MIME encrypted messages.

From James Walker: Exchanging Signed or Encrypted E-mail

Suppose that Alice and Bob are both using Mac OS 10.4 and *Mail*, and want to send signed or encrypted e-mail to each other. They can set it up as follows:

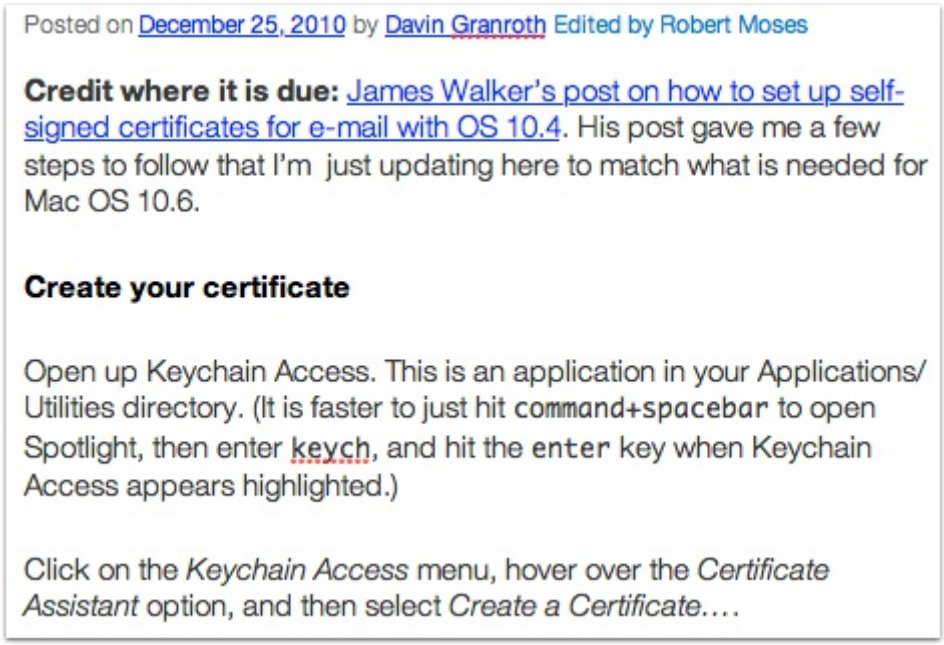
1. Alice creates a certificate for herself.
2. Alice tells her computer to trust her certificate.
3. Alice uses *Mail* to send a signed e-mail to Bob.
4. Bob creates a certificate for himself.
5. Bob tells his computer to trust his certificate.
6. Bob uses *Mail* to send a signed e-mail to Alice.
7. Bob reads the e-mail from Alice. *Mail* indicates that the signature cannot be verified, but the act of receiving the message puts Alice's certificate in Bob's keychain.
8. Bob tells his computer to trust Alice's certificate.
9. Alice reads the e-mail from Bob. *Mail* indicates that the signature cannot be verified, but the act of receiving the message puts Bob's certificate in Alice's keychain.
10. Alice tells her computer to trust Bob's certificate.

Why would you want to send encrypted e-mails?

Hah! "Why wouldn't you want to," is the better question. Actually, if you send or receive sensitive information like usernames and passwords, legal information, or confidential business information, you might really want to consider this. The trick is getting the person you exchange these messages with to also set up S/MIME on their end of the e-mail.

S/MIME For Apple Mail

S/MIME (Secure/Multipurpose Internet Mail Extensions) is a standard for public key encryption & signing of MIME data. S/MIME functionality is built into the majority of modern email software & interoperates between them. S/MIME provides the following cryptographic security services for electronic messaging applications:
Authentication
Message integrity
Non-repudiation of origin (using digital signatures)
Privacy
Data security (using encryption)
Before S/MIME can be used in any of the above applications, obtain and install an individual key/certificate. Encryption requires having the destination party's certificate on store (which is typically automatic upon receiving a message from the party with a valid signing certificate).



