Encrypting Instant Messaging Conversations [Underground Security Paper #1]

By: DIzzIE [c]opyleft 2006

Whenever you talk online with your instant messaging (IM) client of choice, your conversations can be, and in all probability *are*, recorded, monitored, and read. Any data which travels over a network can be viewed using programs known as packet sniffers, with some specially crafted programs, such as IM Sniffer or AIM Sniff, designed exclusively to capture IM communications. No matter how pathetically dull your treacherous life is, chances are someone is bored enough to fuck with it.

What will soon follow is a list of various programs and plugins which you can use with most standard IM clients to encrypt your conversations. The focus will be predominantly on Windows systems (though the tools discussed are often available for other platforms as well), and will also only cover free (as in beer) software, as there?s no need to pay when there are plenty of gratis alternatives (if, however, a time does come when the below mentioned tools stop being free, there?s a textfile on finding serial numbers here: www.dizzy.ws/serials.htm).

Nota Bene: *Always* encrypt your conversations (even the seemingly innocuous ones) and *always* keep regenerating (changing) your encryption keys. The reason for the former is that, unless you are intentionally spreading disinformation which you plan on the sniffers seeing, whatever data you consider to be unimportant can be used to compile a profile of you and your activities, which can in turn be used to gain insight into life habits, password choices or those fun password reminder questions, and so forth. The reason for the latter is that the longer you use the same key to encrypt your conversations, the more data and time the attacker has to spend on trying to crack your encryption. Change your key once every hour, every day, every week, or every chat session. The choice is yours, just remember that the longer you use the same key, the more vulnerable you become. Regenerating your key is also easier to do with some of the programs below than with others, while some even regenerate the key for you.

Now then, with no particular order in mind, on with the list!

Name: **Gaim-Encryption** (http://gaim-encryption.sourceforge.net/)

Key Strength: 512 to 4096 bit RSA keys.

Works With: Gaim (http://gaim.sourceforge.net/)

Operating Systems Supported: Windows/*nix

Protocols Supported: AIM, Jabber, ICQ, **[unconfirmed]**, YIM **[unconfirmed]**, MSN **[unconfirmed]**, Gadu-Gadu **[unconfirmed]**, GroupWise **[unconfirmed]**, Napster **[unconfirmed]**, SILC **[unconfirmed]**, IRC **[not**

supported (while Gaim does act as a primitive IRC client, the Gaim-Encryption plugin **does not work** with Gaim IRC, see below for IRC encryption options)**]**

Installation Example: Download and install Gaim. Download the Gaim-Encryption plugin and run the installer. Run Gaim. Click on Preferences and go down to Plugins on the left-hand side. Find ?Gaim-Encryption? listed on the right, and check the accompanying checkbox. Restart Gaim. Go back to Preferences, and this time you should see ?Gaim-Encryption? listed under Plugins on the left-hand side. Select ?Gaim-Encryption? and in the Config tab on the right make sure that ?accept conflicting keys automatically? is unchecked, and ?automatically encrypt if buddy has plugin?, ?broadcast encryption capability? are both checked. Checking the remaining ?accept key automatically if no key on file? box is optional.

Next, click on the Local Keys tab and select your key. If you don?t see any keys listed there, you will first need to start an encrypted conversation with someone else who is using the Gaim-Encryption plugin. Once the conversation has been started, go back to the Local Keys tab and select your key. Click on Regenerate Key and in the Generate Keys pop-up type in 4096 (the maximum key strength the GE plugin supports at the time of this writing) instead of the 1024 value listed in the Key Size field, and hit OK. On slower machines it will appear as if Gaim has frozen on the ?generating RSA key pair?? screen. This is normal, and therefore you should not attempt to restart Gaim, just give it a few minutes. The person with whom you first initiated the conversation should also be regenerating zir key. Once your key has been successfully regenerated, click on the Trusted Buddy Keys and the Recent Buddy Keys tabs and delete the existing 1024 bit keys from your list.

Finally, restart Gaim and reinitiate your conversation. Both the Tx and Rx locks in the IM window should now be red (you may also see a confirmation dialogue pop up, which asks whether you want to accept the key once or accept it and save it, or reject it. Ideally, you should Accept Once). Now go back to the Recent/Trusted Buddy Keys tabs and make sure that the key now stored there for your chat partner is 4096 bits.

