

---

Subject: Linking standard libraries

Posted by [dolik.rce](#) on Sun, 29 Jan 2012 21:03:20 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Hi everyone,

Is there any important reason why we use our own copy of some common standard libraries, such as libjpeg or libbz2 on platforms where they are almost always present as shared libs? I understand that having the copy of code is useful for windows, but in POSIX platforms we can just link against the shared libs, as it is done already in some plugins, e.g. in plugin/png.

I've noticed this while checking the quality of debian packages with lintian tool. It considers using libjpeg and libbz2 statically linked in to be an error. Here is the explanation, for completeness:lintian -i theide\_4424-0~squeeze0\_amd64.debE: theide: embedded-libjpeg  
./usr/bin/theide

N:

N: The given ELF object appears to have been statically linked to libjpeg.

N: Doing this is strongly discouraged due to the extra work needed by the

N: security team to fix all the extra embedded copies or trigger the

N: package rebuilds, as appropriate.

N:

N: If the package uses a modified version of libjpeg it is highly

N: recommended to coordinate with the libjpeg maintainer to include the

N: changes on the system version of the library.

N:

N: Refer to Debian Policy Manual section 4.13 (Convenience copies of code)

N: for details.

N:

N: Severity: serious, Certainty: possible

N:

E: theide: embedded-library ./usr/bin/theide: bzip2

N:

N: The given ELF object appears to have been statically linked to a

N: library. Doing this is strongly discouraged due to the extra work needed

N: by the security team to fix all the extra embedded copies or trigger the

N: package rebuilds, as appropriate.

N:

N: If the package uses a modified version of the given library it is highly

N: recommended to coordinate with the library's maintainer to include the

N: changes on the system version of the library.

N:

N: Refer to Debian Policy Manual section 4.13 (Convenience copies of code)

N: for details.

N:

N: Severity: serious, Certainty: possible

N:

The changes required to get rid of these errors and to make U++ more standards compliant

should be quite minimal. For starters, I attach patch for plugin/bz2. Converting plugin/jpg might be bit trickier, but I'll try to do it soon as well.

I'm just not sure if there is any real reason that would prevent merging these patches to U++?

Best regards,  
Honza

#### File Attachments

---

1) [bz2.patch](#), downloaded 380 times

---