:**	
Target Milestone:	2.33		
Assignee:	Arjun Shankar	**Flags:**	fweimer: security+
URL:			
Keywords:			
Depends on:			
Blocks:			

Attachments
Add an attachment (proposed patch, testcase, etc.)

Note
You need to log in before you can comment on or make changes to this bug.

Arjun Shankar 2020-07-09 20:21:51 UTC Description

```
I found the following hangs when running an iconv input fuzzer I wrote while trying
to fix ~~bug 19519~~. The hangs are present in master:

echo -en '\x00\x0f' | iconv -t UTF-8 -c -f IBM1364
echo -en '\x00\x0f' | iconv -t UTF-8 -c -f IBM1371
echo -en '\x00\x0f' | iconv -t UTF-8 -c -f IBM1388
echo -en '\x00\x0f' | iconv -t UTF-8 -c -f IBM1390
echo -en '\x00\x0f' | iconv -t UTF-8 -c -f IBM1399

These hangs are presently mentioned but commented out in iconv/tst-iconv_prog.sh
and should eventually be un-commented when this bug is fixed.

The fuzzer itself (attachment 11786 [details]) should also be run against these
character sets after this bug is fixed, because they skip all remaining inputs for
a character set once they encounter a hang in the corresponding converter, and thus
any other hangs (from possibly other bugs) aren't tested for.
```

Siddhesh Poyarekar 2020-11-06 02:19:15 UTC Comment 1

```
Fixed by:

commit 9a99c682144bdbd40792ebf822fe9264e0376fb5
Author: Arjun Shankar <arjun@redhat.com>
Date:   Wed Nov 4 12:19:38 2020 +0100

    iconv: Accept redundant shift sequences in IBM1364 [BZ #26224]

    The IBM1364, IBM1371, IBM1388, IBM1390 and IBM1399 character sets
    share converter logic (iconvdata/ibm1364.c) which would reject
    redundant shift sequences when processing input in these character
    sets.  This led to a hang in the iconv program (CVE-2020-27618).

    This commit adjusts the converter to ignore redundant shift sequences
    and adds test cases for iconv_prog hangs that would be triggered upon
    their rejection.  This brings the implementation in line with other
    converters that also ignore redundant shift sequences (e.g. IBM930
    etc., fixed in commit 692de4b3960d).

    Reviewed-by: Carlos O'Donell <carlos@redhat.com>
```