Assuming you possess a secure email account and/or secure phone line, you should contact each other and confirm the Key Fingerprint to help ascertain the identity of your chat partner, and then hit Close to exit out of the Preferences menu. You should now be ready to engage in secure conversations. Note: if when messaging your chat partner the locks in the IM window do not turn red, make sure you both have the ?automatically encrypt if buddy has plugin? and ?broadcast encryption capability? options checked in the Config tab, and try clicking on the lock icons.

Name: **Off-the-Record (OTR) Messaging** (http://www.cypherpunks.ca/otr/)

Key Strength: ??? (some sort of Diffe-Hellman protocol?) [The description of the OTR protocol is available here: http://www.cypherpunks.ca/otr/Protocol-v2-3.0.0.html. It is complex and convoluted, so I was unable to figure out what the key strength is, if you do, however, then let me know!]

Works With: Gaim, Adium, Miranda IM **[unconfirmed]**, iChat **[unconfirmed]**, Trillian **[unconfirmed]**, vanilla AIM client **[unconfirmed]** [note: with iChat, Trillian, and the vanilla AIM client, OTR works using the OTR proxy program which I couldn?t get to work, however, Gaim, Adium, and Miranda IM use an easier to implement OTR plugin which doesn?t require the proxy tool]

Operating Systems Supported: Windows/Mac (OS X)/*nix **[unconfirmed]**

Protocols Supported: AIM; in theory, most other protocols the aforementioned programs support should work as well (i.e. YIM, MSN, etc, though I haven?t tested them. Oh, and IRC which Gaim/Trillian/others support is also not encrypted, so, once again, see below for IRC encryption options).

Installation Example: Download the OTR plugin for Gaim and run the installer. Run Gaim. Click on Preferences and go down to Plugins on the left-hand side. Find ?Off-the-Record messaging? listed on the right, and check the accompanying checkbox. Restart Gaim. Go back to Preferences, and this time you should see ?Off-the-Record messaging? listed under Plugins on the left-hand side. Select ?Off-the-Record messaging? and click on the Config tab. Be sure that the ?Enable private messaging? and ?Automatically initiate private messaging? fields are checked.

You can now initiate the IM conversation with your chat partner. Once the conversation has been initiated, and assuming you possess a secure email account and/or secure phone line, you should contact each other and confirm the Key Fingerprint to help ascertain the identity of your chat partner. After the fingerprint is confirmed, go back to the Known fingerprints tab and, selecting the screenname of the chat partner whose fingerprint you have just confirmed, select Verify fingerprint and hit Close to exit out of the Preferences menu. You should now be ready to engage in secure conversations.

Name: **SecureIM** (http://www.ceruleanstudios.com/)

Key Strength: 128-bit Blowfish keys

Works With: Trillian

Operating Systems Supported: Windows

Protocols Supported: AIM/ICQ

Installation Example: Download and install Trillian. Run Trillian and, clicking on the globe on the bottom

left (or right-clicking on the Trillian icon in the taskbar and then going to Options), click on Preferences. Go down to AIM and/or ICQ under Chatting Services on the left-hand side, then select Misc. In the SecureIM section, be sure to check both ?Activate SecureIM Capabilities? and ?When possible, make a best effort to automatically maintain a SecureIM session with my contacts.? You?ll need to do this for both AIM and ICQ if you plan on using both protocols. Hit Apply and then OK to exit out of the Preferences menu.

You can now initiate the IM conversation with your chat partner. The locks in your IM window should turn red. You should now be ready to engage in secure conversations.

Name: **SSL Certificates** (Available from syLIkc.NET: http://secure.sylikc.net:8080/self_signed/ and Thawte: http://www.thawte.com/secure-email/p...tes/index.html)

[**IMPORTANT:**: www.aimencrypt.com also offers certificates, or rather *just one same certificate for everybody*, which in turn means that anyone can decrypt your conversations. In other words: **Do not use AimEncrypt!**]

Key Strength: 128-bit keys

Works With: AIM; and possibly other IM clients which allow importation of SSL certificates [such as??know of one? Then email me about it!]

