Scatter Chat

User's Guide

Revision 0.2 (July 14th, 2006)

Table of Contents

A special thanks goes out to Trammel of the cDc Ninja Strike Force for proofreading this document and providing suggestions.

For those oppressed people who feel as if the rest of the world has long forgotten them...

Preface

This User's Guide is intended to be a highly readable document for people of all levels of technical expertise (even those who are entirely non-technical). If any part of this guide is confusing to you after reading it twice, then you have found an error that needs fixing; I would very much appreciate hearing about this since you are probably not the first person to be confused.

Non-technical beginners unfamiliar with cryptography **must** read this guide in its entirety from beginning to end.

0. Introduction

0.1: What Is Cryptography?

Cryptography is the science of scrambling information so that it is unreadable by unauthorized people. This process of turning information into unreadable data is called *encryption;* the reverse process is called *decryption*. For example, say you wanted to protect your personal diary from nosy room-mates, friends, or family members/etc. In this case you would use some kind of encryption software to encrypt the diary text. If one of your diary entries originally looked like this:

```
November 8th, 2004:  Today I purchased
400 shares of SCO stock.  I feel so
guilty!
```

... then after it is encrypted, it may look something like this:

```
Ams0PouYH2DC5cv88BjY14D/scS9+zNHW3qWevq
amKAEDOSZUEP1R08Mb63Vvnz7EqahDriAEqhMVJ
BDPwchQcmenYiwn3QcSrFzRz
```

As you can see, the diary entry is no longer legible unless it is decrypted somehow. The method used to encrypt the entry can easily be applied in reverse to decrypt it and recover the original message. Unauthorized people who do not have access to the encryption method (commonly referred to as the *encryption key*) would have to try decrypting it by all possible methods, which in modern systems would take an *extremely* long time even for well-funded, well-equipped governments (a warning is in order, however: don't get overconfident because, as the saying goes, the chain is only as strong as its weakest link, and this is obviously not the weak link!).

Cryptography is a very complicated branch of mathematics that some people have spent lifetimes studying. Surprisingly, however, you will only need to know and understand a few concepts in order to use this software effectively. They are described in Section 2: “Preparing To Use This Software.”

0.2: What Are Some Practical Uses Of Cryptography?

There are many legitimate uses for cryptography. For example, a politician

planning a campaign certainly wants to ensure only his trusted advisers have access to his campaigning strategies and plans, a democratic activist planning a protest against an oppressive government must keep the details of the operation secret, or an ordinary citizen may wish to keep an e-mail conversation regarding an embarrassing brush with a sexually transmitted disease secret from everyone but their friend. Of course, criminals, spies, and international terrorists all have good reasons to hide their data as well. The positive and negative social implications of cryptography are examined in more detail in Section 1: "Philosophy."

1. Philosophy

1.1: Why Is Cryptography Over Instant Messaging Needed?

Instant messaging(IM) services are highly insecure by their very nature. Below is a diagram showing data passing through an IM network:

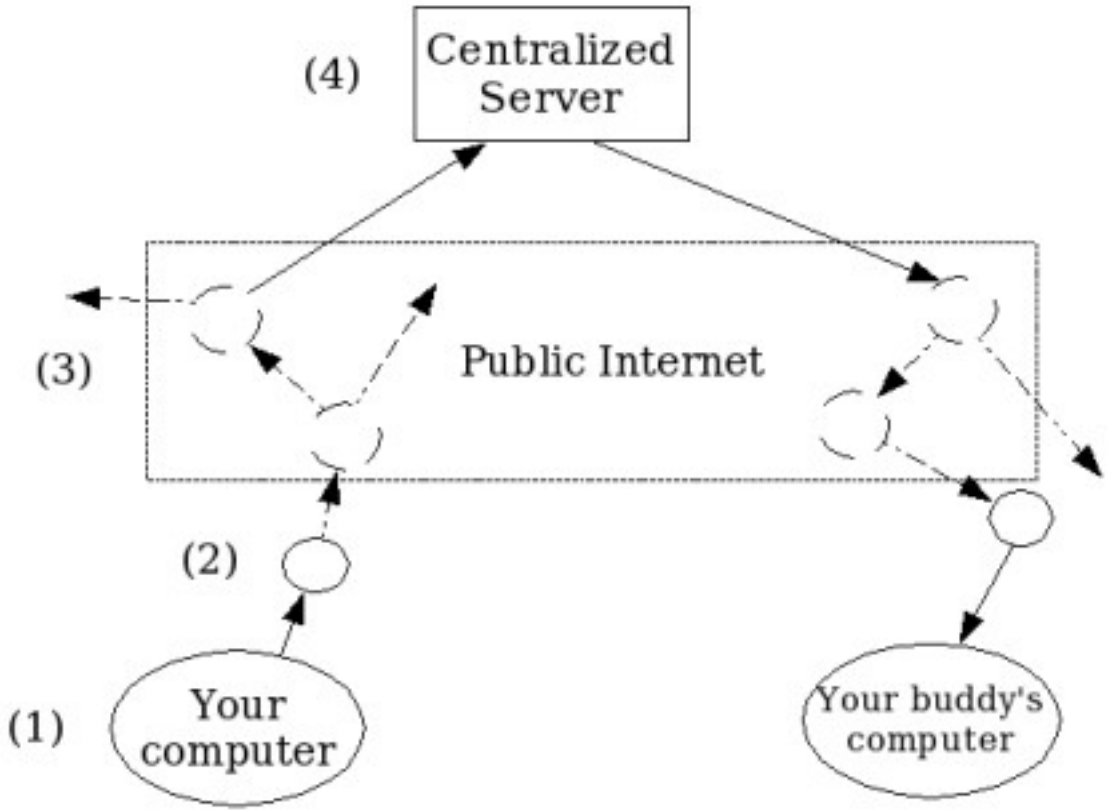


Figure 1.1

When you send a message to your buddy, it may seem as though it was kept private, but in actuality it passes through many points on the Internet.