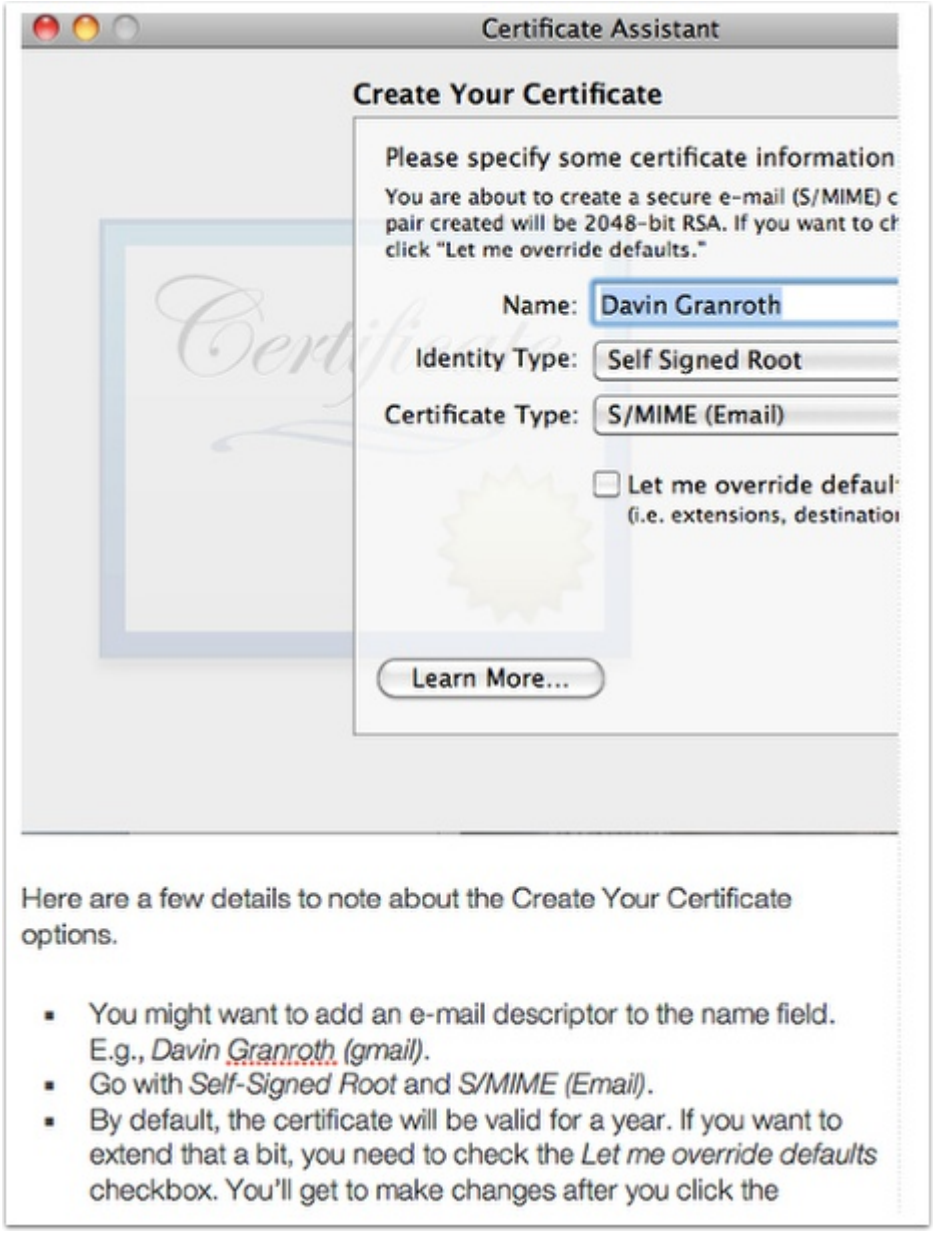
Posted on December 25, 2010 by Davin Granroth Edited by Robert Moses

Credit where it is due: James Walker's post on how to set up self-signed certificates for e-mail with OS 10.4. His post gave me a few steps to follow that I'm just updating here to match what is needed for Mac OS 10.6.

Create your certificate

Open up Keychain Access. This is an application in your Applications/Utilities directory. (It is faster to just hit `command+spacebar` to open Spotlight, then enter `keych`, and hit the `enter` key when Keychain Access appears highlighted.)

Click on the *Keychain Access* menu, hover over the *Certificate Assistant* option, and then select *Create a Certificate…*.



Here are a few details to note about the Create Your Certificate options.

