



Published on **The O'Reilly Network** (<http://www.oreillynet.com/>)
<http://www.oreillynet.com/pub/a/mac/2003/01/20/mail.html>
[See this](#) if you're having trouble printing code examples

How to Set Up Encrypted Mail on Mac OS X

by [François Joseph de Kermadec](#)
01/20/2004

Do you send postcards? Probably. They're great to convey best wishes for the New Year, to send a "get well" note to your neighbor who is in the hospital, or a "hello" note to coworkers from your holiday place.

However, you would never send confidential information on a postcard, would you? Certainly not since the postal workers and every single person who handles the card can read its contents while it travels through the post system.

Well, I have news for you! When you send email to someone, most of the time you're not sending them a letter, carefully enclosed in an envelope. You're basically sending them the electronic version of postcards that can be read -- or worse, altered -- by anyone during their transit over the network.

Indeed, standard mail is not a secure means of communication: every single note you send travels through dozens of servers before arriving in your recipient's inbox and can be stopped, altered, and resent without you or the recipient noticing it.

While this may be acceptable when you're exchange healthy cooking tips with your grandmother, this is certainly not the case in your everyday workflow.

Luckily, Apple just added support for S/MIME in the latest [Mail](#) version, the one that ships with [Panther](#).

S/MIME and Certificates

S/MIME (Secure / Multipurpose Internet Mail Extensions) is a protocol that adds digital signatures and encryption to MIME, the standard format used to send mail.

Thanks to this technology, you can now encrypt emails on the fly. In the real world, this is like putting your postcard into an envelope -- perhaps even better since envelopes can be easily opened. With this system, you can also "sign" messages electronically. Digital signatures attached to a mail prove two things:

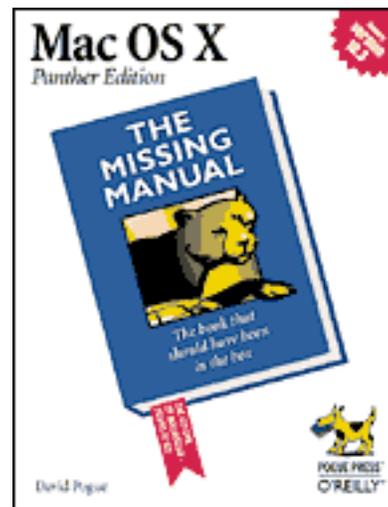
- That the mail has not been tampered with while it was in transit.
- That the mail has been sent by the email address that claims to have sent it.

This better secures your email, since you can make sure that nobody, except the intended recipient, can read them and that they cannot be tampered with unnoticed during transit.

Note that I didn't say that the signature proves that the mail comes from you. The danger lies here: most people think that it does and this can have legal repercussions. As we will see in a moment, you need to hire a third party that will "guarantee" that you are you and give you the necessary elements to sign and encrypt your mails.

However, at the first level, this third party, commonly called a certification authority, only has limited means to check that you are who you claim to be. Used at their full potential, signatures should be able to prove that a mail is sent by a specific person,

Related Reading



Mac OS X: The Missing Manual, Panther Edition
By David Pogue

but since there still is room for impersonation during the certificate creation process, I prefer to say that the signature guarantees that a mail comes from a specific address.

The Encryption Technologies

The mail signing and encrypting process relies on an encryption family of technologies commonly called "asymmetric" or "public key."

These encryption technologies rely on two "keys" to encrypt and decrypt information. What you encrypt with one "key" can *only* be decrypted with another -- and not even with the one that was used to encrypt it in the first place.

This is a great way of securing information. Indeed, you only need to keep one key secret. This key is called your "private key" and should remain secret *by all means*. It identifies you when it comes to signing mails and plays a vital role in this process.

The other key is the public key that you distribute freely so that people can encrypt information that they send to you. Once it is encrypted, only you -- the owner of the private key -- can decrypt it. Needless to say, losing your private key would have catastrophic consequences.

In most cases, this system is way more secure than older ones that only use one key, since they allow you to exchange information securely with dozens, hundreds, or thousands of people securely without having to establish a separate password for every one of them.

The CA: Certification Authority

Generating keys is not that hard -- your Mac can do it if you enter a simple UNIX command in your terminal. However, in our case, keys will not only be used for encryption but for authentication as well.

