Exiv2 - Bug #960

Problem With Exiv2 (Video files)

05 Jun 2014 20:04 - Henrique Fernandes

Status:	Closed	**Start date:**	05 Jun 2014
Priority:	Normal	**Due date:**	
Assignee:	Abhinav Badola	**% Done:**	100%
Category:	video	**Estimated time:**	0.00 hour
Target version:	0.25		

Description

Sorry i might don't be much help but here
https://bugs.kde.org/show_bug.cgi?id=335816

When i try to open digikam it crashes, when i filled the bug they told me the problem might be with exiv2

So here i am!

I am willing to help!

I can separate all my video files to see with one causes the bug if it realy is!

Thanks

Related issues:

Related to Exiv2 - Bug #1091: exiv2 segfaults on Matroska video in Exiv2::Mat...	**New**	**07 Jun 2015**

Associated revisions

Revision 3264 - 19 Jun 2014 13:28 - Abhinav Badola

#960: Added a Buffer Overflow Fix in INFO tags of RIFFVIDEO.CPP

Revision 3265 - 19 Jun 2014 13:43 - Abhinav Badola

#960: Fixed a small bug found by coverity scan results

Revision 3912 - 28 Aug 2015 19:57 - Robin Mills

#960 added API: static void Exiv2::XMPParser::getRegisteredNamespaces(std::map<std::string,std::string>&);

History

#1 - 08 Jun 2014 08:28 - Henrique Fernandes

- File 18.png added

I found the file, i can't upload it here. (27MB)

I am uploading it to drive.
Here the link to the file:

https://drive.google.com/file/d/0B5CHbb3RVwicYXNjbW54VERmTG8/edit?usp=sharing

Here s what exiv2 outputs when i try to run:

```
$ exiv2 bugado/12/31/100_0182.AVI
*** Error in `exiv2': malloc(): memory corruption (fast): 0x0000000001a260e0 ***
```

And the terminal gets stuck. I have to Ctrl + C to get out.

exiv2 info:

```
[sfrique@arch-desktop Videos]$ exiv2 -Vv
exiv2 0.24 001800 (64 bit build)
Copyright (C) 2004-2013 Andreas Huggel.
```

```
This program is free software; you can redistribute it and/or
modify it under the terms of the GNU General Public License
as published by the Free Software Foundation; either version 2
of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public
License along with this program; if not, write to the Free
Software Foundation, Inc., 51 Franklin Street, Fifth Floor,
Boston, MA 02110-1301 USA
exiv2=0.24.0
platform=linux
compiler=G++
bits=64
dll=1
debug=0
version=4.8.2 20131219 (prerelease)
date=Jan 24 2014
time=11:25:52
executable=/usr/bin/exiv2
library=/usr/lib/libexiv2.so.13
library=/usr/lib/libstdc++.so.6
library=/usr/lib/libgcc_s.so.1
library=/usr/lib/libc.so.6
library=/usr/lib/libdl.so.2
library=/usr/lib/libz.so.1
library=/usr/lib/libexpat.so.1
library=/usr/lib/libm.so.6
library=/lib64/ld-linux-x86-64.so.2
```

Computer info:

```
[sfrique@arch-desktop ~]$ uname -a
Linux arch-desktop 3.14.5-1-ARCH #1 SMP PREEMPT Sun Jun 1 07:36:23 CEST 2014 x86_64 GNU/Linux
[sfrique@arch-desktop ~]$ cat /etc/os-release
NAME="Arch Linux"
ID=arch
PRETTY_NAME="Arch Linux"
ANSI_COLOR="0;36"
HOME_URL="https://www.archlinux.org/"
SUPPORT_URL="https://bbs.archlinux.org/"
BUG_REPORT_URL="https://bugs.archlinux.org/"
```

Here teh info i got from smplayer:

```
100_0182.AVI
Geral
Arquivo
/RAID/sfrique/Videos/bugado/12/31/100_0182.AVI
Tamanho
26848 KB (26 MB)
Duração
00:01:41
Demuxer
avi

Informação do clip
Software
EASTMAN KODAK COMPANY KODAK EASYSHARE Sport Camera, C123

Vídeo
Resolução
640 x 480
Tamanho do Vídeo
1.33333
Formato
MJPG
Taxa de bits
2021 kbps
Quadros por segundo
```

```
30.039
Codificador selecionado
ffmjpeg

Transmissão Inicial de Áudio
Formato
7
Taxa de bits
128 kbps
Taxa
8000 Hz
Canais
2
Codificador selecionado
ulaw

Transmissões Áudio
#
Idioma
Nome
ID
0
<vazio>
<vazio>
1
```