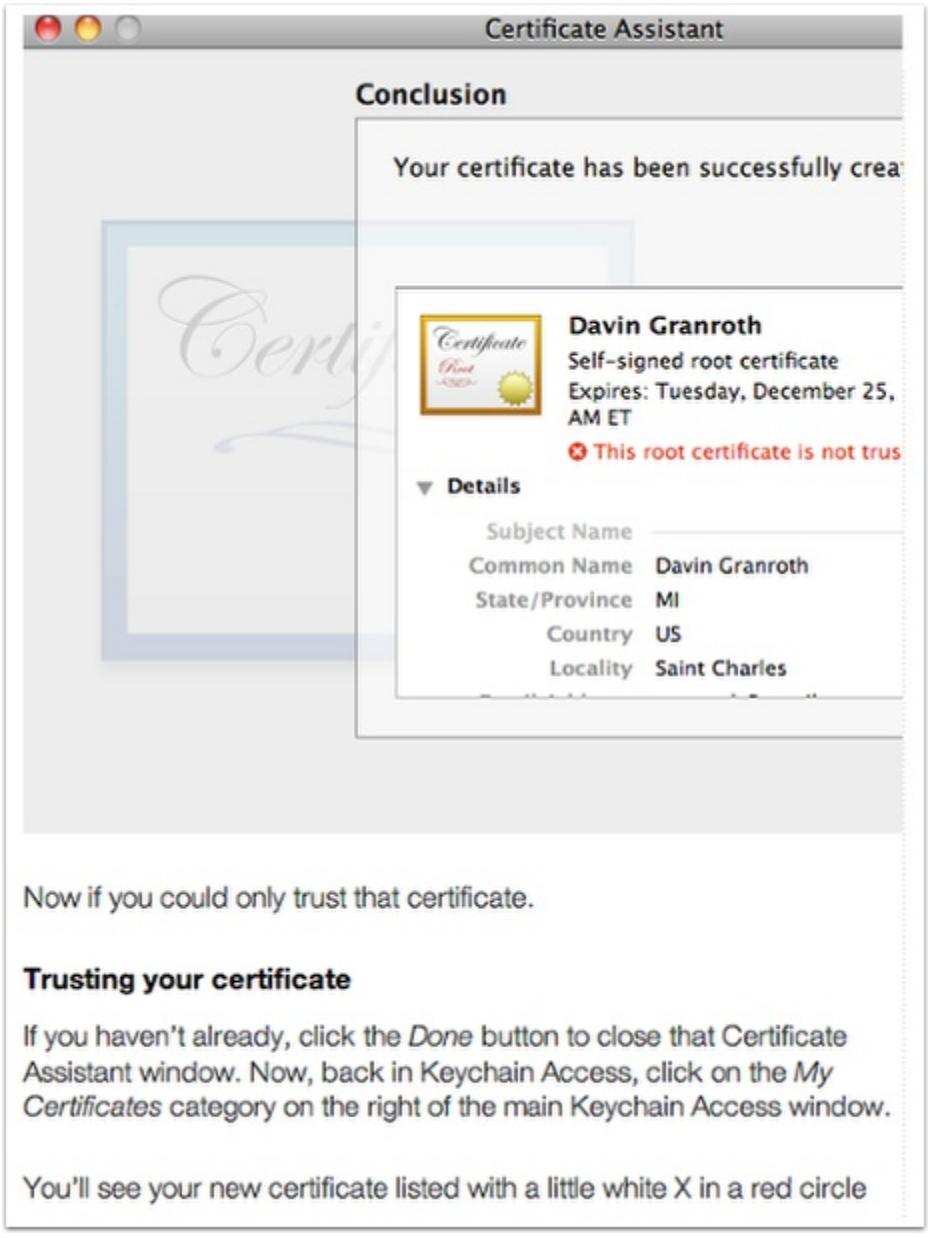
- You might want to add an e-mail descriptor to the name field. E.g., *Davin Granroth (gmail)*.
- Go with *Self-Signed Root* and *S/MIME (Email)*.
- By default, the certificate will be valid for a year. If you want to extend that a bit, you need to check the *Let me override defaults* checkbox. You'll get to make changes after you click the

Continue button.

- If you need a certificate for your non-primary e-mail account, you'll need to check the *Let me override defaults* box for that too.

If you checked the override box, you'll eventually see a series of *Extension* windows. Just go with the default values. Apple figures out what you need based on the first screen where you chose the certificate type.

Continue and you'll see a window with your new certificate information in it. Congratulations!


Certificate Assistant
Conclusion
Your certificate has been successfully crea
Davin Granroth
Self-signed root certificate
Expires: Tuesday, December 25,
AM ET
This root certificate is not trus
Details
Subject Name
Common Name Davin Granroth
State/Province MI
Country US
Locality Saint Charles
Now if you could only trust that certificate.
Trusting your certificate
If you haven't already, click the Done button to close that Certificate Assistant window. Now, back in Keychain Access, click on the My Certificates category on the right of the main Keychain Access window.
You'll see your new certificate listed with a little white X in a red circle

on the icon. That indicates the certificate is not trusted. Double-click on the certificate, and a new window will open with details of the certificate.

Near the top of that window you'll notice the word *Trust* with a little triangle to the left of it. Click the triangle to twist open the Trust options.

In the *When using this certificate* select list, select *Always Trust*. Then close that window. You'll be prompted for your administrator

password. Enter it, and you should be all set. Your new certificate should now be trusted.

Sending signed or encrypted e-mails

At this point, if you restart Apple Mail, you'll notice a new option available when you compose a message.