The message begins at your computer (1), then is passed onto your Internet Service Provider's (ISP) router (a special Internet device which handles traffic—represented in the picture above with circles) at (2), which then gets sent off on the public Internet at (3). Depending on how your ISP is inter-connected with other ISPs, there may be a few routers it passes through, or there may be

many. Eventually, your message will reach a central server which is responsible for forwarding the message on to your buddy (on the AOL Instant Messaging network, this central server is owned and operated by America Online, Inc.; on the Yahoo! IM network, this server is under the control of Yahoo!). From here, it goes back out onto the public Internet where it eventually finds its way to your buddy's ISP and his/her computer.

It is important to note that *any and all* routers in the path could potentially record your message and/or send it to a third party. Your message could be seen by the legitimate operators of those routers, hackers who have taken illicit control of them, or any individuals, corporations, or government entities who could intercept the physical transmission *in between* the routers.

Furthermore, all information passes through a centralized server at (4) which naturally makes it a critical target; to intercept all messages on the entire AOL Instant Messaging network, an entity need only strike this point. Though it may (or may not) be hard to accomplish, a successful attack on this point would yield sensitive information from millions of users.

The deceptively innocent image of instant messaging regularly fools users into discussing embarrassing personal details, sensitive political topics, secret business strategies, and probably even top-secret government intelligence. World-wide hackers, organized criminals, industrial spies, and foreign (and local) intelligence agencies are all interested in this wealth of information that passes through IM networks; one would be foolish to think otherwise. Given the knowledge on how an IM network functions, the kind of information that passes through it, and the entities interested in intercepting that information, it is safe to say that anyone who believes their conversations are private is simply naïve.

1.2: Morality Issues with Cryptography

One moral issue that is repeatedly raised concerning cryptography is: “is it moral to build and distribute cryptographic systems if criminals and terrorists can use them to evade justice?” It is easy to let one's emotions lead to an answer of “No,” but a more careful analysis of the issue yields useful insight.

All technology can be abused. Because terrorists are known to use cellular phones to plan and coordinate attacks, would it make sense to ban cellular phones for everyone? With the advent of digital cameras, child pornographers no longer require expensive equipment and expertise to develop their own pictures; does this mean that digital cameras are bad and should be abolished? High-quality computer printers are routinely used to counterfeit currency; should these be outlawed?

Of course not. The technologies mentioned above are legal, and should remain legal, because they perform valuable functions for us. Even if we did banish them, motivated criminals will simply adapt and find alternatives, or

smuggle items in from foreign countries. No matter what, the original problem of criminal behavior and terrorism will persist.

Although it is recognized that some may abuse this software, it is strongly believed that its positive potential nevertheless justifies its existence. Millions of people can use this software to protect their data from identity thieves, malicious computer hackers, stalkers, organized crime syndicates, corporate spies, foreign (and domestic) intelligence agencies, stalkers, and criminals of all varieties. Previously, most people were unaware they could do anything to protect themselves, if they were even aware of the problem at all.

But the real reason why this software was created was to assist with improving human rights world-wide. Currently, various oppressive dictatorships are moving to censor and control information. This is unacceptable. By manipulating and censoring published news, the people are effectively blinded from the truth and cannot question the activities of their government. But given access to powerful technology that can keep their communications secret and secure, the people have a glimmer of hope. By knowing and spreading the truth, the people can eventually wrestle control back over their lives and restore peace, justice, and freedom.

Over time, and with your help, I strongly believe that this software can become a critical component in the fight for freedom. It is this reason that I have chosen to construct and distribute this work.

2. Preparing To Use This Software

2.1: First Things First

It is imperative to know that in any security system, the users of the system are *almost always* the most susceptible to attack. Although weaknesses in security technologies are quite common, the ease with which a smooth-talking adversary can trick users into submission is alarmingly high (this type of attack is commonly referred to as *social engineering*). Only a well-prepared and knowledgeable user can avoid being fooled.

It is because of this threat that it is strongly recommended that you become familiar with instant messaging (IM) basics prior to using this software in a serious manner. A complete novice who has never used an IM client will certainly become overwhelmed once cryptography is thrown into the mix. This type of user has little hope of using cryptography *correctly*.

Once the proper experience is gained, it would be wise to set up a test environment where this software can be experimented with. This is another important step in ensuring that you will know how to react in a critical situation. This can be done simply by using two IM screen names to practice sending encrypted messages back and forth. Most IM services allow you to create more than one screen name, as long as you sign up with differing e-mail addresses (oops, did I say that out loud?). Note that you may sign on each screen name from two different computers, or you may run everything on the same computer.

2.2: Critical Cryptographic Concepts

In the real-world, when you lock a safe that contains valuables, you are using two components: a specific brand of lock and a unique key. The security of this system is based upon the fact that only the owner has the key to unlock the safe, and not outside attackers. You also hope that the brand of lock you're using is secure and not flimsy.

Cryptography has related concepts from this model: it too uses locks and keys. Some locks (called *ciphers*) are very good, while others are very weak. And still others are weak, but appear to look strong at first glance. Luckily, when you use this software, you do not need to concern yourself with the ciphers in use since they were already carefully chosen by the author (for all you curious people, it uses AES, ElGamal, and DSA, which are believed by many mathematicians to be very strong. See the cryptography module's documentation for all the gory

technical details).