#2 - 08 Jun 2014 11:59 - Robin Mills

- Status changed from New to Assigned

- Assignee set to Robin Mills

- Target version set to 0.25

Henrique

Thanks for providing the test file and the analysis. I've reproduced this on Mac/Windows/Linux. Here's the Mac Terminal output

```
577 rmills@rmills-mbp:~/gnu/exiv2/bugs $ exiv2 100_0182.AVI
*** exiv2(11723,0x7fff77ba9310) malloc: ***
error for object 0x7fdd63e01858: incorrect checksum for freed object
- object was probably modified after being freed.
set a breakpoint in malloc_error_break to debug
Abort trap: 6
578 rmills@rmills-mbp:~/gnu/exiv2/bugs $
```

Curiously, when I rerun it on the Mac, it's OK.

```
580 rmills@rmills-mbp:~/gnu/exiv2/bugs $ exiv2 100_0182.AVI
File name       : 100_0182.AVI
File size       : 27492808 Bytes
MIME type       : video/riff
Image size      : 0 x 0
100_0182.AVI: No Exif data found in the file
581 rmills@rmills-mbp:~/gnu/exiv2/bugs $
```

#3 - 08 Jun 2014 12:00 - Robin Mills

- Category set to video

#4 - 17 Jun 2014 16:42 - Robin Mills

- Assignee changed from Robin Mills to Abhinav Badola

Abhinav

I think this is being caused by localization of Xmp keys. riffvideo.cpp calls exvGettext() about line# 880.

I've changed this to:

```
    xmpData_[exvGettext(tv->label_)] = buf.pData_;
```

to

```
    xmpData_[tv->label_] = buf.pData_;
```

I'm not certain, however I suspect that exvGettext() is not thread safe and returning a char* to a static buffer. Sometimes the buffer is OK, and sometimes he has been freed when XmpData[key] = value is processed. Of even worse, exvGettext() doesn't handle unknown keys gracefully and returns random pointers for unknown input strings.

I'm not sure why the key is being passed through exvGettext() at all. If it's essential to use exvGettext, it might be a good idea to have an additional API:

```
  std::string& exvGettext(const char*, std::string&);
```

This additional exvGettext() would update the string passed to him and return a reference to the passed string. A lot safer than a static buffer. You'd call it like this:

```
    String key;
    xmpData[exvGettext(tv->label_,key)]=buf.pData;
```

There about 60 calls to exvGettext() in the video code. Before I change them all, I'd appreciate your review and opinion. Please reassign the issue back to me with your comments.

Robin

#5 - 17 Jun 2014 17:01 - Robin Mills

I'm even more confident that it's exvGettext() that is causing this. When I configure and build with --disable-nls, the problem goes away. When NLS is disabled, exvGettext() is compiled as:

```
const char* exvGettext(const char* key)
{
    return key;
}
```

No more mysterious calls into dgettext. Being a native English speaker, I don't have a strong feel for localization. However I don't see a need to localize XMP keys.

#6 - 19 Jun 2014 13:22 - Abhinav Badola

Hi Robin, Henrique,

Robin Mills wrote:

> I'm even more confident that it's exvGettext() that is causing this. When I configure and build with --disable-nls, the problem goes away. When NLS is disabled, exvGettext() is compiled as:
> [...]
> No more mysterious calls into dgettext. Being a native English speaker, I don't have a strong feel for localization. However I don't see a need to localize XMP keys.

I took the sample file and was able to reproduce the bug on my Linux Machine as well.
I tried to do away with all the exvGettext(), but it didn't solve the problem for me.

So I took the problem a little deeper and saw what was being fetched in the XMP container.

The program was getting stuck because the data to be read was too large for the buffer allocated. The default size of the buffer was 100, and the data that was to be read was 8192 in size.

```
infoSize :8310   8310
infoTagsHandler :DTIM
infoSize :20
before :20
after :20Data : 2012:01:01 00:07:14Size : 8278
true for label Xmp.video.DateTimeOriginal
Info Tag Xmp.video.DateTimeOriginal       infoData :2012:01:01 00:07:14
infoTagsHandler :IKEY
infoSize :8192
before :8192
after :8192Data : Size : 78
true for label Xmp.video.PerformerKeywords
Info Tag Xmp.video.PerformerKeywords       infoData :
infoTagsHandler :ISFT
```

```
infoSize :60
before :60
after :60Data : EASTMAN KODAK COMPANY KODAK EASYSHARE Sport Camera, C123Size : 10
true for label Xmp.video.Software
Info Tag Xmp.video.Software       infoData :EASTMAN KODAK COMPANY KODAK EASYSHARE Sport Camera, C123
infoTagsHandler :SRAT
infoSize :2
before :2
after :2Data : Size : 0Exit and runReached the DECODE BLOCK
JUNK
Reached the DECODE BLOCK
LIST
Reached the DECODE BLOCK
movi
```

When I looked into the data of the file closely, it was a bit clear where the potential problem was lying.
On looking at the data of the TAG - IKEY, it seemed strange that the data was out of range.

```
infoTagsHandler :IKEY
infoSize :8192
before :8192
after :8192Data : Size : 78
true for label Xmp.video.PerformerKeywords
Info Tag Xmp.video.PerformerKeywords       infoData :
```

Generally it is supposed to have a few words as information, but the size of the TAG in this file was exceptionally large, and at the same time, empty.

