The Joys of Importing & Using an S/MIME Certificate

CCIT Publication

Table of Contents

The Joys of Importing & Using an S/MIME Certificate

S/MIME (Secure Multipurpose Internet Mail Extensions) is a widely supported standard used to secure email. It is client-based and allows email messages to be digitally-signed and encrypted.

Message encryption is used to preserve the confidentiality of the exchange – only intended recipients should be able to decrypt and view the content. This gives the ability to send sensitive information through email – like passwords, PII (Personally Identifiable Information), payroll, HR, or institutional data that is not for public release etc. – in a secure fashion.

Digital-signatures are a way of assuring the integrity of the received message: that it has come from the individual who sent it with non-modified content. This way a message bearing a valid signature from Derek Wilson, Phil Romig III, or any other CCIT staff member, can be viewed as not being some form of email spam spoofing the From: address or modifying the original messages for malevolent purposes.

Selection has been made of an S/MIME provider for, initially, all CCIT staff members: Comodo. Longer-term: expanding these secure-email certificates across to other selected school departments or out to a wider campus audience is to be considered.

Comodo has trusted root authority certificates (under 'Comodo CA Limited' in the certificate store) in all of the major operating systems. Looking at the signing hierarchy Comodo uses for an S/MIME certificate: AddTrust CA root (again, trusted in all major OSs) signs UTN-USERFirst (Client Authentication and Email certificate) which signs the underlying Comodo Client Authentication and Secure Email CA. Finally, this then signs the individual S/MIME entities bound to each individual. Both AddTrust AB (Sweden) and UTN-USERFirst (USA) have established and trusted relationships with Comodo. There are valid security concerns related to such hierarchies of trust existing amongst root Certificate Authorities and selected resellers or business partners – but, realistically, this must be managed as an appropriate risk. In the past, exploitation of these relationships has allowed attackers to target “weaker” links to abuse the trust inherent to issued rogue certificates.

The main body of this document will cover the steps needed to obtain and install your issued S/MIME certificate. The systems covered include: Windows with Outlook, Windows with OWA access via Internet Explorer, OSX with Apple Mail, OSX with Outlook, iOS with Apple Mail, Android with default email reader application, and Mozilla Thunderbird cross-platform. If use is being made of other email clients or operating systems, extra research might be needed to correctly configure everything.

Requesting and backing-up your S/MIME certificate

Per Figure 1, an email message will arrive from 'Comodo Security Services <noreply_support@comodo.com>'. This will have 'Applying for your Corporate Secure Email Certificate' as the subject. Initially, this message was being flagged as spam, but this has been resolved.

Either click the button if visible or, preferably, copy the link location (https://secure.comodo.com/products/CorporateSecureEmail) and paste this into your browser. This gives more assurance that the location being visited is truly https://secure.comodo.com and not elsewhere in the event of a targeted malicious message.

Through the application process you will need the displayed 'Certificate Password'.

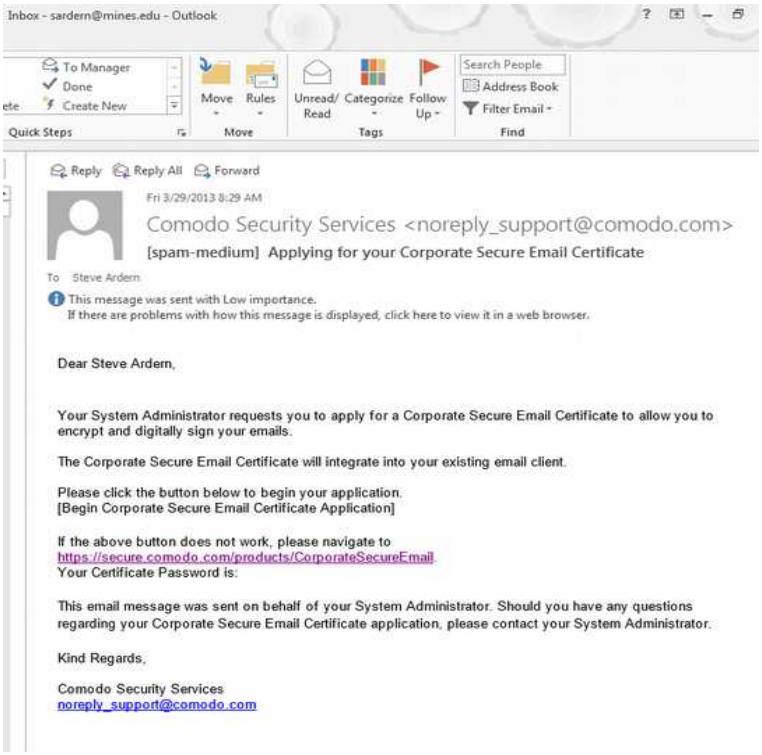

Figure 1: Comodo's application message for S/MIME certificate

Browsers being discussed, in the following section, for performing the steps of requesting, initial installation, and making backups of your S/MIME certificate include: Mozilla Firefox, Internet Explorer, Google Chrome, and Safari.