The *encryption keys*, on the other hand, require special attention. After you create them (how to do this is covered in a section 3.1), you must store them in a safe place. Anyone who was to steal your encryption keys will automatically be able to read all messages encrypted with them--at least, this is how it works in general; this software uses a certain advanced trick called *perfect forward secrecy* to prevent that situation from occurring even if the keys were stolen. Regardless, there are other nasty things that can happen to you, so you must still be extremely protective of your keys. Section 3.1 explains specifically how to create and protect your keys.

2.3: Authentication

So far, you have only seen that cryptography is useful for hiding data. However, it can also be used to prove (or *authenticate*) one's identity to another person. Here's a basic overview of how it works: say two people--we'll call them Alice and Bob--share an encryption key. If Bob moves across the country and gets a new e-mail address, he can still contact Alice later over the Internet and prove to her that he is Bob, even though she doesn't recognize the new address. To do this, she makes up a random English sentence, like:

```
You must be my lucky star!  'cause you
shine on me wherever you are!
```

... and she uses the shared key to encrypt it (by the way, that random English sentence above really isn't random. It is a quote from Madonna's song “Lucky Star”). She sends the encrypted sentence, which now looks something like:

```
63VvHW3qWevqamKAEDOSZUEP1R08MbiAEqhMVJB
DPv88BjY1wchQcmenYiwn3QcSrFzRzEqahDrAms
0PouYH4D/scS9nz72DC
```

... and sends it over the Internet to Bob. Since Bob has a copy of the encryption key, he decrypts it and reads the original message. He now can prove to Alice that he is Bob by sending back the original sentence she generated. When Alice sees her original message, she knows she is communicating with Bob since only Bob could have decrypted the sentence.

Although the above example is a bit oversimplified, it highlights how cryptography can be used to prove one's identity. In real applications, however, *digital signatures* are used for authentication purposes. Digital signatures provide more security than the example above, but they are out of the scope of this document.

2.4: Traffic analysis

As you have seen before, cryptography can be used to hide information. However, in many situations an eavesdropper can still gather intelligence from watching an encrypted conversation using a technique called *traffic analysis*.

The pattern of a conversation itself can sometimes reveal information that should remain secret. For example, by observing the lengths of the encrypted messages that pass by, an eavesdropper may determine the kinds of messages being sent. Furthermore, the frequency of messages, times that conversations begin and end, and the ratio of sent messages to received messages can all give an eavesdropper valuable information. Sometimes these details can be as equally sensitive as the messages themselves.

Unfortunately, there is little that this software can do to protect against traffic analysis. Random data is automatically mixed into each message sent, frustrating message length analysis. But the frequency of messages, timings of the conversation, and other bits cannot be regulated by this software in a meaningful way—not without placing too great a restriction on the ease of use. Therefore (and unfortunately), the responsibility to manage this risk falls on the end-users themselves.

Always remain mindful that even though all your messages are encrypted, you may nevertheless be leaking information through your manner of usage.

3. Using This Software

3.1: Encryption Key Generation

To exchange encrypted messages with a buddy, you first need to create an encryption key. When you load this software up for the first time, you will see a window titled 'Encryption Initialization Window':

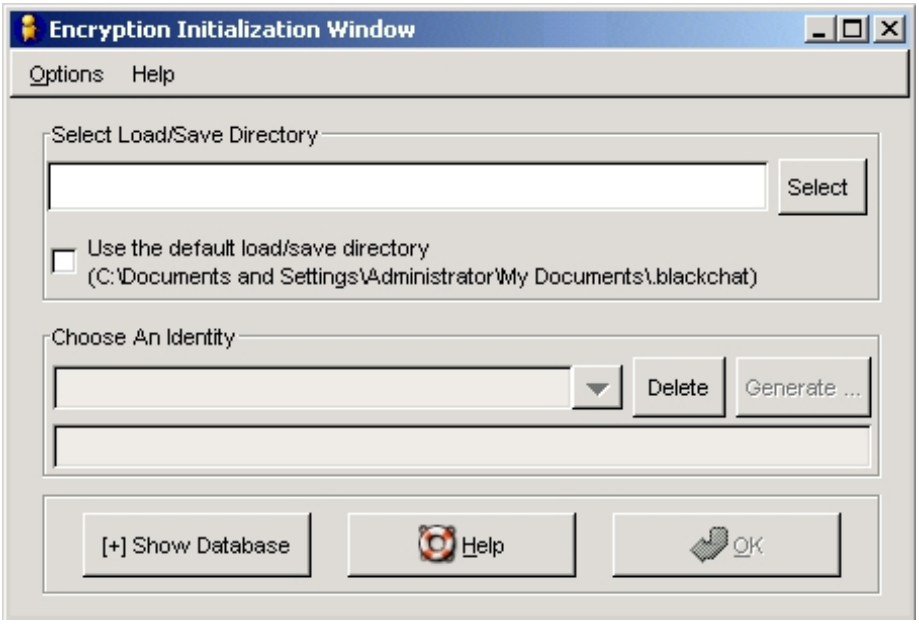


Figure 3.1

You must choose where you would like to store the new encryption key in the top section labeled 'Select Load/Save Directory'. For most users, it is sufficient to put the encryption key somewhere on your hard drive. Other (paranoid) users may want to store them on some kind of removable media such as a floppy diskette or a USB flash drive that way a close eye can be kept on them.

Beginners should store the new key in the default location by clicking on the checkbox labeled “Use the default load/save directory.” More knowledgeable users may wish to store the key(s) in another location; the 'Select' button will allow you to browse the hard drive.

