**Bug #817**

**Crash on invalid tuple D-Bus request**

**August 09, 2018 02:31 - Tom Thorogood**

<table>
<thead>
<tr>
<th>Status:</th>
<th>Closed</th>
<th>Start date:</th>
<th>August 09, 2018</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Major</td>
<td>Due date:</td>
<td>100%</td>
</tr>
<tr>
<td>Assignee:</td>
<td></td>
<td>% Done:</td>
<td></td>
</tr>
<tr>
<td>Category:</td>
<td>core</td>
<td>Estimated time:</td>
<td>0.00 hour</td>
</tr>
<tr>
<td>Target version:</td>
<td>3.10.1</td>
<td>Affects version:</td>
<td>3.10</td>
</tr>
</tbody>
</table>

**Description**

I was able to trigger an easily reproducible crash through the D-Bus interface. The org.atheme.audacious.SongTuple method does not valid the requested tuple in `do_song_tuple` before calling `get_value_type`. This triggers an assert in `get_value_type`, but could potentially still crash otherwise. Instead of a crash, I'd expect either an empty string or an error to be returned.

The crash can be reproduced with:

```
dbus-send --session --print-reply --dest=org.atheme.audacious /org/atheme/audacious org.atheme.audacious.SongTuple uint32:0 string:x
```

It crashes with the following backtrace:

```
#0 0x00007ffff638afeb in raise () at /lib64/libc.so.6
#1 0x00007ffff6375c1 in abort () at /lib64/libc.so.6
#2 0x00007ffff6375491 in __nls_load_domain.cold.0 () at /lib64/libc.so.6
#3 0x00007ffff6383752 in () at /lib64/libc.so.6
#4 0x00007ffff7bb1d26 in Tuple::get_value_type(Tuple::Field) const (this=0x7fffffffc970, field=field@entry=Tuple::Invalid) at tuple.cc:450
#5 0x0000555555555f66d in do_song_tuple(_ObjAudacious*, _GDDBusMethodInvocation*, unsigned int, char const*) (obj=0x55555557a4ce0, invoc=0x5555555cf3260, pos=0, key=<optimized out>) at dbus-server.cc:644
#6 0x00007ffff5a9e03e in ffi_call_unix64 () at /lib64/libffi.so.6
#7 0x00007ffff5a9df9f in ffi_call () at /lib64/libffi.so.6
#8 0x00007ffff7379a5 in _gclosure_marshal_generic () at /lib64/libgobject-2.0.so.0
#9 0x00007ffff7378add in _g_closure_invoke () at /lib64/libgobject-2.0.so.0
#10 0x00007ffff73bf43 in signal_emit_unlocked_R () at /lib64/libgobject-2.0.so.0
#11 0x00007ffff73941f9 in g_signal_emitv () at /lib64/libgobject-2.0.so.0
#12 0x0000555555555f66d in do_song_tuple(_ObjAudacious*, _GDDBusMethodInvocation*, unsigned int, char const*) (obj=0x55555557a4ce0, invoc=0x5555555cf3260, pos=0, key=<optimized out>) at aud-dbus.c:15178
#13 0x00007ffff7909f6 in g_dbus_interface_method_dispatch_helper () at /lib64/libgio-2.0.so.0
#14 0x00007ffff76e50 in call_in_idle_ch () at /lib64/libgio-2.0.so.0
#15 0x00007ffff709acb in _g_idle_dispatch () at /lib64/libgio-2.0.so.0
#16 0x00007ffff709a6d in g_main_context_dispatch () at /lib64/libgio-2.0.so.0
#17 0x00007ffff709d78 in g_main_context_terminate.isra () at /lib64/libgio-2.0.so.0
#18 0x00007ffff709f4a2 in g_main_loop_run () at /lib64/libgio-2.0.so.0
#19 0x00007fffffcd3265f in gtk_main () at /lib64/libgtk-x11-2.0.so.0
#20 0x00007ffff79c5e5 in interface_run() () at interface.cc:163
#21 0x00007fffff7baf3e in aud_run() () at runtime.cc:323
#22 0x0000555555555e774 in main (argc=<optimized out>, argv=<optimized out>) at main.cc:395
#23 0x00007ffff637724b in __libc_start_main () at /lib64/libc.so.6
#24 0x0000555555555ed7a in __start () at main.cc:410
```
I'm running into this with Audacious 3.10-beta1 (from https://copr.fedorainfracloud.org/coprs/mschwendt/audacious-next/).

History

#1 - August 19, 2018 18:17 - John Lindgren
- % Done changed from 0 to 100
- Target version changed from 3.10 to 3.10.1
- Status changed from New to Closed
- Category set to core

Fixed:
https://github.com/audacious-media-player/audacious/commit/1528e780825b1fc49639ea16d332c5752051dab7

#2 - August 19, 2018 18:19 - John Lindgren
- Affects version 3.10 added