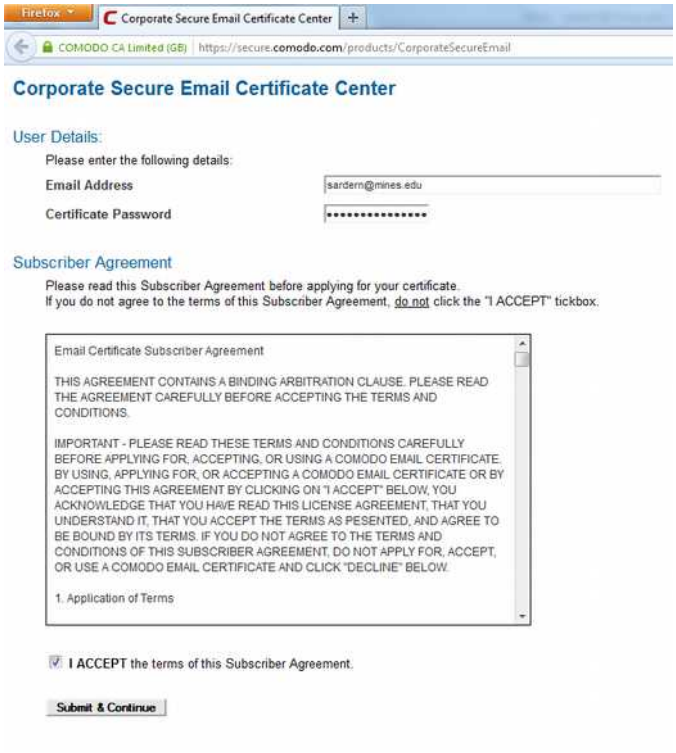

Figure 2: S/MIME certificate application page

Having visited Comodo's S/MIME application page, as in Figure 2, please input your email address and your certificate password, as given in the original message. This is the only place where this password is needed – later on in the process you use your own choice of password for protecting this certificate. Double-check that the browser padlock and security information shows in green and is for 'COMODO CA Limited (GB)'. Read through the subscriber agreement and check the 'I ACCEPT the terms of this Subscriber Agreement' before clicking through 'Submit & Continue'.

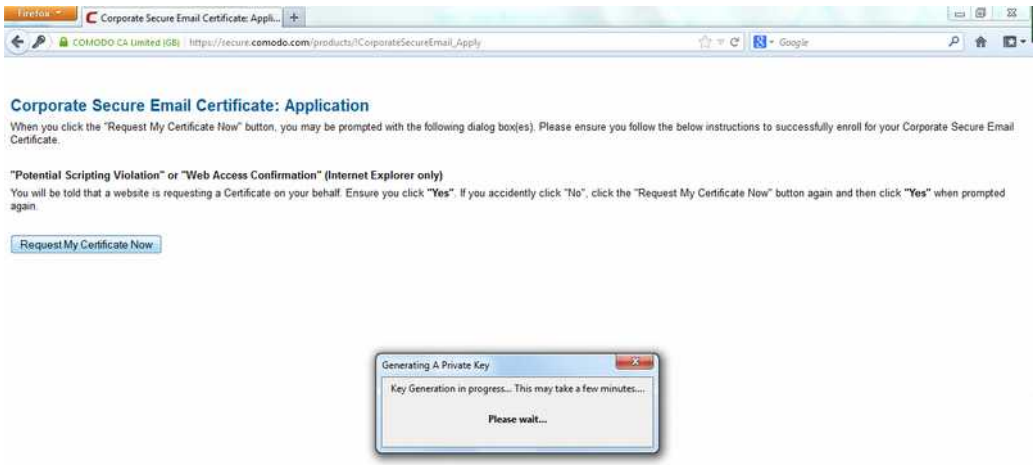

Figure 3: Request S/MIME certificate

Next you should hit the page shown in Figure 3. Read through the warning and click through

'Request My Certificate Now'. A popup box related to private key generation should show. Once this completes you should see something akin to Figure 4 (depending on browser being used).

Figure 4: Personal certificate has been installed

At this point your certificate has been installed into the associated certificate store. Per the instructions, you should make a backup copy.

Mozilla Firefox

For the Mozilla Firefox browser reference Figure 5. Select 'Options' → 'Advanced' tab → 'Encryption' sub-tab. Click on 'View Certificates' then select 'Your Certificates' tab. Select the 'COMODO CA Limited' entry – it should show your name – and click the 'Backup' button. Save the file somewhere appropriate. Next (Figure 6), input your password. It is important to use a memorable password as this is how you, firstly, protect the contained private-key (this must be known only to you and no-one else), as well as being needed when importing this certificate object onto other systems you use.

Having completed these steps you should see a confirmation dialog telling you that everything has been backed-up (Figure 7). Save the resulting file somewhere safe – our backed-up network drives would be preferable. Remember: your private-key is protected by your selected password.