Since, by definition, mail users should not trust the sender of

the mail -- that can be impersonated easily -- we need a third person to tell them, "I have given the key to this person. Therefore, if you receive a mail with this key, you are dealing with this person." Of course, this is with the very important limitations we talked about above.

Therefore, you cannot generate the keys yourself, or the certificate would not have the authentication value it is supposed to have.

There are a few well-known certification authorities out there -- some will say too few, others too many -- and some of them even deliver the basic certificates that we want for free.

In a nutshell, they will perform some checks to "make sure" that you are who you claim to be -- usually by asking you precise personal questions -- and deliver a special signed pair of keys that you will use to sign and encrypt your mails.

Your recipients will then know that the keys are certified to be valid and will be more confident. The resulting signed public key is a "certificate."

Of course, to make sure that you are who you claim to be, these companies would need to perform more detailed checks: meet with you and ask you to show them your passport, for example. And that's exactly what they do!

Most of them deliver two kind of certificates. One that says "Yeah, we are 90% sure this person is who he or she claims to be" (and this is the certificate we are getting together today), and others that say "We are 99.99% sure this person is who he or she claims to be."

Open Source and Networks of Trust

You may have heard of a similar open-source system called PGP (Pretty Good Privacy) that delivers such keys. Unlike S/MIME, this system does not require that you ask for the certification of a third-party authority.

However, it requires that you rely on a network of people to whom you will show proofs of identification. Would you not meet other members of the group and show them your passport (or any other relevant document), the key you get is like any other homemade key.

With S/MIME, you still have to rely on a network of people to obtain 99.99% accuracy, but you can still reach a good level by dealing with your CA. Most CAs have their own network and can give you a list of people to whom you may want to talk to, so that they can give more value to your key pair.

Both PGP and S/MIME have their advantages, but S/MIME is widely used by companies and is probably far less intimidating for many people -- although this is only my personal opinion.

Serious About Cryptography?

Public key encryption and mail signing is a very complex subject if you want to learn the theory behind it. It is, however, as fascinating as it is complex, and I could not recommend enough an excellent O'Reilly book, [Web Security, Privacy & Commerce](#) by Simson Garfinkel and Gene Spafford.

It's an 800-page detailed look into all the aspects of what we are talking about today, while still being easily understandable by the average advanced user -- and experts will love it too.

Here is a (slightly adapted) quote from this book:

All that digital signatures prove is that the person or program signing the digital signature has access to a particular private key that happens to match a particular public key that happens to be signed by a particular certification authority. The goal of this identification system isn't to eliminate the possibility of impersonation but to reduce to acceptable levels the risk of impersonation and the resulting losses.

A big thanks to Simson Garfinkel for allowing me to reproduce it here.

Obtaining a Certificate

At first, obtaining a certificate seems complex; but actually, you only need to take your time and do things step-by-step.

As we have seen together, S/MIME requires that you obtain a certificate from a third-party certification authority. Since most Mac users will probably obtain their certificate from [Thawte](#), I will use their web site as a basis for the next part of the tutorial.

There is no obligation. Thawte delivers email certificates for free and is widely seen as a serious company, belonging in fact to the huge VeriSign group. If you want to use another certification authority, simply keep in mind that you will be required to give accurate confidential information to them and you should therefore only deal with trusted partners. You should also take your local legislation into account, and make sure that you are authorized to obtain a certificate and use cryptographic methods in your country.

I have personally dealt with Thawte. However, I would like to say that I do not specifically recommend or endorse them. You be the judge.

Choose Your Tools

The certificate-obtaining process will happen over an encrypted connection. Therefore, it is required that you use a web browser capable of handling SSL in a solid manner.

Also, at the end of the process, you'll be given a certificate file. These files use a specific format that only a few browsers can understand. Unfortunately, Safari is not one of them.

Therefore, you may want to download [Mozilla Firebird](#) and use it to get your certificate. In my experience, Mozilla complements Safari nicely and I love the fact that, like Safari, it comes from an open source effort.

Installing Mozilla is a simple matter of drag-and-drop.