For now I would be inserting this buffer overflow fix.

@Robin,
You are absolutely right about fixing the issue with exvGettext(), as it may not be always safe to call it.

Please do tell me what is your opinion on this issue as of now. Does it seem fixed..?

I think it would be best if I change the code in the places where exvGettext() has been used, for the following reasons -
[a] It is mostly used in Video code which has been written by me.
[b] I have a lot of video files at my home PC which I can use to test and verify that all changes have been done without breaking the previous working of the code.

#7 - 20 Jun 2014 06:32 - Henrique Fernandes

Hello,

If needed i can try to use the code in multiple videos files as well. That is the most i can do, not much help with the code!

I did run exiv2 in all my videos files before to find this problem.

#8 - 20 Jun 2014 11:03 - Robin Mills

Henrique and Abinhav

Thank You both for your work on this. We're close to resolving this.

@Henrique
There were bugs reported with DigiKam <-> Exiv2 crashing (#961 and #963). These were in the exception handler in libkexiv2 (the sandwich between DigiKam and Exiv2). That part of the code has been made more robust. So, it's been fixed and DigiKam won't crash here any more. The fix isn't part of Exiv2. I've no idea how/when it willl appear in DigiKam.

However, it's only the crash that has been fixed. You will still not be able to open this file. I think we'll resolve that puzzle in the next few days.

@Abhinav
Thanks very much for offering to undertake the fix. I think you've discovered the real issue here. There is a buffer overflow. For sure you know more about this code. It sounds as though the overflow is being caused by the file containing erroneous data and 8000 bytes are traveling a road designed for less that 100 bytes. I encourage to find a fix which prevents the overflow by chopping the data (say after 90 bytes), or throw an Exiv2 exception.

I think my concerns about exvGettext() are probably a lucky, but bad, guess and do not deserve further investigation/attention at the moment.

@Henrique and Abhinav
I think it's very helpful that you have loads of test files. When we get this fixed, please run every test you can. You cannot perform "too much testing!".

Robin

#9 - 30 Nov 2014 20:59 - Abhinav Badola

Hi Henrique,
Shall we close this bug now..??

Can you please confirm if this got fixed at your end as well..?

#10 - 30 Nov 2014 21:41 - Henrique Fernandes

Hi Abhinav Badola,

Sorry but the error persists for me

```
*** Error in `digikam': malloc(): memory corruption (fast): 0x00007ffcb802f370 ***
<pre>

And i keep geting weird errors.

Problem is i am not using the latest digikam!

I still with 4.2.

Rigth now i can't do more debuging or testing... i don't even have time to fix my just broken digikam again.

Sorry =/
```

#11 - 04 Dec 2014 04:38 - Abhinav Badola

Hi Henrique,

Are you building digiKam from source..?
Or are you using some distribution of digiKam..?

#12 - 04 Dec 2014 13:28 - Henrique Fernandes

Hello,

I am using it from distro.
Manjaro or Arch

I think manjaro update it to lastest. So i might be able ot test it later today!
But at the end of the week i will test with digikam 4.5

#13 - 05 Dec 2014 04:19 - Abhinav Badola

Hi Henrique,

I would like to inform you that this bug won't get fixed even after you update digiKam to the latest.

The fix is currently in our trunk (i.e. main development svn repository).
If you install digiKam with default packages, it will use the older version of Exiv2, i.e. 0.24

The fix to your specific bug will be shipped in the next version of our library.(0.25)

To fix digiKam for the time being, you will have to manually install the latest version of Exiv2 from the svn repository.

It is pretty easy to install Exiv2 as it only depends on very few C++ libraries, and most of them are part of Standard C++ Libraries for development.
The below links can help you get started with installing the latest version of Exiv2.
[1] http://www.exiv2.org/download.html
[2] http://dev.exiv2.org/projects/exiv2/wiki/How_do_I_build_Exiv2_on_the_XYZ_platform

Henrique Fernandes wrote:

> Hello,
>
> I am using it from distro.
> Manjaro or Arch
>
> I think manjaro update it to lastest. So i might be able ot test it later today!
> But at the end of the week i will test with digikam 4.5

#14 - 05 Dec 2014 15:41 - Henrique Fernandes

Hi Abhinav,

Thanks for the info, but one reason that i am usign arch now if to get up to date packages from repository.
When in ubuntu i always had to updatd packages by hand and for security fix and etc this way doesn't work well.

But still, i am glad to see it was fixed. I will simple remove this video file from my library, later on i can put it in again!

So if you know 0.25 fixed the issue you can close the ticket. The only video i have that gives me any errors is the one i uploaded here!

Thanks!

#15 - 02 Jun 2015 23:31 - Alan Pater

- Status changed from Assigned to Resolved

- % Done changed from 0 to 100

#16 - 21 Jun 2015 16:41 - Andreas Huggel

- Status changed from Resolved to Closed

Files

18.png	105 KB	08 Jun 2014	Henrique Fernandes