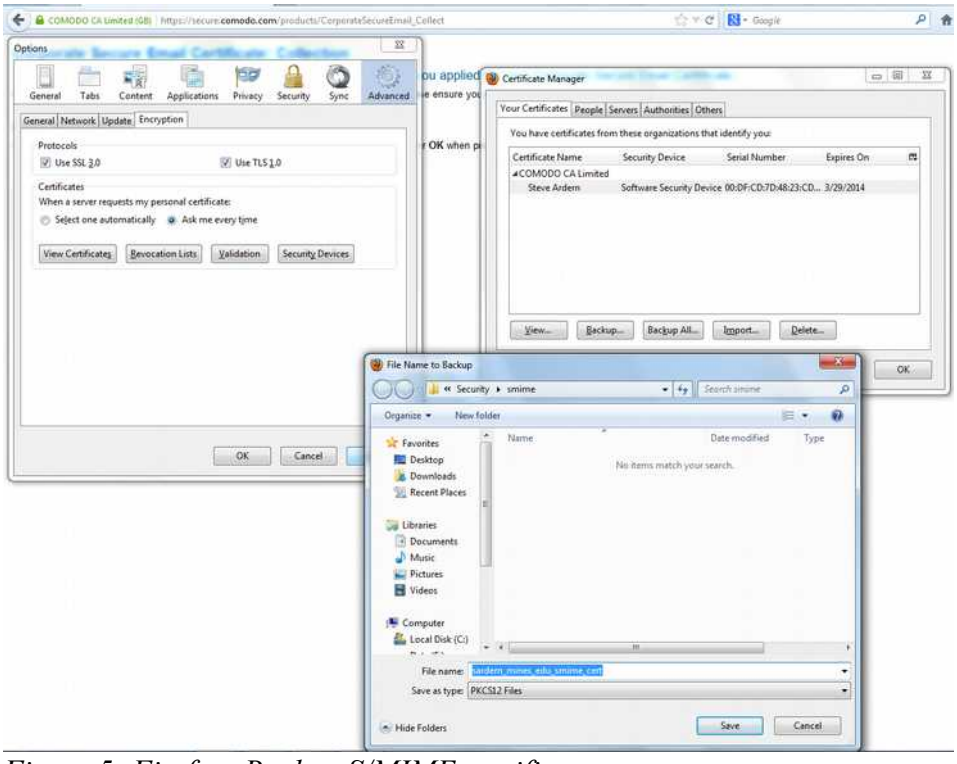

Figure 5: Firefox: Backup S/MIME certificate

Figure 6: Firefox: Choose a certificate backup password

Figure 7: Firefox: Successfully backed up security certificate

Internet Explorer

See Figure 8 for Internet Explorer: 'Internet Options' → 'Content' tab → 'Certificates' → 'Personal' tab → 'Export' button.

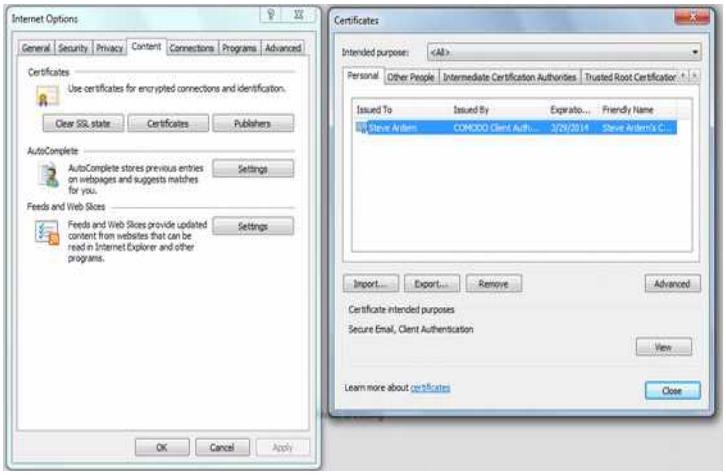

Figure 8: Internet Explorer: Certificate backup

Google Chrome

For Google's Chrome browser you should be looking at the client certificate informational message shown in Figure 9. Click 'View' to be able to start the backup process.

Figure 9: Chrome: 'View' button

Switch from the 'General' tab, showing certificate information, to the 'Details' tab. Click on 'Copy to File...' to start the Certificate Export Wizard (Figure 10).

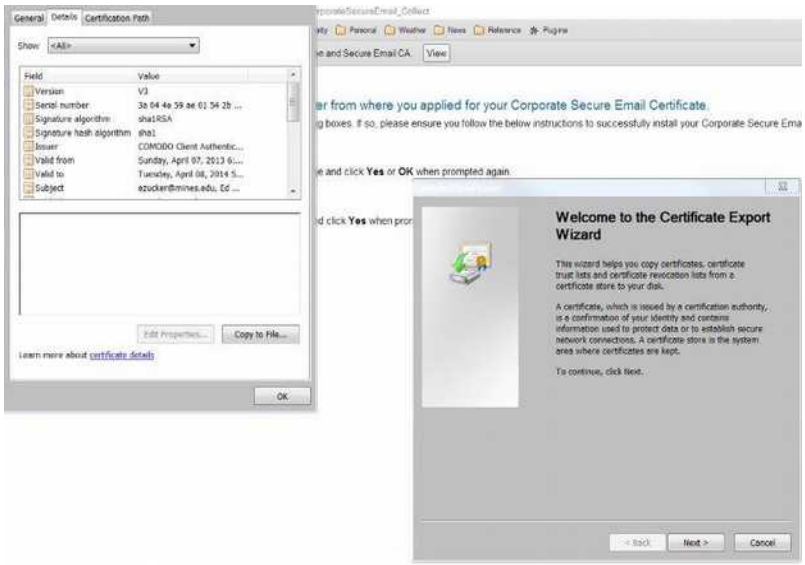

Figure 10: chrome: Details → certificate export wizard

Through the wizard it will ask whether you want to export the private-key: answer 'Yes, export the private key' (Figure 11).

Save the file using 'Personal Information Exchange – PKCS #12 (.PFX)', selecting 'Include all certificates in the certification path if possible' and 'Export all extended properties'. Ensure that the middle option, 'Delete the private key if export is successful' is NOT checked (Figure 12).

Input your backup password and then the resulting file should be stored somewhere safe – a backed-up network drive would be preferable.

It seems that chrome – certainly on Windows – appears to save the backup file, by default, to a somewhat out-of-the-way disk location. Look inside somewhere similar to the following and then copy this file to somewhere on our network drives:

```
C:\Users\<login>\AppData\Local\Google\Chrome\Application\<version>\
```

Figure 11: Chrome: export private key

Figure 12: Chrome: export as PFX file

Safari

Using Safari on Mac, the certificate should be imported directly into your keychain. To make a backup, open the 'Keychain Access' application, ensure you are in the 'login' keychain, select 'My Certificates', highlight your new Comodo S/MIME certificate – it should be titled with your name – and select 'Export'. Make sure you export to file format 'Personal Information Exchange (.p12)' (Figure 13).

Figure 13: Mac keychain certificate export

Which browser you used for this step may determine which of the remaining sections of this document applies. For example, if you used Microsoft's Internet Explorer and use Outlook as your email client, then the specific instructions for manually importing the certificate into Outlook will probably not be needed. This is due to the fact that Outlook references Microsoft's certificate store, wherein the certificate will now already be present.

Having performed the above steps, you should have received another message from Comodo to confirm that your certificate issuance has been completed (Figure 14).