When connecting to the certification authority web site, please make sure that Java and JavaScript are turned on, that you accept cookies, and that you do not block pop-up windows. Indeed, nothing must interfere with the creation process. Otherwise you may run into issues.

Connecting to Thawte

To go to Thawte's email certificate site, simply click [here](#).

You may want to take a moment now and have a look at their FAQs and support pages, available from the menus located on the lefthand side of the page. These pages do not contain any breakthrough information, but they are quite well written and will probably answer most of your questions about certificates. Also, it is important to know the certification authority you are going to deal with as well as you can.

For your convenience, here are a few quick picks: [personal certificates](#), [general concerns](#), [enrollment and privacy concerns](#)...

Their software pages do not mention Apple products or Mac OS X. There's no need to worry, however: the certificate you will get will be compatible with Mail and other Mac OS X mail clients.

Take a Deep Breath and Go

Once you are ready, you can click on the flashy "Join" button to begin the process. For more information about the encryption technology used by Thawte, click on the little key icon that Mozilla displays at the bottom left of the window. This will open an information window with some security-related stuff.

The first page is the very boring, traditional "Terms and

Conditions" text. It is essential that you read it very carefully. I did and it's a lot easier than one would think at first sight. The text can be understood easily and it provides a nice introduction to the legal responsibilities that are associated with keys and certificates.

If you accept the conditions, click on "Next."

We are now going to provide some traditional identification information to Thawte. Up until now, it's all very mundane but I cannot stress enough the importance of entering accurate information. Double-check everything, especially the date of birth: it's day/month/year.

Thawte provides a charset pop-up menu to allow international users to enter their names -- have a look at mine and you will understand what I mean! However, they suggest that you remove any non-English characters from your name if they have an easily understandable equivalent. Indeed, this will allow older mail applications and web browsers to deal with the certificate easily.

The next page asks for more personal information, including an Identification number. This number allows Thawte to perform mathematical checks and, in theory, will prove that you are who you claim to be. Needless to say, this is not a 100% guarantee and this is why the certificate that you will get won't contain your name -- to include your name on your certificate, Thawte requires that you either pay a fee and allow them to investigate or rely on a "network of trust."

Pick very carefully the number that you want to give and try not to

Related Reading



Running Mac OS X Panther
Inside Mac OS X's Core
By James

give a number that would immediately allow a malicious person to impersonate you. Of

course, all of them are very sensitive information. However, in some countries -- the U.S. for example -- a social security number is used very commonly, while others may give you more time to react in case you realize they have been stolen. It's up to you and depends entirely on the country in which you reside.

Again, the number should be accurate. Otherwise, the verification process can fail and you could run into trouble.

On the same page, you are going to create your account. Enter a valid email address that you can check securely -- there are some excellent articles on the Mac DevCenter about secure mail reading, like [this](#) one from [Jason McIntosh](#).

Note that you should provide the address *for which you want to obtain the first certificate*: certificates are linked to your email address and you will have to obtain one per address. Of course, you will only have to go through this registration process once.

Using the certificate you will obtain with an address for another email address is a bad idea and will cause most email clients to complain, effectively ruining the trust effect you are trying to establish.

Thawte provides you with a privacy guarantee and certifies that your mail won't be used for advertisements purposes. Of course, you are free to forbid them entirely to send you mails, but I don't suggest that you do that -- in case there really is an emergency, you don't want to discover it too late.

On the next page, you will be asked for language preferences. I suggest that you choose English to be on the safe side, but you are welcome to pick anything else. I left the charset to the default, but some international users may want to modify it.

Let's pause for a second here and note how lucky we are to use

Mac OS X: indeed, its support for multiple languages and charsets really gives us all peace of mind.

Next, you will be asked for password information and to provide five "password recovery" questions -- on two pages. This is perhaps the most important part of the process: indeed, if someone were able to break into your account, this person could effectively impersonate you and cause you to run into very serious legal trouble for illegal actions you didn't perform.

Thawte's password page is quite well written and gives you some important information about how the system will use it. Again, read it carefully.

Make sure that you use a *secure* password and write it down -- yep, for once, I suggest that you write it down. Of course, make sure that you write it on a special piece of paper that *nobody* can see.

