Planet SoC

Week 4 Updates: Moving

Sun, 06/24/2007 - 03:11 — jeff

This week I moved and started a new job I'll have over the summer, in addition to Google Summer of Code. Which may seem like a lot of work to be committed to, but since the birth of my MySpaceIM protocol plugin for Pidgin on April 7th I've been working full time (except for a short time off between jobs) and made good progress, so I'm too not concerned about not having enough time to work on msimprpl. It is good to be busy. This week was hectic because of getting settled in at my new place and doing things like setting up Internet (which, unfortunately is not very reliable here) but I did manage to work on a several important parts of msimprpl.

I updated the msimprpl branch to be in sync with the Pidgin v2.0.2 release. No changes in 2.0.2 conflicted with my changes in msimprpl.

I setup a mingw32 cross-compiler on Linux to build a Windows DLL of msimprpl (I've been planning to do this for a while now). I wrote a simple shell script to automate packaging a source release and Win32 binary release for msimprpl, since it is currently being developed and released out of line with the main Pidgin development. Now I won't have to reboot into Windows to produce Windows builds of msimprpl (important especially now, since I currently don't have a Windows installation to test on--but I'm working on that.) Msimprpl 0.7.1. is the first release built from a mingw cross-compiler.

I've been asked about binary releases of msimprpl for Linux, but really having releases of msimprpl separate from Pidgin is only an interim solution, until (if possible) msimprpl is integrated within Pidgin.

As for coding, I refactored two related functions, refactor msim_escape() & msim_unescape(), so that they now read perform replacements based on the identical information. That is, I can change two parallel arrays in one part of the code, and both functions will recognize the new escape sequences. This is important if new escape sequences are discovered (besides /1 and /2 for slash and backslash) and also from a maintainability standpoint. To make sure the new functions worked, I wrote a small test suite, msim_test_all(), that ensures the functions basically work as expected.

msim_unrecognized() is now called for unknown messages, dumping the message to the debug log.

I confirmed that offline instant messages work as expected, just the same as online instant messages. You can IM someone that is offline, and they'll receive it once they sign on.

I've completed quite a few small TODOs in the source code. Today, I'm down to 31 in myspace.c, 13 in message.c, and 5 in myspace.h (some duplicates).

"Get Info" now works on all users, even those not on your buddy list. It does this by querying the user info, using msim_lookup_user(), rather than reading cached information from the buddy list. When a reply is received, the profile information will be displayed.

I'm just about ready to have msimprpl released for more widespread testing, and am currently looking into ways to do

this. Stay tuned.

Tags: msimprpl Pidgin
Source: jeff's blog
Login or register to post comments

Comments

Using MSIM in Windows!

Tue, 06/26/2007 - 00:01 — Anonymous

Hey there, I'm using your plugin on the Windows client, so I can help you with testing if needed.

From your TODO list:

Find out why passwords are limited to 8 characters. Is there a mistake in msim*compute*login_challenge, or does the official client have this limitation too?

MySpace limits you to a 10-character password. I noticed this when attempting to use a longer than 10-character password in the client, this actually returned a error that crashed Pidgin. I can probably provide logs for this crash if you can tell me where they are and need them,…

I'll start using it more tomorrow and see if I can flush out some more of your TODO items for you.

Login or register to post comments