Fri 3/29/2013 9:37 AM

Comodo Security Services <noreply_support@comodo.com>

Collecting your Corporate Secure Email Certificate

To Steve Ardern

If there are problems with how this message is displayed, click here to view it in a web browser.

Dear Steve Ardern,

Your Corporate Secure Email Certificate has now been issued and is ready to be collected.

Please click the button below to begin collection.
[Begin Corporate Secure Email Certificate Collection]

If the above button does not work, please navigate to
https://secure.comodo.com/products/CorporateSecureEmail.
Your Certificate Password is:

This email message was sent on behalf of your System Administrator. Should you have any questions regarding your Corporate Secure Email Certificate application, please contact your System Administrator.

Kind Regards,

Comodo Security Services
noreply_support@comodo.com

Figure 14: Comodo S/MIME certificate issued message

Interlude: Encryption Hurdles

Having obtained your S/MIME certificate and then installed and configured everything for your email clients per the instructions below, mention must be made about some encryption hurdles.

Firstly, this discussion is not about being unable to send digitally-signed messages. You will be able to send a signed message to anyone – because this is based on the private-key you control and have just installed. Signing messages is for you to declare, “I am me and have just sent this message to you.” No problems there. In fact we should definitely have signing set as a default action for all messages we send – this will be covered in the instructions below.

However, the first-time you try sending encrypted messages to someone, even potentially yourself depending on which client you use, there is a good chance that your email client will complain and not be able to encrypt your message. The item it complains about is related to it not possessing the recipient(s) certificate(s). How do you get other peoples' certificates? Most email clients will collect them for you transparently – in that whenever you read a message if it has an embedded S/MIME certificate then that is stored and you can encrypt messages to that user.

Where you are testing against yourself, you can have your own certificate be collected for you (into the right place, per your client) by sending yourself a signed message. Read this message – ensuring it shows the 'signed' icon – and you should now be able to send encrypted messages back to yourself.

As a department CCIT has around 50 employees. Therefore being able to communicate amongst ourselves and being able to securely encrypt messages containing sensitive information is a part of what we can do because of our S/MIME certificates. However, due to the above caveats and the fact that the majority of our messages really don't need to be encrypted this is optional for each of us.

We should, probably, not select encryption as a default action – the steps for doing this is discussed below. Then on an individual basis, we can decide which messages we want or need to encrypt. An example: I need to send Phil a list of passwords – this will most definitely be encrypted. I ensure that I possess Phil's certificate – if I don't have it then I send Phil a signed message asking for his reply. Upon completing this transaction – we can now encrypt our messages to each other.

Following this kind of usage model, we have the setup where the people we most commonly communicate with are the ones where we can encrypt if we need to.

As will be discussed below in the Outlook on Windows section: a way we can publish our certificates within our Active Directory environment (to the Global Address List (GAL).) This gives people who use supported clients – like Outlook, most Apple products, as well as others – a way to be able to interrogate ADIT for the recipients S/MIME certificates. This level of centralization will bring benefits including not necessarily needing to have a users certificate ahead of being able to send them encrypted messages.

Configuring S/MIME certificate with Microsoft Outlook

Options → 'Trust Center' → 'Trust Center Settings...' (Figure 15). In earlier versions of Outlook this is 'Tools' → 'Options' → 'Security'.

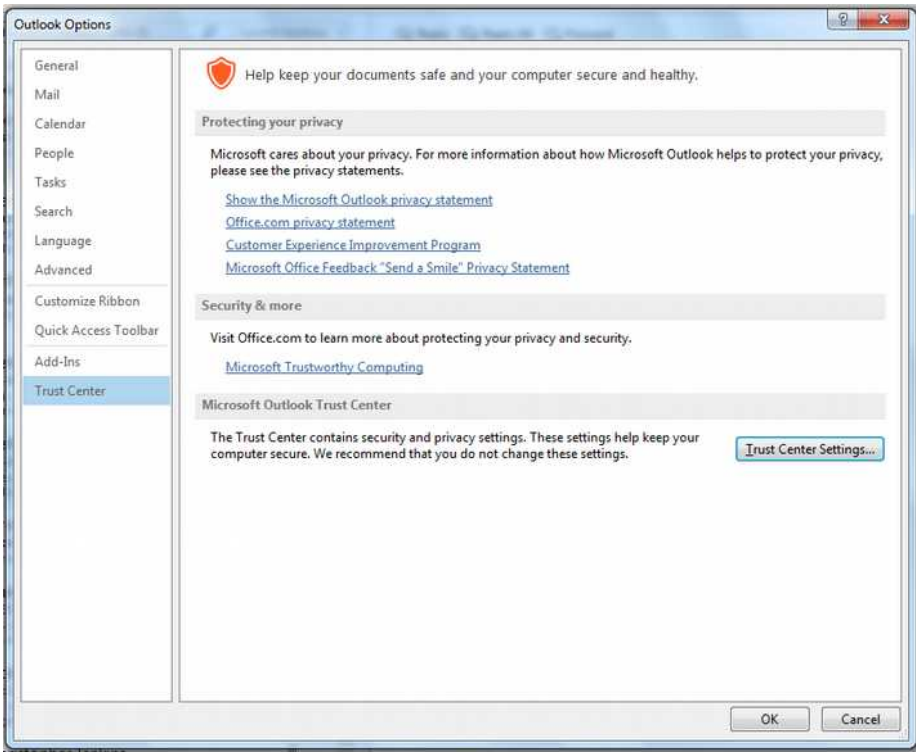

Figure 15: Outlook Options: Trust Center Settings...

From the 'Trust Center' screen select 'E-mail Security' on the left-hand side and click the 'Import/Export...' button. Select the saved certificate file. Input the password you used to protect this file. Hit OK. (Figure 16). In some versions of Outlook you may also need to input a value into the 'Digital ID' field: for this use your name.

