

Help RichM Secure Enterprise IM

del.icio.us

Discuss in Forums {mos_smf_discuss:RichM}

Chat is pervasive throughout our enterprise. Its use is an integral part of how most businesses get work done. It is understandable considering how much faster it is than talking on the phone or even sending an email. The ability to exchange information in real-time, cutting right to the chase, is what makes instant messaging (IM) so enticing to corporate society. The problem is that users are just as susceptible if not more so to malicious attacks, since they have not been educated as to the dangers that IM can bring. This is why I chose to eliminate my dependence on outside entities for my organizations internal chatting needs.

Each month, I do my best to find an interesting topic to focus on or a new project that I can share. I sought out a way that could eliminate third party chat clients, namely AOL Instant Messenger (AIM) and outside servers that are not in our control. I also sought to do this all while making the project cost effective and more secure. That was the intent, anyhow, but sadly I fell short. And this is where I need your help.

The Current Problem

AIM is a solid client used by millions daily, but it is insecure and a popular target for opportunistic hackers. Also, AIM tends to be a bit of a resource hound, loading at startup by default, and inundating users with movie trailers, and other such process intensive nuisances. AIM is not the only client that has vulnerabilities. In fact instant messaging makes the SANS Top 20 vulnerabilities of 2006. I wanted to create something that allowed me to control who would and would not be able to chat with my users, eliminating chances of a malicious payload upfront.

Initial Research

After some quick preliminary searching, I found an open source client, Gaim which supports MSN, AIM, Yahoo (among others), and it also supports a client that works with an open source protocol, jabber. Jabber is an open source server that allows admins to encrypt traffic on the Jabber server, maintain membership (through username and passwords) and most importantly use the Jabber client available through Gaim natively.

I thought I had struck gold. It was too obvious. First I get my users to adjust to AIM through Gaim, then, once they were comfortable, announce a new enhanced in-house server that would fulfill all our needs. This would also help to make communication confidential and secure, thus weaning them off AIM and onto Jabber using the same client (Gaim), which they have recently become accustomed to. There was only one problem. Jabber is a bit of a mystery when it comes to installation, configuration and usage and support is a somewhat non-existent.

There are some great resources available such as ejabberd, but unfortunately they are not maintained. There have been tremendous advances in linux kernels and the packages they support. I searched for almost a solid week and could not find one guide, book, or tutorial written after 2004. I set about trying to install Jabber and get it to function properly. I was able to follow the instructions and modify package names (with the latest releases) but, after trying to connect nothing I did seemed to work.

I found another organization Process-one. They have a fantastic Jabber client, but since it is designed to run in a GUI environment, that is not the solution for me. When it comes to a linux box, the characteristic I most appreciate is how secure they are, because they lack a GUI and the vulnerable 1,000s of lines of code that come along with it. I searched various forums, but there wasn't any real recent activity to speak of. That is when I decided I could not tackle this on my own.

Call To Action

I am asking for anyone interested to join me in a quest to create current (up to date) documentation for a solid TUI based Jabber server implementation. I am willing to use any distro and any Jabber protocol (I tried ejabberd, since they had the most documentation). Is there anyone out there interested in tackling this? The good people at ejabberd have done most of the legwork (<http://ejabberd.jabber.ru/tuto-install-ejabberd>). We just need to fill in some of the blanks and locate the equivalent packages for the distro we have chosen.

I would say that we should stay with the major players: Madriva, Ubuntu, Fedora Core, and Debian. This was an unfortunate setback that occurred, and it would be nice to right this wrong and help to contribute to the infosec community, something that is so vital and will only become more important as time passes. I will be responsible for all the documentation and submissions to the Ethical Hacker Network. So if you have a few hours to spare and would like to participate, please PM me, or post in the forum under this topic thread.

No IM Graphic from Secure Computing.