After clicking on "Next," you will be asked for the password questions. I do not like the default questions much... Indeed, the answer to them is quite easy to find and almost everyone could, with a bit of research, give correct answers.

Therefore I strongly recommend that you use a mix of default and personal questions -- and please, oh please, do not use your mother's maiden name and the number of pets you have!

In any case, do not forget the checkboxes next to the questions, and try to provide answers that do not include accented characters. I prefer not to risk running into trouble for this.

Again, write down your choices on *another* piece of paper.

Your questions are as important as your password since anyone could get your password by using them. So you should make sure that the two pieces of paper are in two different safe places. Ideally, bank safes, and certainly not drawers or bookcases in your office or home.

You will also be asked for a phone number. Provide a number

that you are likely to keep for awhile and that you usually answer personally. You should also make sure that this is a number that can be called from outside your country -- i.e., not a special toll-free, short, or extra-costly one. And include the appropriate international codes.

Finally, you will be presented with a summary of the information you have entered. Check it twice and click on "Next" so that Thawte can ping your email address.

Remember when we said that keys are linked to an email address more than to anything else? To make sure that you are who you claim to be, Thawte will send you a mail and ask you to click a link and enter a special code contained in the mail. This is another reason why you should make sure that your email inbox is safe and that you are the only one who can access it.

Once you have successfully confirmed the "ping," you have completed the registration process. This is the longest and most dangerous part since you can now request multiple certificates without having to through it again. You can then safely delete the mail.

Claiming Your Certificate and Creating Additional Ones

Now that you are a registered Thawte user, it is time to claim your first certificate. Click on the "Next" button to begin the process. Later, to obtain another certificate associated to another email address, you will simply need to repeat this process.

Remember that it is now crucial that you use Mozilla Firebird!

You now need to select the certificate type. As the page itself says, you should select "X.509."

More Forms to Fill In

You will now have to go through an 8-page form to select your certificate preferences. Here are a few tips that will allow you to

do it quickly and painlessly:

1. On the first page, you need to select the format. Thawte seems somewhat PC-centric, but you want to select "Netscape Communicator or Messenger," the default value, and click on the Request button.
2. Remember what we were saying about your name not being included in the certificate for now? Here is an example! For now, this page is useless, so you can click on "Next" without changing the default values.
3. You will now need to select the email address to which you want to link this certificate. Since the Thawte system only knows the one you indicated as your default address, you simply need to click on the checkbox and click on "Next." Later, you will be able to enter additional addresses and request certificates for them as well.
4. On the next page, click on "Next" directly.
5. You will then need to select the "extensions" that will be included in your certificate. Simply click on "accept" to include the default ones. It will work beautifully.
6. You now have to select an encryption strength for your keys. I recommend that you select the highest available (2048 at least) and click on "Next."
7. Here, Mozilla will wake up and ask you for a password for the "Software Security Service." This "Software Security Service" is the Mozilla component that will handle the keys handed out by Thawte -- and this is the component that other browsers lack. This password is *as important as the one given to Thawte* since a user who could break into the Mozilla Certificate Manager could get your keys. It should, however, not be the same. Write it down and put it in a safe, like the others. Make sure that nobody ever knows it. You have now reached the final stage, the actual key generation process. You can then click on "Next."
8. You will then be given a chance to review your information one last time.

The process is now completed and Thawte's computers are working to create your certificate. You can click on "Next" to

go to the certificate manager and close the small window.

The certificate manager contains a list of all the certificates that you have requested. Although Thawte has powerful computers, the generation process will take a little while.

After a few minutes, the status of your certificate will change from "Pending" to "Issued." You can now use Mozilla to "fetch" it -- i.e., download it into Mozilla's certificate manager.

Once it is issued, click on the "Navigator" link, in the "Type column." This will direct you to a page with important certificate information, like the expiration date and the assurance level.

At the bottom of the page is a cute "dog" button to fetch it. Click on it and stare in amazement at a Mozilla that does ... nothing.

Well ... not really. In fact, Mozilla has installed the certificate into the certificate manager for which you previously provided a password. It simply didn't tell you that the operation was successful.

You can now exit Thawte's site with confidence.

Transferring the Certificate

We chose Mozilla because its certificate manager understands the certificate file format and can actually handle it. However, for the exact same reason, we need to perform an additional step.