Upon importing you will go through the 'Importing a new private exchange key' wizard (Figure 17). First you need to select a security level (Figure 18), picking between Medium (request permission when using) and High (request permission with a password when using). If a high security level is selected then provide a password (Figure 19) – this can be different to the import-password used with the certificate file.

Back at the 'E-mail Security' page, verify the certificate is selected by clicking on the 'Settings...' button. The displayed settings should be acceptable (Figure 20) and do not need to be modified – hash algorithm: SHA1, encryption algorithm: AES (256-bit). Close out this dialog.

Finally, ensure that in the uppermost section called 'Encrypted e-mail' the second two check-boxes should be enabled (Figure 21). These turn on default digital-signatures using your S/MIME certificate. The first checkbox is related to default encryption that has already been discussed in the section, "Interlude: Encryption Hurdles". Also, as mentioned in the aforementioned section, click the 'Publish to GAL...' button to send your certificate information up to our ADIT environment. This should benefit the users who do this when you need to send an encrypted message to another GAL-published CCIT staff member.

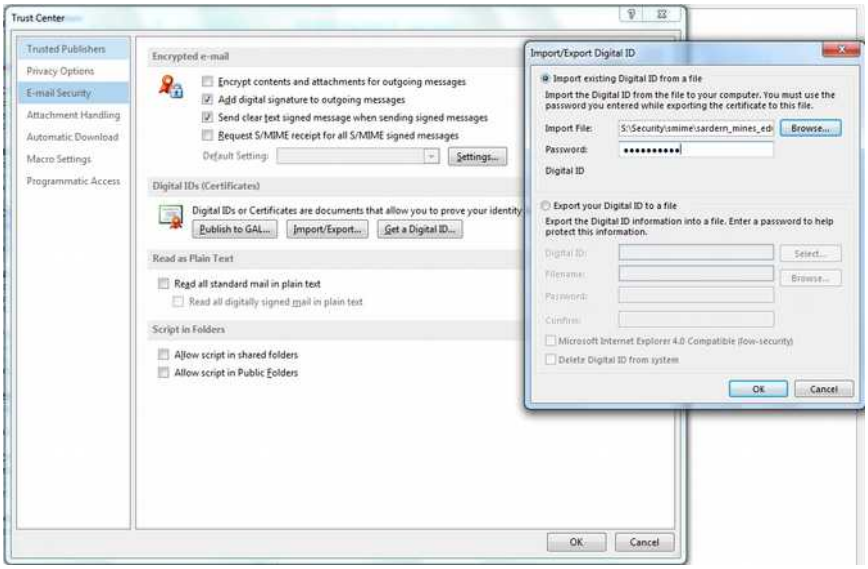

Figure 16: Outlook: certificate import

Figure 17: Outlook: importing a new private exchange key

Figure 18: Outlook: Choose a Security Level

Figure 19: Outlook: High Security Level password

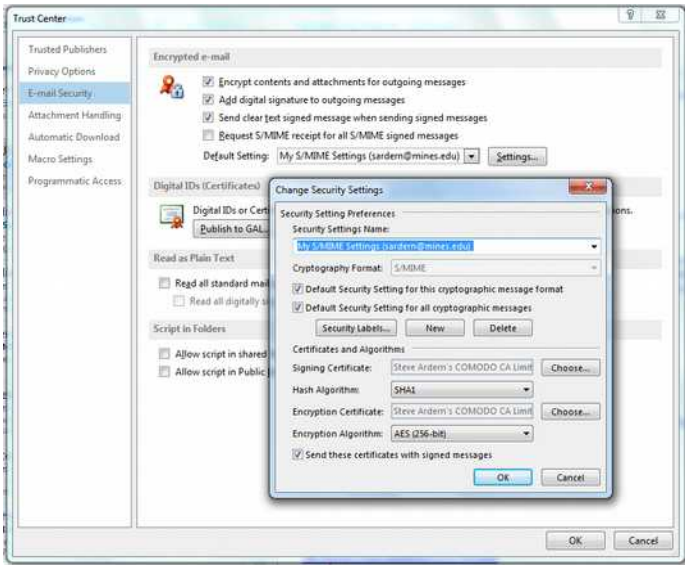

Figure 20: Outlook: Finalizing settings

Figure 21: Outlook: select default signing

To test these settings: firstly restart Outlook. Compose a new message and send it to yourself. The defaults should kick in and digital-signatures should be in-place (Figure 22).

Having sent the message to yourself, you should now observe the ribbon meaning the message is signed (Figure 23). (The padlock icon, as shown, will be present if the message is also encrypted – however, please read the section, "Interlude: Encryption Hurdles", about likely complications with

performing encryption.)

In Outlook, upon first using your S/MIME certificate, you will be explicitly asked for permission, per the security level you selected earlier. It will, at least, ask to be granted access – along with requiring a password at the higher security level. This seems to happen only once per session, although it might have some associated timeout value that has not been identified through the testing performed.

