I was able to trigger an easily reproducible crash through the D-Bus interface. The `org.atheme.audacious.SongTuple` method does not validate 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:

```bash
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:

```bash
#0 0x00007ffffff638afeb in raise () at /lib64/libc.so.6  
#1 0x00007ffffff63755c1 in abort () at /lib64/libc.so.6  
#2 0x00007ffffff6375491 in __nix_load_domain.cold.0 () at /lib64/libc.so.6  
#3 0x00007ffffff6383752 in () at /lib64/libc.so.6  
#4 0x00007ffffff7bb1d26 in Tuple::get_value_type(Tuple::Field) const (this=0x7fffffffc970, field=field@entry=Tuple::Invalid) at tuple.cc:450  
#5 0x000055555555f66d in do_song_tuple(_ObjAudacious*, _GDBusMethodInvocation*, unsigned int, char const*) (obj=0x55555557a4c0, invoc=0x5555555cf360, pos=0, key=) at dbus-server.cc:644  
#6 0x00007ffffff5a69df in ffi_call_unix64 () at /lib64/libffi.so.6  
#7 0x00007ffffff5a69df in ffi_call () at /lib64/libffi.so.6  
#8 0x00007ffffff73795a5 in g_closure_marshall_generic () at /lib64/libgobject-2.0.so.0  
#9 0x00007ffffff7378add in g_closure_invoke () at /lib64/libgobject-2.0.so.0  
#10 0x00007ffffff72bbf43 in signal_emit_unlocked_R () at /lib64/libgobject-2.0.so.0  
#11 0x00007ffffff739419f in g_signal_emitv () at /lib64/libgobject-2.0.so.0  
#12 0x000055555555f66d in do_song_tuple(_ObjAudacious*, _GDBusMethodInvocation*, unsigned int, char const*) (obj=0x55555557a4c0, invoc=0x5555555cf360, pos=0, key=) at dbus-server.cc:644  
#13 0x00007ffffff5a69df in ffi_call_unix64 () at /lib64/libffi.so.6  
#14 0x00007ffffff5a69df in ffi_call () at /lib64/libffi.so.6  
#15 0x00007ffffff73795a5 in g_closure_marshall_generic () at /lib64/libgobject-2.0.so.0  
#16 0x00007ffffff7378add in g_closure_invoke () at /lib64/libgobject-2.0.so.0  
#17 0x00007ffffff739419f in g_signal_emitv () at /lib64/libgobject-2.0.so.0  
#18 0x00007ffffff739419f in g_signal_emitv () at /lib64/libgobject-2.0.so.0  
#19 0x00007ffffff739419f in g_signal_emitv () at /lib64/libgobject-2.0.so.0  
#20 0x00007ffffff739419f in g_signal_emitv () at /lib64/libgobject-2.0.so.0  
#21 0x00007ffffff739419f in g_signal_emitv () at /lib64/libgobject-2.0.so.0  
#22 0x00007ffffff739419f in g_signal_emitv () at /lib64/libgobject-2.0.so.0  
#23 0x00007ffffff739419f in g_signal_emitv () at /lib64/libgobject-2.0.so.0  
#24 0x00007ffffff739419f in g_signal_emitv () at /lib64/libgobject-2.0.so.0  
```

May 01, 2020 1/2
I'm running into this with Audacious 3.10-beta1 (from https://copr.fedorainfracloud.org/coprs/mschwends/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