Indeed, as with any good Mac OS X application, Mail looks for security and cryptography elements in the "Keychain," not Mozilla's very own management system.

Therefore, our last step will be to transfer the certificate from Mozilla to the Keychain utility.

Preparing the Keychain

This step is optional, but I recommend that you follow it. Indeed, although most users use the Keychain without even thinking about it, this application has some features that can greatly enhance the security of your data.

A Keychain is, in fact, an encrypted file that contains sensitive information like passwords, secure notes, and yes, private keys.

When you log in, Mac OS X's default behavior is to "unlock" the keychain. In other words, it decrypts the file.

When a Mac OS X application needs a password, it automatically asks the Keychain for it. If the keychain is unlocked, Mac OS X will look at the access authorizations for the password.

If it is set to "Allow all applications to access this item," it will give the password to the application silently. Or, if you have it set to "Confirm before allowing access," it will ask for your permission first.

This is a very secure system, since you can set the access authorizations yourself -- Mac OS X pre-sets them for you if you don't want to deal with this.

However, since private keys are so important, we want to keep them in a "locked keychain" (encrypted file) that we will only unlock on demand.

Sure, we could change the Mac OS X default behavior and not unlock Keychain automatically at login, but this is not convenient for our less secure passwords such as Safari auto-fills and mail accounts. For them, having the Keychain unlock itself automatically and setting access authorizations on a password-by-password basis should be enough. However, you be the judge.

Therefore, we are going to create



an additional keychain where we are only going to store our certificates. In order to do so, open the "Keychain access" utility, located in the "Utilities" folder.

Then use the "File" menu to create a new Keychain. Give it a good name and click on create. The next step is to create a *good* keychain password. Again, this password is as important as your Thawte account password but should not be the same. You should also be able to learn it by heart since you will have to type it to use your certificates.

Here's a tip: use the Keychain Access "View" menu to select "Show status in Menu Bar." This will be handy later on.

Now that the Keychain is created, minimize the "Keychain Access" window and go back to Mozilla.

The Transfer Process

To transfer the certificate, you will first need to access the certificates manager.

In order to do so, use the "Mozilla Firebird" menu to open the "Preferences" sheet. Then click on "Advanced" and use the disclosure box located next to the "Certificates" item if needed. Finally, click on "Manage Certificates."

The window that appears will show you all your key pairs. Select the one you want to export and click on "Backup." This will tell Mozilla to package the pair into an (encrypted) file and

Related Reading



Web Security, Privacy & Commerce, 2nd Edition By **Simson Garfinkel**

[Table of Contents](#)
[Index](#)
[Sample Chapter](#)

Read Online--Safari

Search this book on Safari:

Only This Book

Code Fragments only

to save a copy of it somewhere where you can access it directly.

Give the backup file a name and save it onto the Desktop. Then pick a password for it. The password can be weaker than the others -- but not too weak, of course. You do not need to write it down, but simply to remember it for 2 minutes.

Once the file is on your desktop, you can quit Mozilla. Now double-click on the file as if you wanted to open it. This will launch (or unminimize) Keychain Access and it will ask where you want to import it.

Select the Keychain that you just created and click on "OK." The Keychain will now contain your private key and the associated certificates.

Certificates contain no secrets and are made public when you send a signed mail. There is therefore no need to protect them better than what we have done.

Your private key, however, is very important. To protect it even better, we are going to restrict access to it. To do so, click on it once and select "Access control" in the bottom half of the window.

In the panel that appear, deselect "Allow all applications" and pick "Confirm before allowing access." Now, Mac OS X will prompt you for confirmation before allowing an application to access the private key, *even when the Keychain is unlocked*.

The most paranoid of us (in the positive sense of the term) will want to check the "Ask for keychain password" box. When this option is selected, Mac OS X will ask you for the keychain password before allowing access to the private key *even when the keychain is unlocked*.

There is one minor drawback that you should be aware of. With this method, when you want to send a signed mail, Mail will begin the signing process, ask for your permission before fetching the certificate, and sending the mail. If you, for any

reason, deny access to the certificate, the recipient would receive a mail with a message that states that the signature wasn't verified successfully, leading him to think that the mail has been tampered with.