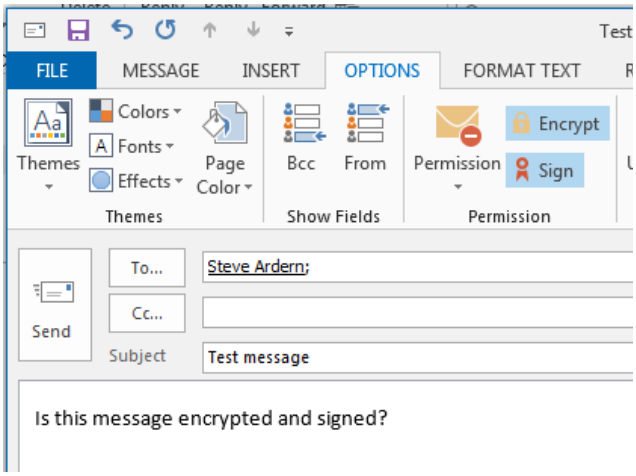

Figure 22: Outlook: sending an encrypted and signed message

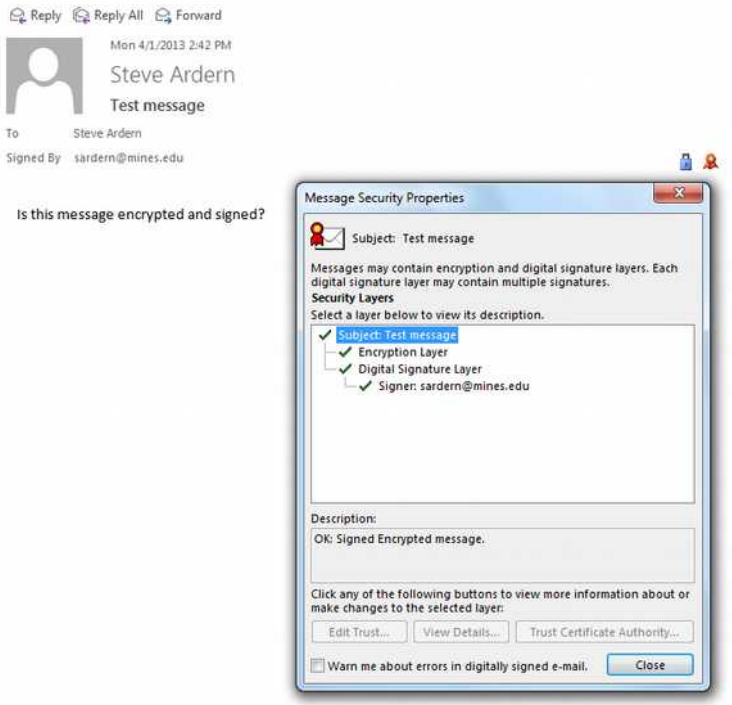

Figure 23: Outlook: message is encrypted and signed

Installing S/MIME plugin into OWA

Using Microsoft Exchange's web-interface, OWA (Outlook Web-App), does support S/MIME certificates – but only through Internet Explorer. There is an issue with pure 64-bit IE versions – but, most of the time, Windows should have both 32-bit and 64-bit versions available.

Please note that if you have imported your S/MIME certificate into Microsoft's certificate store then Internet Explorer should be aware of your certificate to use through OWA.

Login to OWA and select Options. Click on Settings and go to the S/MIME tab (Figure24). Click on the 'Download the S/MIME control' link.

Figure 24: Internet Explorer OWA S/MIME control

Once the control is installed, check that the 'Add a digital signature to all messages I send' checkbox is selected (Figure 25). The default encryption option is optional. Click Save.

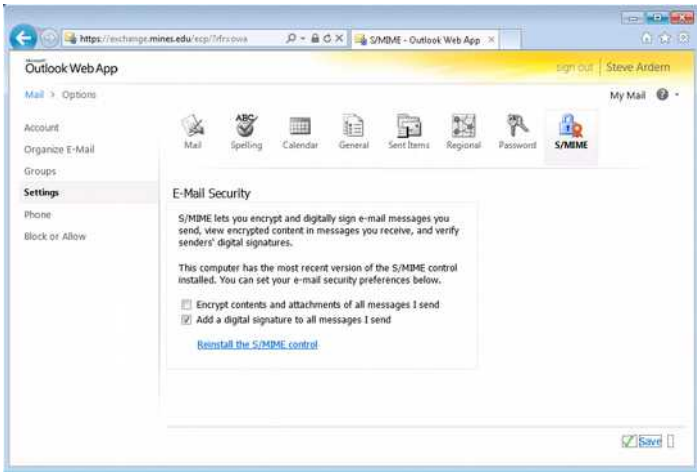

Figure 25: OWA S/MIME plugin options

Configuring S/MIME certificate for Apple Mail on Mac OSX

Installing your S/MIME certificate is as easy as double-clicking the file. You will be asked to confirm which keychain you want to add it to – accept the defaults (keychain: login) (Figure 26). You will need to input the certificate password (Figure 27) before your S/MIME certificate installs itself into the keychain (Figure 28).

Figure 26: OSX: add certificate

Figure 27: OSX: enter certificate password

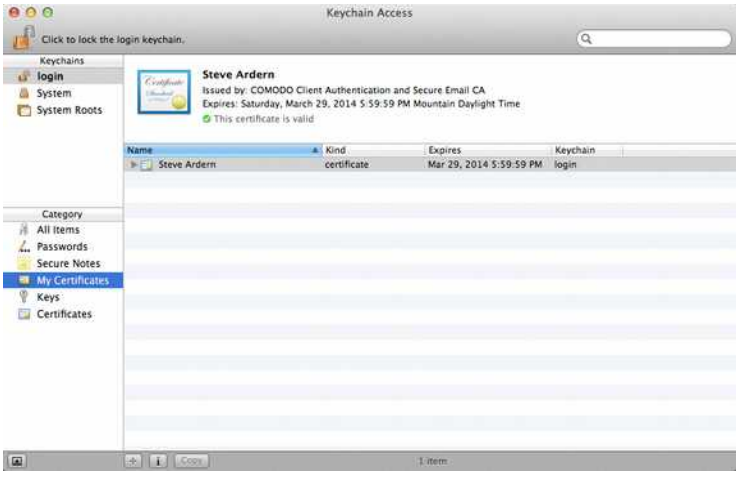

Figure 28: OSX: certificate installed into keychain

Apple Mail may need to be restarted at this point. It will automatically pick up on the S/MIME certificate being present. Test it by sending a message to yourself – notice the padlock and the ribbon icons (Figure 29). If the padlock is sealed then the message will be encrypted, if the ribbon

has a tick inside then the message will be digitally signed.