Operating Systems Supported: Windows/Mac[**unconfirmed**]/*nix [**unconfirmed**]

Protocols Supported: AIM; (same as Works With)

Installation Example: pr0to has written a great tutorial on generating/installing a Thawte-issued certificate: http://www.rorta.net/index.php?page=aimcrypt, and the sylikc.net import instructions are here: http://secure.sylikc.net:8080/self_signed/aim.php. After generating/importing the certificate, you should now be ready to engage in secure conversations.

Name: **SimpLite** (http://www.secway.fr/us/products/all.php)

Key Strength: 1024 to 2048 bit RSA keys

Works With: Gaim, Trillian, and the following vanilla clients: AIM, ICQ, MSN, YIM, Jabber

Operating Systems Supported: Windows

Protocols Supported: AIM, ICQ, MSN, YIM, Jabber **[unconfirmed]**

Installation Example: Download and install SimpLite for your particular protocol (note that each protocol has a separate SimpLite program that you need to download). Run your particular flavour(s) of SimpLite and the Keys Generation Wizard should pop up. If it doesn?t, click on Keys in the menu and go down to Generate key pair. Follow the instructions and after a few steps you should have your key.

Run your supported chat program of choice, making sure that SimpLite is still running in the background. After sending a message to your chat partner, you should see your partner?s key show up in the SimpLite program, and your conversations should be under the Green authenticated/encrypted arrows.

Assuming you possess a secure email account and/or secure phone line, you should contact each other and confirm the Key ID to help ascertain the identity of your chat partner. You should now be ready to engage in secure conversations.

Name: **FiSH** (http://fish.sekure.us/)

Key Strength: 1080 bit Diffie-Hellman keys

Works With: mIRC, irssi, xchat

Operating Systems Supported: Windows/*nix/Mac (OS X) [**unconfirmed**]

Protocols Supported: IRC

Installation Example: Download the latest FiSH archive and extract the contents into your mIRC directory (wherever mirc.exe is located). Run mIRC and type ?**/load -rs1 FiSH.mrc**? (sans quotes). Close mIRC. Run the patch executable that matches your version of mIRC (click on Help, then About (or just click on that yellow icon on the far right of your toolbar) in mIRC to find out your version number).

When you extracted all of the files into your mIRC directory, you should have extracted a file called blow.ini-EXAMPLE. Open this file in Notepad and copy all of the contents. Close this file and open a blank Notepad window. Paste the contents and save the file as blow.ini (being sure to select ?All Files? from the Save As menu). You just did this so that you have a nice clean backup copy of the ini file in case you completely screw up this copy. For detailed information regarding setting up the blow.ini file, read the FiSH.txt file included in the FiSH archive you downloaded. However, a bare bones blow.ini file will look something like this:

[FiSH]

process_incoming=1
process_outgoing=1

plain_prefix="+p "

[#RORTA]
key=d8SfskY0riaqsfd19ks220dUtQZmKdeWrp8ksfdLjsoig4 9dp7G
encrypt_topic=1

The first two lines mean that FiSH will drecrypt all incoming messages and encrypt all outgoing messages, respectively. The plain_prefix line says that all messages you send that start with ?+p ? (note the trailing space) will be sent as plaintext (unencrypted). The next line is the name of the channel you want to encrypt (you can add more channels below, following the same format). The key value is the encryption key for your channel, be sure to make it difficult to guess by using a long string of mixed-case letters and numbers. The encrypt_topic line asks if you want to encrypt the topic in the channel (1 for yes, 0 for no).

As the FiSH.txt file rightly points out, exchanging channel key information in plaintext is a security risk. Thus, you should ideally tell other members of your channel the channel encryption key only through an IM window that has been encrypted using one of the aforementioned methods.

To encrypt private messages, either double-click on the user?s name to open up a private message window or message the user manually (**/msg username moo!**) and wait for a reply to get a PM window open (if you two aren?t in the same channel). Then right-click in the PM window and go to FiSH-->Auto-KeyXchange-->Enable, and then either close/reopen the PM window or/and click on DH 1080 KeyXchange (which is also in the PM right-click window under FiSH). You should now be ready to engage in secure conversations.