Finally, drag the backup file created by Mozilla to the Trash and use the "Finder" menu to "secure empty" it. If you want, you can remove the certificate from Mozilla's certificate manager -- since you do not want to keep unneeded copies of such sensitive files on your hard drive.

You can now safely quit the Keychain Access application.

Before sending signed mails, use the "Keychain" menu to unlock the keychain that contains your private key and certificates, although you can also do that on-the-fly while sending the mails. When you are done, use the menu again to lock the Keychain, greatly enhancing the security of your keys.

Using Mail

Now that we have gone through this lengthy process, we can go back to the typical Apple way of doing things.

It's now time to fire up Mail and to click on the "New" button to create a blank mail. Mail will automatically detect that you are the proud owner of a certificate and display a button on the top right of the mail-composing window.

If you have multiple accounts in Mail, you will need to use the "Account" pop-up menu to select the account that the certificate is associated with before being able to see the button.

Signing Messages

The mail-composing process does not change at all. Just make sure that the button is clicked (it is filled with a dark gray color) and contains a checkmark in a black badge). This means that the message will be signed when you send it.

If you not want to sign a message, click on the button. The

color lightens and the badge contains a small cross.

You can send signed message to everyone. Mail will send the message along with the necessary elements for the other computer to check your signature -- your public key.

Receiving Signed Messages

You receive signed messages like any other ones. The only difference lies in the last header of the message, displayed at the top of the window. You will see a header containing the small "Signed" badge, indicating that this is indeed a signed message.

If the message does not contain the public key or has been modified by a malicious user, a big yellow band will appear at the top of the message window, stating that Mail was unable to verify the message signature. This is usually a bad sign and should ring warning bells immediately.

As soon as you receive a signed message, Mail will import the sender's certificate into your login keychain.

Sending Encrypted Messages

Remember, to send an encrypted message, Mail needs to know the recipient's public key so that he can then decrypt it with his private key. Therefore, you can only send encrypted messages to people whose public keys you already have in your Keychain.

The easiest way to obtain someone's public key and immediately send this person encrypted messages is to ask her to send you a signed message. Upon arrival, Mail will store the certificate in the keychain and allow you to encrypt messages that you send to this person.

The process is exactly the same as when signing messages. However, this time you need to pay attention to a second button: the one with a padlock icon on it.

The padlock can be unlocked (the message won't be encrypted) or locked (the message will be encrypted).

You can send an encrypted message without signing it. However, this is not really a good thing to do since the message you are sending is probably important, and adding an authenticity check to it greatly improves the security of the transfer.

Receiving an Encrypted Message

In typical Apple fashion, receiving an encrypted message is completely transparent. When you open the message, you will immediately be able to see its contents, and this leads some users to think that the process failed.

However, the security header will state that it has been encrypted during the transfer.

Final Thoughts

Although obtaining a certificate is not the most straightforward thing in the world, it's easy enough to do, as is installing the certificates you obtain.

Apple's implementation of S/MIME support in Mail allows every user, whether they are experienced or are using a Mac for the first time, to protect the mail they are sending by encrypting them. And that's a very good thing. Indeed, using certificates will greatly increase the security of mail communications by reducing (not eliminating) the risk of impersonation, and preventing mails from being tampered with.

Talk to people about mail certificates and signing and try to use this method as often as possible. The security and comfort it provides are great and, since it can be integrated into your everyday workflow without any difficulty, it can only be an improvement. Encourage them to get certificates and to use them too.

However, you should not forget that signing a mail is like

signing a piece of paper. Sure, someone can falsify your signature like someone can steal your private key, but in most countries, you are held responsible for what you sign. A signed mail comes with legal consequences and you should take every single step you can to protect your private key. For example, do not use them on shared computers. Keys are not something to play with, but they definitely are something to use when you are serious about the integrity of your written communications.

François Joseph de Kermadec has been a full-time Mac OS X user since the 10.0.4 release. He began posting on the Apple Discussions in February 2002, where he enjoys helping fellow Mac users and writing user-contributed FAQs.

Return to the [Mac DevCenter](#)

oreillynet.com Copyright © 2003 O'Reilly & Associates, Inc.