You may need to play with the level of header visibility to see the 'Security:' header information. In this example it tells you that this message is both encrypted and signed (Figure 30).

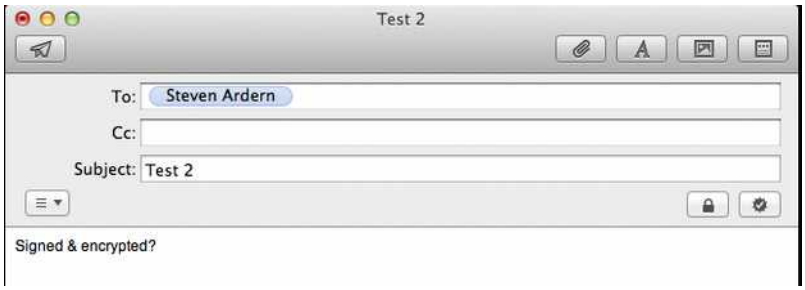

Figure 29: OSX: Creating a signed & encrypted message

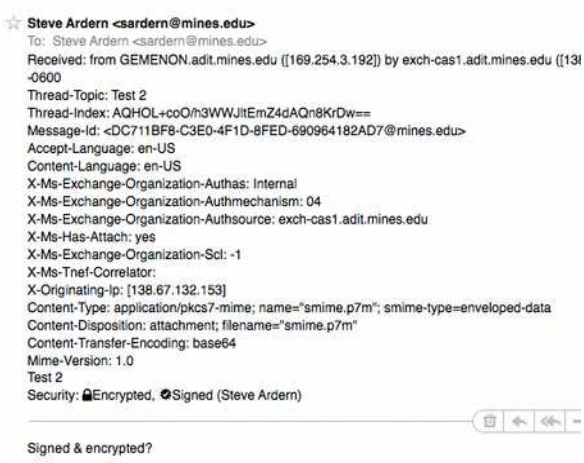

Figure 30: OSX: header showing Security: details

OSX will ask for permission to access your S/MIME certificate private-key (Figure 31) – whether you allow it each time, deny (which will mean your S/MIME signing and decryption capabilities will not function), or select 'Always Allow' is dependent on how you use your Mac. If you are the only user then it should be fine selecting 'Always Allow'.

Figure 31: OSX: allow access to confidential information

(See this interesting note, if you have multiple S/MIME certificates installed on your Mac.)

Configuring S/MIME certificate for Outlook on Mac OSX

Having installed the S/MIME certificate into your keychain by double-clicking on the file (more details here) – Outlook on Mac should be fairly easy to configure. You may need to restart Outlook after completing the certificate importation into your keychain step.

In Outlook for Mac: 'Tools' → 'Accounts...' → 'Advanced'. Go to 'Security' tab and make sure your certificate is selected in the 'Certificate:' field. Ensure that the three checkboxes are all selected in the uppermost 'Digital Signing' section (Figure 32). Encryption is optional, remembering the caveats expressed in the section, "Interlude: Encryption Hurdles".

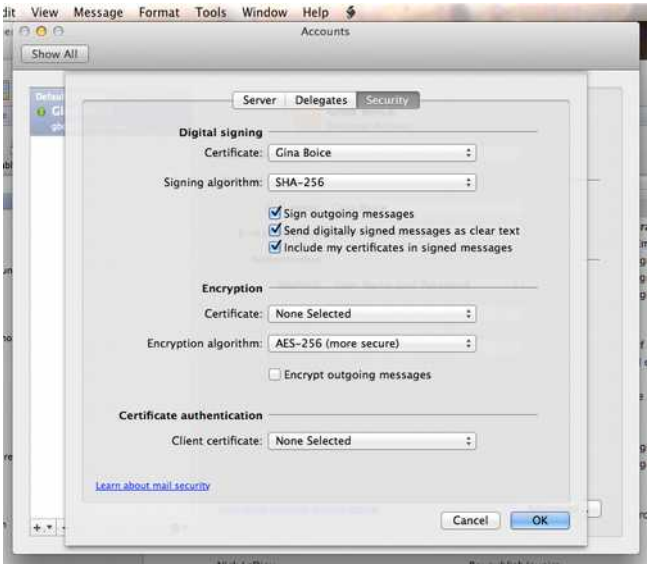

Figure 32: Outlook on Mac: setting up certificate

Please note: select your own, freshly installed, certificate in the 'Encryption' → 'Certificate' section shown in the middle of Figure 32. The reason for doing this is because, otherwise, when you send an encrypted message to someone Outlook does not know how to encrypt it for your own local "Sent Mail" folder. It does alert you multiple times, but the message is not instantly obvious (you may, for example, think that Outlook is moaning about not possessing the recipient's certificate, whereas this is, specifically, concerned with your own certificate).

Essentially, if you proceed through the alert messages, Outlook encrypts the message to the recipient, using their certificate, just fine. However, Outlook cannot encrypt your local copy, therefore it will be forever non-accessible to you sitting in your 'Sent Mail' folder. In almost every scenario this will not be what is intended.

Configuring S/MIME certificate for Apple Mail on iOS

These instructions are for setting up S/MIME on Apple iOS devices, like iPads and iPhones. This testing was done on the latest iOS 6.x (6.1.3, at this time) running on an iPad – it may well be a little different with alternative setups. From the research performed it looks like iOS < 5.0 does NOT support S/MIME – at least not "natively". Successful testing has also been completed on iPhones.

The easiest way to configure is to send your S/MIME file to yourself via email, then hop onto your iOS device and bring up the Apple Mail application. Your file should be in a PKCS #12 format – easiest check is to see if the last part of the file is p12, e.g. my_smime_file.p12.