Once you have chosen the location to store your key, enter the name you would like to call your new key in the box under the 'Choose An Identity' label (a

name is necessary to distinguish multiple keys you may create later on):

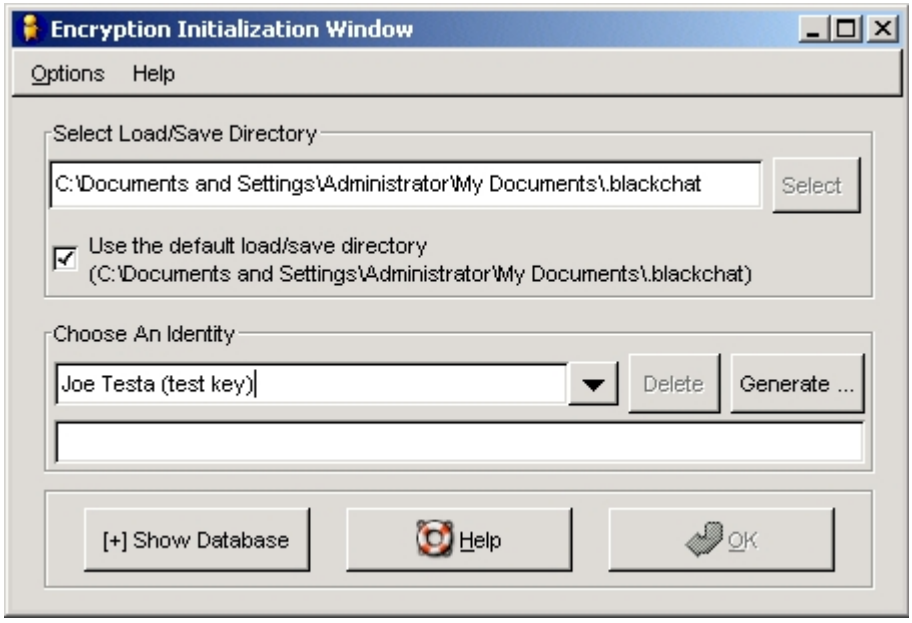


Figure 3.2

Next, click on the 'Generate' button to the right. Another window will appear called 'Key Generation,' which asks you to enter a password. This password will be used to protect the file containing your new key. You will need to provide this password each time you wish log on with the key in the future. Choose a good password so that unauthorized persons may not use your key. Enter the password a second time to ensure that it was entered correctly. Click on the 'Generate' button to begin the key generation process:

Figure 3.3

This process may take a few minutes to complete, depending on the speed of your computer. Once this is done, you can click the 'Close' button on the 'Key Generation' window to return to the 'Encryption Initialization Window.'

Congratulations on successfully generating an encryption key!

If this is the first time you've used this software, you probably don't want to use it for any serious purposes until you have more experience with it (as described in Section 2.1). Use this new key for testing, then create your “official” key another time when you are ready.

3.2: Setting Up Instant Messaging Accounts

Note: Do not proceed to this section unless you have a registered account already.

After generating a key, click the 'OK' button on the 'Encryption Initialization Window'. You'll be asked for the password to unlock the key; this is the same password you entered while creating that key.

Once you enter the correct password, the following window will appear:

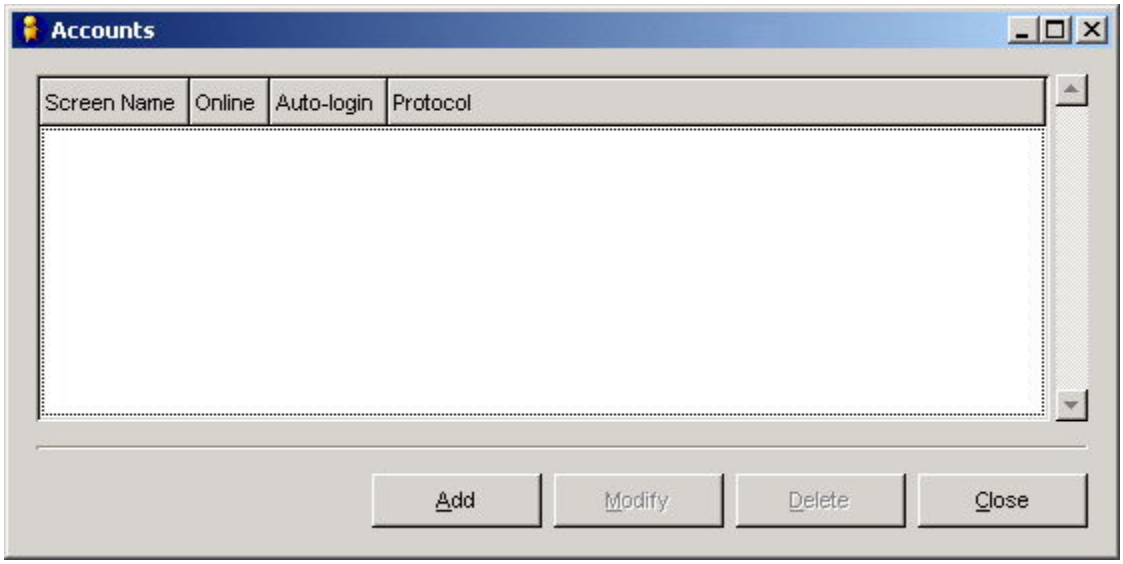


Figure 3.4

Click the 'Add' button to add an account. The following window will appear:

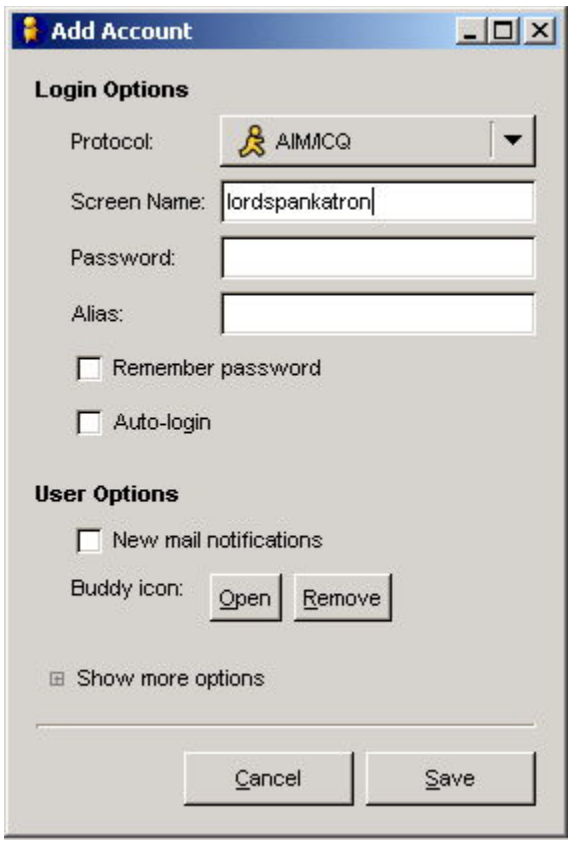


Figure 3.5

Select the type of service you have an account for in the 'Protocol' box. If you have an AOL Instant Messenger screen name, then leave the default 'AIM/ICQ' setting alone.

Next, enter the screen name you wish to add in the box labeled 'Screen Name:' and click the 'Save' button.

You will be taken back to the 'Accounts' window where you can either add more accounts if you have any, or close it and proceed to the main login window:

Figure 3.6

Now you can enter your screen name's password, and log onto the service. If successful, you will be shown your buddy list.

3.3: Using Encryption With Your Buddies

You can use this software to communicate with any of your buddies, but you can only use encryption with those who have this same software installed.

A typical instant messaging (IM) window looks like this:

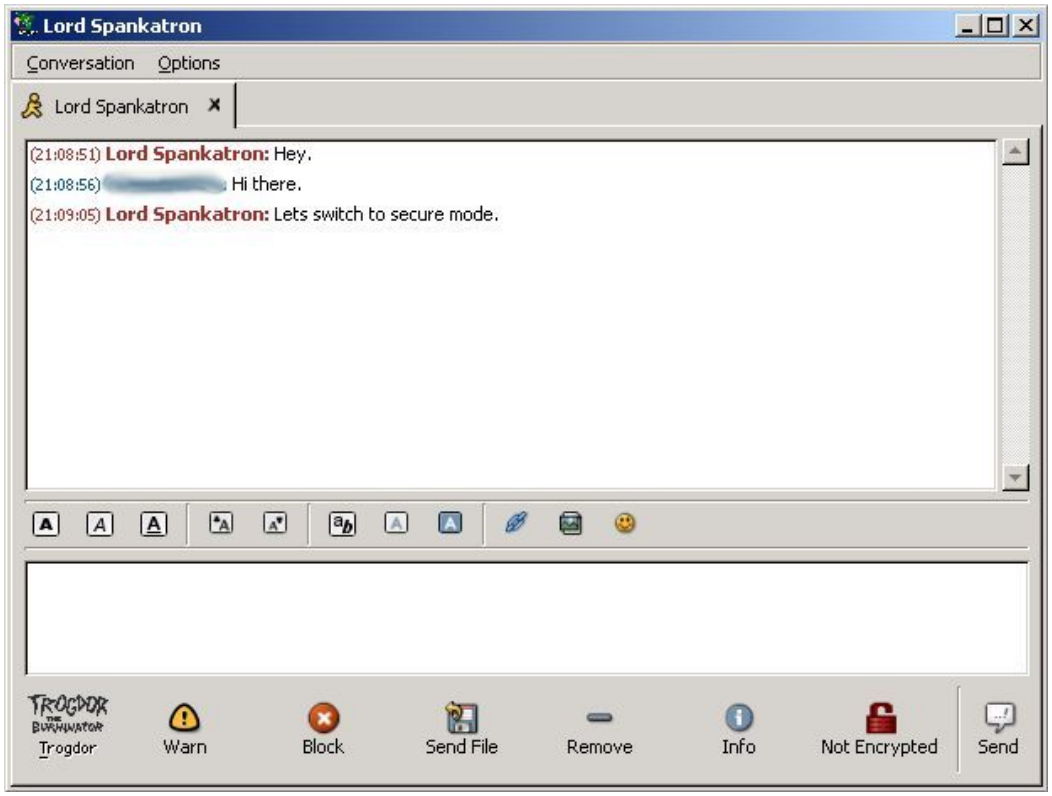


Figure 3.7

The conversation above is not encrypted (see the red lock icon labeled 'Not Encrypted' at the bottom of the image?). If the buddy you are speaking with is also using this software, you can enable encryption by clicking on that red lock icon and sending a message to your buddy like you normally would. Notice below that once the icon is clicked, it turns yellow to signify that your next message will start the secure channel:

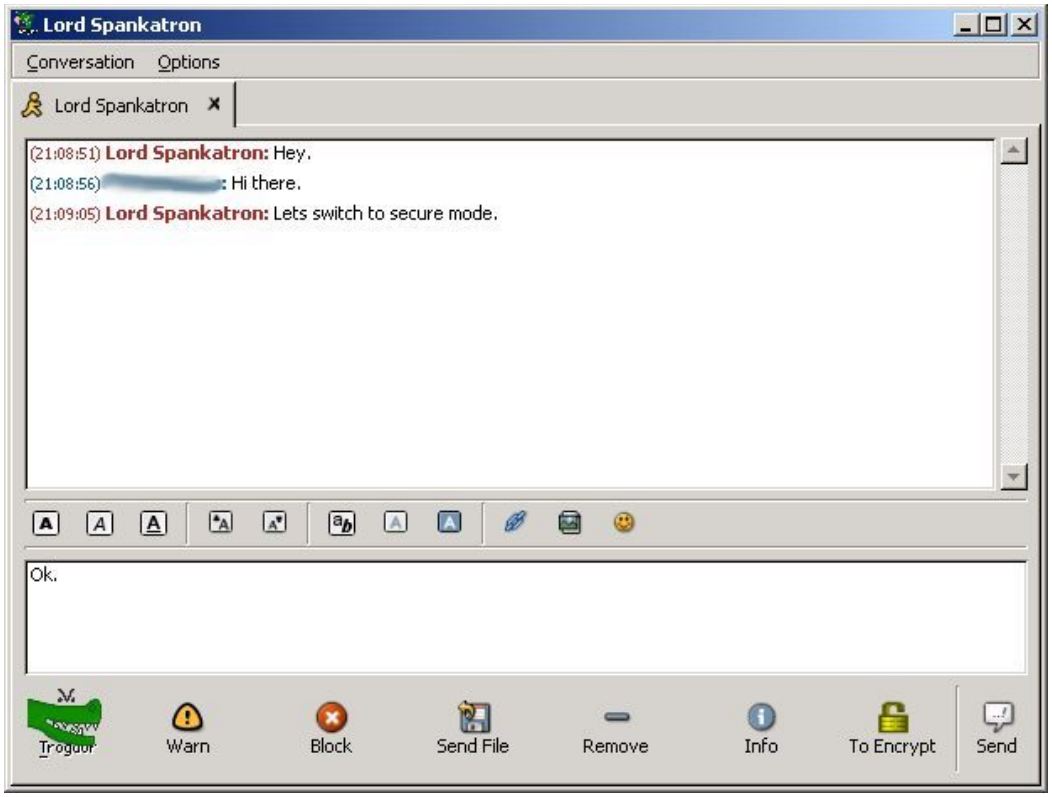


Figure 3.8

Once the secure conversation has begun (and after the buddy's key has been verified—see next section), the window will show a green icon:

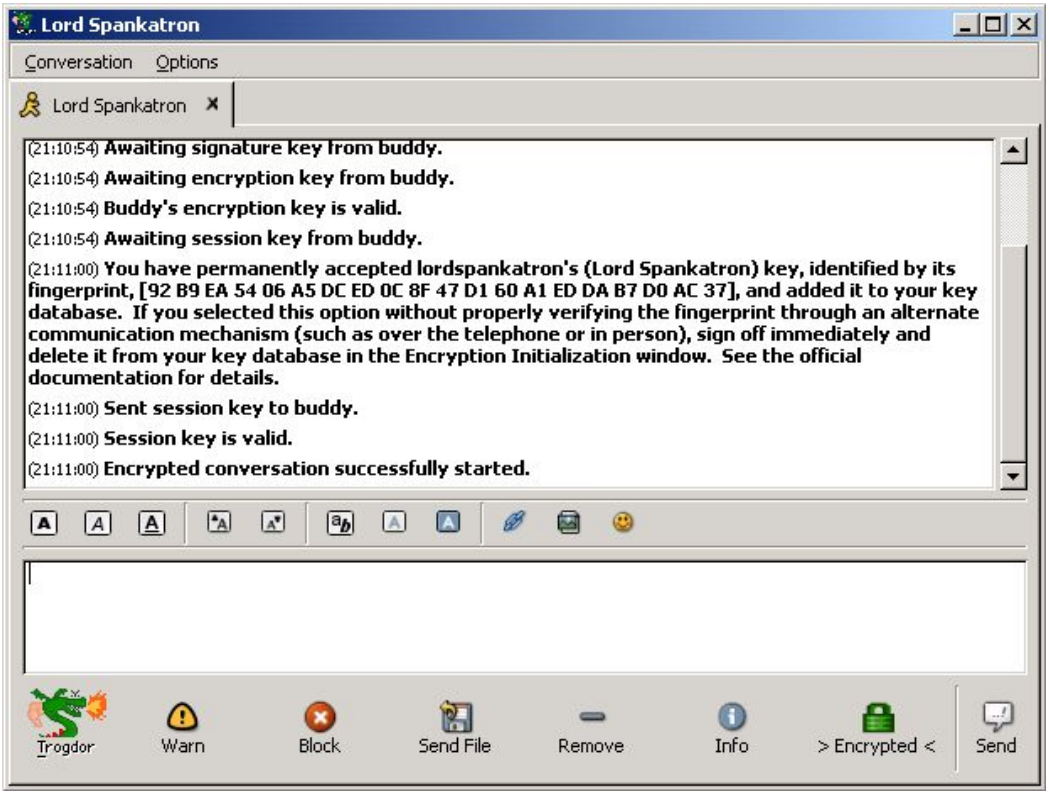


Figure 3.9

3.4: Key Verification

If the software ever receives an encrypted message from a user whose key it has never seen before, the 'New Key Fingerprint Detected' window will appear:

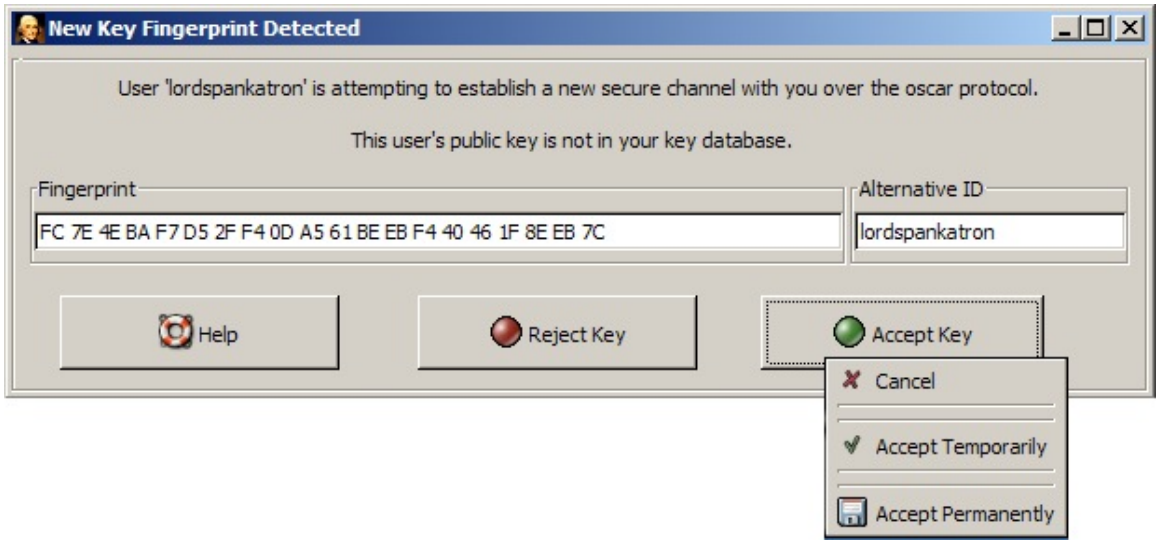


Figure 3.9

The window above informs you that a buddy (in this case, named 'lordspankatron') wants to establish an encrypted conversation with you, but your computer has not seen his key before. This can happen in three circumstances:

(1) The buddy is a new user of Scatter Chat and they have never set up an encrypted conversation with you before.

(2) The buddy has set up an encrypted conversation with you before, but they are using a new key that you have not seen before.

(3) Another person is attempting to impersonate your buddy.

It is important to note that it is impossible to tell the difference between (2) and (3) from your point of view. This is why key verification is **very important**.

The 'New Key Fingerprint Window' presents you with three choices: (1) to reject the key (along with the secure conversation), (2) to accept the user's key temporarily, or (3) to accept the user's key permanently. Below is a break-down of what each option does:

(1) Reject Key: this refuses the encrypted conversation. The key sent by the buddy will not be remembered by this software in the future. Furthermore, if you did not initiate the conversation, rejecting the key will prevent the buddy from knowing that you are capable of encrypted communications (this is useful for paranoid users who do not want to be “probed” by strangers).

(2) Accept Temporarily: this accepts the current encrypted conversation, but future conversation requests will need approval (the same window will appear and prompt you). Choosing this option will reveal to the buddy that you are encryption-capable.

(3) Accept Permanently: this accepts the current encrypted conversation, and causes the software to automatically accept future conversations from the same buddy key. Choosing this option will reveal to the buddy that you are encryption-capable. This option should only be selected once you have verified your buddy's key fingerprint (see below).

If you wish to keep your encryption capability private (like if you live in a country where cryptography is illegal), then you should reject all keys from strangers. This would prevent them from finding out you are using this software versus the

plain Gaim software which it is based upon.

Now that you know when this window appears and what the options are, here is a guide to what should be done to verify the key fingerprint.

First write down the key fingerprint that is reported by the 'New Key Fingerprint Detected' window. Next, you should either reject the key or temporarily accept it for this session, depending on how paranoid you are (see above for an explanation of what each choice will do). **Do not accept the key permanently.** If you chose to temporarily accept the key, keep in mind that the person you believe you are talking to may (or may not) be an impostor. Regardless of your choice, you must verify the key fingerprint you wrote down *using another reliable communications mechanism.* Ideally, you would do this by meeting in person and comparing the key fingerprint you recorded with the one reported in his/her 'Encryption Initialization Window' (see Figure 3.1). If they match, then the key is verified to belong to the correct owner and not an impostor. You can now safely trust this key fingerprint and accept it permanently the during next session, or manually add it to the key database (see Section 5.1). If they do not match, then the key cannot be trusted (if this key was accepted permanently on accident, it must be removed from the key database; see section 5.1 on how to do this).

Note that you *cannot* verify the key using the instant messaging conversation you just started since you don't know if you are really talking to person you think you are. This would be like asking the buddy, “are you an impostor?” Either way, the person on the other end is going to say “no.”

4. How To Avoid Getting Tricked

While there are no known attacks on the technical workings of this software, there are a few notable ways that a novice user can be tricked into lowering their defenses. Remember that in any security system, the weak link is almost always the user.

The first way is to establish an encrypted conversation for the first time with a buddy and verify the fingerprint through the new connection (refer to Section 3.4, “Key Verification”). This does not properly verify the key fingerprint because you do not know for sure that you are really speaking with your buddy and not an impostor (after all, determining the identity of the person speaking with you is the purpose of the key verification phase). So if ever you begin an encrypted conversation with a buddy for the first time and he/she suggests that you trust the key for any reason, then they are an impostor! Either that or they didn't read this manual.

A second way that someone could trick you is to steal your buddy's IM account password, generate a new key, then start an encrypted conversation. Because the key is new, the 'New Key Fingerprint Detected' window will appear (see figure 3.9), and you must verify this new key. If you do not properly follow *all* the steps necessary to verify this new key, then the impostor will successfully trick you. Be sure to follow all the instructions *every time* you are presented with the 'New Key Fingerprint Detected' window.

