




Planet SoC

[Home](#) > [Blogs](#) > [jeff's blog](#)

Coding Officially Begins Soon

Thu, 05/17/2007 - 05:08 — [jeff](#)

So the coding portion of Summer of Code begins on May 28th (you can add a Google Calendar I made of SoC events here:

). Google has built in additional time this year to have "Students learn more about their project communities", so I thought I'd share what I've been doing in this time, in preparation for coding the [MySpaceIM protocol plugin for Pidgin](#).

I branched the `im.pidgin.pidgin` Monotone branch into my own personal branch, `im.pidgin.soc.2007.msimprpl`. As of this writing there are also `im.pidgin.soc.2007.monoloader`, `im.pidgin.soc.2007.remotelogging`, and `im.pidgin.soc.2007.telepathy` branches for other SoC students this year (I look forward to their contributions). I created my branch before Pidgin 2.0.0 and began working on my plugin independently from changes made by other Pidgin developers, so I wasn't affected by their work-in-progress code. After Pidgin 2.0.0 was released, I merged their changes back into my branch using `mtn propagate`. Since my plugin is fairly self-contained, there were few conflicts, but I did have to manually merge `libpurple/cipher.c`.

[Nathan Peterson](#) made some progress in independently reverse-engineering the MySpaceIM protocol, at about the same time as me. Before he stopped, Nathan has made more progress reversing the protocol and I have made more progress implementing the Pidgin plugin, so our efforts complemented each other nicely (although we both worked independently until I was accepted for GSoC). I have since merged his documentation into [my specification](#).

The MySpaceIM protocol uses the [RC4 stream cipher](#) for logging in. While I had implemented RC4 in my plugin using GPL'd RC4 code from the Samba project, Nathan implemented RC4 into Libpurple's Cipher API—a much more elegant solution. I adapted my protocol to use RC4 in Libpurple and applied the patch to my branch. In propagating the changes from the main `im.pidgin.pidgin` branch, I used [meld](#) to perform a 3-way merge between a) the pre-2.0.0 `cipher.c` b) 2.0.0's `cipher.c` and c) the patched `cipher.c` with RC4. Meld made this very easy and I committed a merged `cipher.c` with RC4 support and the latest Libpurple 2.0.0 changes in no time.

It soon became apparent that the official Libpurple cipher library should have RC4 as well, so that a special libpurple is not required. To accomplish this, I used `mtn pluck` to take the changes I just described that I applied to my `im.pidgin.soc.2007.msimprpl` branch, and apply them to the main `im.pidgin.pidgin` branch. Now the next release of Libpurple will have RC4 support. Trac helpfully displays the [tree of revisions](#) to show you in what context the changes were applied. It is very refreshing to work with a modern version control system. I don't know how some developers can live with archaic systems like PVCS.

I'm becoming more comfortable with Trac. I committed [my first patch](#) to the official `im.pidgin.pidgin` branch — a small documentation fix, but it's a start. I and a handful of others submitted bugs for my alpha-quality plugin. Tickets were closed, tickets were opened. Richard Laager has been quite helpful in helping me get my code up to standards, but of course I still have a ways to go before `msimprpl` is ready for general usage.

I bought the 20th-anniversary Edition of *Gödel, Escher, Bach: an Eternal Golden Braid* by Douglas R. Hofstadter. This book has been [selected](#) for the GSoC '07 Book Club and I've been meaning to read it forever. In reading the first three chapters, I'm already hooked. This is going to be an interesting summer.

Tags: [cipher](#) [dvcs](#) [monotone](#) [msimprpl](#) [mtn](#) [myspace](#) [myspaceim](#) [Pidgin](#) [pidgin](#) [rc4](#)

Source: [jeff's blog](#)

[Login](#) or [register](#) to post comments
