1 Introduction

This document describes some guidelines for people participating

in lwIP development.

2 How to contribute to lwIP

Here is a short list of suggestions to anybody working with lwIP and

trying to contribute bug reports, fixes, enhancements, platform ports etc.

First of all as you may already know lwIP is a volunteer project so feedback

to fixes or questions might often come late. Hopefully the bug and patch tracking

features of Savannah help us not lose users' input.

2.1 Source code style:

1. do not use tabs.

2. indentation is two spaces per level (i.e. per tab).

3. end debug messages with a trailing newline (\n).

4. one space between keyword and opening bracket.

5. no space between function and opening bracket.

6. one space and no newline before opening curly braces of a block.

7. closing curly brace on a single line.

8. spaces surrounding assignment and comparisons.

9. don't initialize static and/or global variables to zero, the compiler takes care of that.

10. use current source code style as further reference.

2.2 Source code documentation style:

1. JavaDoc compliant and Doxygen compatible.

2. Function documentation above functions in .c files, not .h files.

(This forces you to synchronize documentation and implementation.)

3. Use current documentation style as further reference.

2.3 Bug reports and patches:

1. Make sure you are reporting bugs or send patches against the latest

sources. (From the latest release and/or the current CVS sources.)

2. If you think you found a bug make sure it's not already filed in the

bugtracker at Savannah.

3. If you have a fix put the patch on Savannah. If it is a patch that affects

both core and arch specific stuff please separate them so that the core can

be applied separately while leaving the other patch 'open'. The prefered way

is to NOT touch archs you can't test and let maintainers take care of them.

This is a good way to see if they are used at all - the same goes for unix

netifs except tapif.

4. Do not file a bug and post a fix to it to the patch area. Either a bug report

or a patch will be enough.

If you correct an existing bug then attach the patch to the bug rather than creating a new entry in the patch area.

5. Trivial patches (compiler warning, indentation and spelling fixes or anything obvious which takes a line or two)

can go to the lwip-users list. This is still the fastest way of interaction and the list is not so crowded

as to allow for loss of fixes. Putting bugs on Savannah and subsequently closing them is too much an overhead

for reporting a compiler warning fix.

6. Patches should be specific to a single change or to related changes.Do not mix bugfixes with spelling and other

trivial fixes unless the bugfix is trivial too.Do not reorganize code and rename identifiers in the same patch you

change behaviour if not necessary.A patch is easier to read and understand if it's to the point and short than

if it's not to the point and long :) so the chances for it to be applied are greater.

2.4 Platform porters:

1. If you have ported lwIP to a platform (an OS, a uC/processor or a combination of these) and

you think it could benefit others[1] you might want discuss this on the mailing list. You

can also ask for CVS access to submit and maintain your port in the contrib CVS module.