Nota Bene I: The FiSH encryption key is **not** the same thing as the channel key (mode +k). Naturally, your channel should also be set to modes +sk to further protect the conversation. First, type **/mode #channelname +s** (this prevents the channel from showing up in either **/whois** or **/list**), followed by **/mode #channelname +k yourchannelkey**. Your channel key should be **different** from your FiSH key, and merely means that no one can join the channel without knowing this key (to join the channel type **/join #channelname yourchannelkey**), whereas the FiSH key means that no one can *read* the conversation, irrespective of whether or not they can join the channel or not (network administrators can monitor all traffic on their server, even if they?re not in the channel with you).

Nota Bene II: You can further secure your IRC connection by using SSL (Secure Sockets Layer) (assuming both your client and the particular IRC network support it).

If you are using the latest version of mIRC (6.14+), instructions for setting up SSL are available here:

http://www.mirc.co.uk/ssl.html (the needed DLLs can be downloaded here: http://remus.oru.se/tsub/mirc-ssl/mirc-ssl.zip, or extracted from the OpenSSL installer linked to on the abovementioned mIRC site). Once you install the necessary DLLs, type **//echo $sslready** and you should get a reply of ?$true.? To connect to an SSL server you can use the **-e** switch before the server address or/and a plus sign (**+**) before the port number, for instance: **/server -e irc.rizon.net +9999**.

Consult the readme files of other clients for information on their SSL implementation capabilities. For instance, if you are using xchat on *nix, install the OpenSSL libraries (www.openssl.org) and then when connecting to the particular IRC server with SSL support add a plus sign before the port, e.g. **/server irc.rizon.net +9999**.

Some networks also let you set certain modes for the channel (for example, +S on Rizon), which require SSL to be enabled in order to join the channel (ask in #help or browse the network?s website to find out if SSL servers and SSL-Only channel modes are supported).

Nota Bene III: The great thing about IRC encryption is that you can encrypt entire channels, and thus have secure conversations between groups of more than two partners (something which, as far as I know, is not possible with any of the other aforementioned encryption tools), so appreciate it and enjoy it! :)

Caveats & Miscellanea

As you have doubtless noticed, there?s a plethora of encryption plugins, with various levels of key strength. The Gaim-Encryption plugin provides by far the strongest key pair (at 4096 bits), however, it doesn?t fly well on Macs. Therefore a feasible scenario may have one user running Adium on a Mac, while another runs Gaim on Windows, with both using the OTR plugin. Keep your options open, and always use the strongest key pair possible (combine malleability with security!).

There is no such thing as ?perfect security.? When I have repeatedly stated that ?you should now be ready to engage in secure conversations? don?t come crying when your key is compromised due to poor key handling on your part (insecure storage of keys, infrequent regenerations, etc.). In other words: **don?t get sloppy, you lazy sack of shit** (this is a note to self as much as it is general advice ;)).

On the subject of log files: many IM clients have the option to store logfiles of your conversations (and in many clients this option is enabled by default!?so be sure to scan the preferences/settings areas of your clients to disable logging). Logs are often (read: almost always) stored in plaintext, *even when you use the various encryption plugins!* Therefore if you do decide to enable logging, be sure to encrypt the logfiles themselves (info on encrypting data will be presented in a future segment of this Underground Security Paper series).

You?ve probably noticed that various clients/protocols/OSes have the ?**[unconfirmed]**? label after them. This is simply due to the fact that I haven?t yet tested the particular encryption tool on those protocols/systems. If you have, please let me know so I can update the information in subsequent versions of this textfile!

Finally, note that the ?installation examples? are just that: examples. As stated at the outset of this textfile, the focus has been on Windows and therefore the examples lean towards Windows scenarios. (Don?t take them too literally).

If you have any comments, suggestions, see any IM encryption plugin which wasn?t mentioned, or anything else, feel free to drop me a line at xcon0@yahoo.com. This is also the first textfile in a series I?m calling the Underground Security Papers. Successive papers will discuss encrypting emails, miscellaneous data files, as well as tips on maintaining anonymity and the like. To be kept abreast of more USPs, send me an email with ?USP? in the subject.

Enjoy!

_____

Under the pleasant norms of Parisian life, beneath the veneer of culture and civilisation, one of the bitterest and most sadistic underground wars of modern history was fought out.

Last edited by DIzzIE; 18th June 2006 at 07:32 PM.