Exiv2 v0.24.1

KDE Security Issue

Project Status

- v0.23 released April 2012
  Canon CR2. Pentax DNG. Bug Fixes
- v0.24 released December 2013
  Includes GSoC2012 video-read code. Bug Fixes
- v0.25 code complete December 2014
  Many bug fixes, function changes and new features
  Many maker-note updates
  Option to build GSoC2013 cloud ready code (http etc)
  GSoC2013 video-write code deferred to v0.26
- v0.26 over the horizon
  GSoC2013 video-write
  GSoC2015 webp/webm
  XMPsdk
  Accessory ‘overflow’ file

Meeting Objective

- To decide how to address KDE's Security Issue
- *** No other meeting objective ***
- Duration: 1 hour maximum
- No finger pointing, blaming or complaining
- Other useful discussion points if time permits
  Introduce new team members
  Discuss v0.25
  Discuss v0.26
  Discuss the Exiv2 Logo Project

KDE Security Issue

makosoft@gmail.com

Someone filed a Debian bug report for the first one a few weeks back and I expect they'll find the others shortly if they haven't already: https://bugs.debian.org/cgi-bin/bugreport.cgi?bug=781123 (This was "fixed" a while back by increasing the buffer size, which didn't actually fix the security issue.) Even though all of these are video metadata related and KDE only uses exiv2 for image files, they still affect KDE because if you rename a video file to .jpg, KDE thinks it's an image and uses exiv2 to get metadata from it.

The others are a heap buffer overflow in nikonTagsParser here:

```
io_->read(buf.pData_, 2);
tagID = Exiv2::getULong(buf.pData_, littleEndian);
io_->read(buf.pData_, 2);
dataSize = Exiv2::getULong(buf.pData_, littleEndian);
temp -= (4 + dataSize);
```

A stack buffer overflow (underflow?) here:

```
while(dataSize) {
        std::memset(buf.pData_, 0x0, buf.size_);
        io_->read(buf.pData_, 1);
        str[(4 - dataSize) * 2] = (char)(Exiv2::getULong(buf.pData_, littleEndian) + 48);
        --dataSize;
}
```

This pair is particularly nasty because the stack buffer overflow allows the address of the heap buffer to be partially overwritten, making it possible to write to other areas of memory, and because the stack is executable in certain libexiv2-based applications. With ASLR disabled, I can get full arbitrary code execution by using this to overwrite a vtable pointer, and I reckon it should be possible to exploit this on real systems with ASLR on (it's just really fiddly to do -

I believe there's another heap buffer overflow in the AsfVideo::codecList here:

```
io_->read(buf.pData_, 2);
descLength = Exiv2::getUShort(buf.pData_, littleEndian) * 2;
io_->read(buf.pData_, descLength);
```

Current Status

- Robin has offered that Exiv2 will produce a dot release
- This meeting should decide on our actions:
  1. Who will ‘own’ this matter
  2. Who will conduct the discussion with KDE
  3. Our options to address this matter
  4. Assignments to Team Members
  5. Schedule
- Caution:
  Until we have a strong line of communication with KDE, assignments and schedule are preliminary

Analysis of Security Issue

- We must isolate the video code as a build option (as we done with webready for v0.25)
- The video code requires to be “hardened” to deal with suspect files
- The video code requires simplification
  - refactoring into functions
  - constants defined with enums in headers
  - code review recommendations to be implemented
  - more and tougher tests (5Gbyte video files over http)

Options for v0.24.1

- The video code should be either:
  1) declared “experimental” and not built by default
  2) removed and reintroduced in v0.26 (or later)

- Start with v0.24
  - do as little work as possible to isolate the video code
  - add selected trunk fixes (such as maker notes)
  This can be done quickly if the selection is small
  Time Guestimate: 2 weeks

- Release v0.25
  - isolate the video-read code
  This cannot be done quickly. v0.25 has a lot of new code (cloud-ready) and many bug fixes and new features
  Time Guestimate: Unknown

Exiv2 Project Status

- exiv2.org Redmine Forum/Wiki/Issues
  - lots of user activity
  - we are very responsive
- The code is widely used
- exiv2.dyndns.org:8080 Jenkins build server works well
- Our release interval is getting longer
- The team is growing. 3 new hires in 2015
  Thomas S Thomas B Alan
- Islam's GSoC 2015 webp/webm project
- Robin is overloaded

Exiv2 v0.26

- Video Code (read and write)
- webp/webm support (GSoC 2015)
- Accessory ‘overflow’ file (Thomas B)
- More features and bug fixes
- More platforms supported (Android and iOS)
- Upgrade code to link XMPsdk as an external library
- Better Program Management
- Improved infra-structure
- Team members should ‘own’ responsibilities

The Future of Exiv2

- Exiv2 is a great open source project
  - Useful
  - Highly functional
- Most C++ code and test harness is excellent thanks to:
  - Andreas Brad
  - Gilles Volker
- Excellent and fast updates for cameras and accessories:
  - Niels Thomas B (v0.26)
- Our build and user support is excellent thanks to:
  - Robin Alan
- Our team work can be improved thanks to:
  - Everybody: Me, you, users, contributors