5. Tor

5.1: Tor Defined

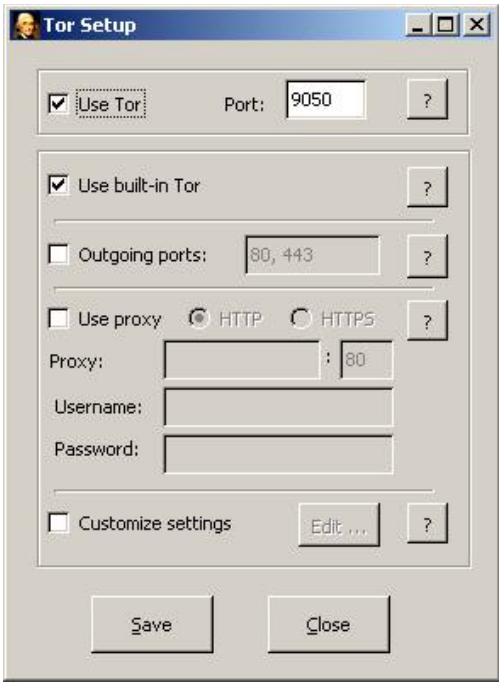
Tor is the name of a software project sponsored by the Electronic Frontier Foundation which allows computers to communicate with each other anonymously (see <http://tor.eff.org/>).

Tor can be used in conjunction with this software to provide an extra layer of security. Normally, when a user connects to an instant messaging service, eavesdroppers can easily tell what the user is communicating, and from where. When encryption is enabled in a conversation, the messages themselves are made private but the identities of the two buddies are still known by the eavesdroppers. In some cases this may not matter, but in some oppressive countries, encryption itself is illegal regardless of what it is being used for. Or, it may be the case that using encryption might raise too much suspicion. Tor can help in these situations by making it very difficult for eavesdroppers to tell who is communicating.

Thus, by encrypting a conversation and using Tor, an eavesdropper cannot tell *who* is communicating nor *what* is being communicated.

5.2: Using Tor

Tor can be enabled by selecting the *Options->Configure Tor* menu in the Encryption Initialization Window:



On the Windows platform, Tor comes prepackaged and is ready for immediate use with no further setup necessary; simply select the 'Use Tor' option and leave the 'Use built-in Tor' option enabled.

Although Tor will function for the majority of people using the default options, some people may have special arrangements that require attention. People behind firewalls that restrict outgoing connections will not be able to use Tor unless they connect through specific ports. Check the 'Outgoing ports' option and fill in the port(s) to connect out with. Do not select this option unless you are sure that you have to. Other people may need to connect to the outside world with a proxy; in this case, check the 'Use proxy' option and fill in the relevant boxes. This option also should not be enabled unless you are sure it is necessary.

Advanced users can specify custom options directly to Tor by selecting the 'Custom settings' option.

5.3: Using Tor Correctly

As of the time of this writing, Tor is still an experimental system. It should not currently be relied upon when strong anonymity is needed.

If you wish communicate anonymously over the instant messaging networks, *you cannot use an identifiable screen name*. This means that you must anonymously create a new name that no one will know belongs to you. You are not anonymous if you sign onto the America Online network with a screen name that you have used for years. Furthermore, you must never directly sign on to this

new screen name otherwise your identity will be discovered.

You must also be careful that you do not leak information through your habits as well. The way you type, the grammar you use, the capitalization of your sentences, the style of your language, and even the times you sign on and off can all leak information about you. You must be mindful of this at all times.

6. Advanced Features

6.1: The Key Database

The key database holds a record of all the trusted keys from your buddies. New keys are added when you select the 'Accept Permanently' option in the 'New Key Fingerprint Detected Window' (see Figure 3.9) at the beginning of an encrypted session. Optionally, you may edit this database manually.

Manually adding a key to the key database:

From the 'Encryption Initialization Window', click on the button titled, 'Show Database'. A section titled 'Public Key Hash Database' will appear. Click on the 'Add' button and a window titled 'Add Entry' will appear. Inside this window, you can add the key fingerprint and an associated UM ID (see section X). Click the 'OK' button to add the information into the key database.

Manually editing a key in the key database:

From the 'Encryption Initialization Window', click on the button titled, 'Show Database'. A section titled 'Public Key Hash Database' will appear. Highlight the entry you would like to edit and click on the 'Edit' button. Once your changes are complete, click the 'OK' button to save them into the database.

Manually deleting a key in the key database:

From the 'Encryption Initialization Window', click on the button titled, 'Show Database'. A section titled 'Public Key Hash Database' will appear. Highlight the entry you would like to delete and click on the 'Delete' button.

7. Dealing With Problems

This section gives guidance for various problems that you may encounter.

"I accepted a key I shouldn't have. What do I do now?"

First off, don't panic. Stop talking with all buddies and sign off immediately. On the 'Encryption Initialization Window' (see figure 3.9), click the 'Show Database' button. Find the key you wish to delete in the 'Public Key Hash Database' list, highlight it, and click 'Delete'. Once that is done, everything is reset back to before you mistakenly accepted the key.

However, if you realized your mistake after giving out critical information to an untrusted person, there is nothing this software can do to help (sorry!).

"The Chinese/North Korean/Cuban/Iranian government raided my house and stole my encryption keys. Now what?"

The bad news is that with your encryption keys, the government (or anyone else who has them), can impersonate new conversations with your buddies. You should notify your buddies if you can do so without giving them away to the authorities or getting yourself into deeper trouble. In the meantime, hope that your buddies are observant enough to detect changes in your writing style, mood, or personality.

The good news is that any previous conversation that may have been recorded still cannot be read. This is the result of a design feature called *perfect forward secrecy*: the only way to decrypt old conversations is by using the keys from *both* buddies. Unless the government raids your buddy's house also, your old conversations will continue to remain secure.