Click on this attachment, it should bring up the 'Install Profile' window showing your 'Identity Certificate'. Click on 'Install' → 'Install Now'. Enter firstly your iPad or iPhone access code, followed by the password you used to backup your certificate with. You will be presented with the 'Profile Installed' screen. Click 'Done' to exit.

Now to configure S/MIME: go into 'Settings' → 'Mail, Contacts, Calendars'. Click on your Mines' Exchange account. Click through the 'Account' button, which should be the first button you see at the top. Scroll down and at the bottom you should see an S/MIME slider – turn this to on. Also enable the 'Sign' action. For encryption, remember the caveats expressed in the section, "Interlude: Encryption Hurdles".

Upon selecting 'Sign', you will go into a sub-section that shows you which certificate will be used for signing your messages. Unless you have more than one personal certificate installed, this should be the one you have just imported.

Interesting note: if multiple S/MIME certificates are installed

If you have multiple S/MIME certificates installed on your Apple product and wish to ensure the right associations are in-place, i.e. this S/MIME certificate is associated to your @mines.edu email address. Bring up 'Keychain Access' application and right-click on the appropriate certificate. Select 'New Identity Preference...' and type in the right email address to associate with.

Configuring S/MIME certificate for Mail Reader on Android

The following steps were successful in configuring S/MIME on an Android device. However, given the somewhat fragmented ecosystem, i.e. differences between Samsung versus HTC versus whoever else brands, different versions of the Android OS and vendors' customizations etc. you may need to use different steps to get everything working on your specific Android device.

Steps, using the “native” Mail Reader application:

1. Copy the certificate file to somewhere on the phones internal memory. We did encounter issues when we initially copied the certificate file out to the external SD card. Again, your mileage may well vary depending on which phone you have.
2. Open Mail Reader application and select your Mines account.
3. Select 'Options' → 'More' → 'Account Settings' → 'Security Options' → 'Email Certificate'
4. Select 'Install' then select the certificate file.
5. To unlock, input your certificate password.
6. Optionally rename what your phone tags your certificate with.
7. Select 'Sign all outgoing mail'.
8. Optional choice for 'Encrypt all outgoing mail' (read section “Interlude: Encryption Hurdles”).
9. Accept all other defaults.
10. Test – your phone should now be sending signed (and, optionally, encrypted) emails.

Configuring S/MIME certificate with Mozilla Thunderbird

Go to 'Edit' → 'Account Settings', then select 'Security' (Figure 33).

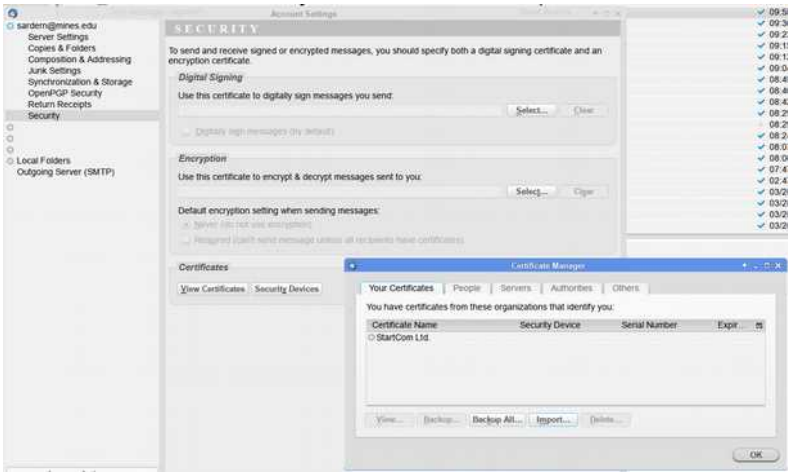

Figure 33: Thunderbird: importing S/MIME certificate

Click on 'View Certificates' → 'Import...'. Select the certificate file and input the password to unlock. It should display the certificates information (Figure 34), check through and click 'OK'. It should now display both a 'Successfully restored your security certificate(s) and private key(s)' message (Figure 35), as well as prompting you about enabling encryption as a default action (Figure 36). (Please recall the caveats regarding encryption discussed in the section, "Interlude: Encryption Hurdles".)

Your final initial configuration should be as in Figure 37.

Testing against yourself, you should see both the wax-sealed envelope (signed) and the (optional) padlock (encrypted) icons showing that Thunderbird is working correctly with your S/MIME certificate (Figure 38).

Figure 34: Thunderbird: imported certificate information

Figure 35: Thunderbird: Successful import

Figure 36: Thunderbird: default encryption action

Figure 37: Thunderbird: initial security settings

Figure 38: Thunderbird: signed and encrypted message

Addendum: Publishing certificate fingerprints

It is our intention to collect together the issued certificate fingerprints for each of us and centrally publish these values somewhere. This will give any of us the ability to check the actual fingerprint against those published – any discrepancies should be investigated appropriately. This way we have another layer of being able to trust the fact that, for example, Phil Romig III's email message is really from Phil Romig III and not a pseudo-Phil-spambot coming out of the darkened underbelly of the internet!

The certificate signature values – as issued – are available to CCIT's security team through the administration console we have through our Comodo account (Figure 39). This way collection can occur from this single point-of-issuance rather than needing each of us to communicate this information in a more manual fashion. After-the-fact verification should still occur, however, to ensure everything is valid, i.e. check that the published certificate signature matches your actual certificate signature.

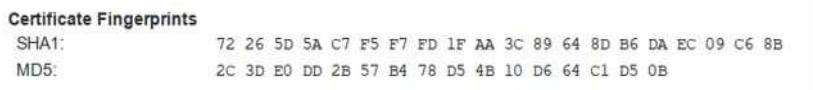

Figure 39: certificate fingerprint information available through Comodo's admin console