Configuring Outlook's built-in S/MIME Cryptosystem (Mac & Windows)

December 4, 2017

Contents

1. Prerequisites

- MS Outlook 2013 or later is recommended. Earlier versions also officially support S/MIME but users often report difficulties, particularly with Outlook 2010.

- A browser is only needed initially to create SSL keys, and only if you are going to use a Certificate Authority ("CA"). The browser will not be used thereafter. Any of these browsers will work:

Browser	***Key storage consistent with Outlook***	***Notes***
Chrome/Chromium (OS/X)	yes	Some versions of Chrome may have problems with key generation.
Chrome/Chromium (Windows)	no (browser uses its own internal key store)	Works but needs some extra steps (section 3.2) to copy keys.
Firefox (all platforms)	no (browser uses its own internal key store)	
~~Internet Explorer (OS/X)~~	–	Don't use IE on Mac; latest version (4.0) is discontinued.
Internet Explorer (Windows)	yes	
Safari (OS/X)	yes (Safari automatically installs S/MIME keys on a "Keychain Access"-reachable keyring, which Outlook uses according to Comodo docs and MS docs.	Some versions of Safari may have problems with key generation.
Safari (Windows)	? (undocumented, left as an exercise for the readers ;])	

Browsers above that are indicated "yes" for sharing the same key storage as Outlook are more convenient for this setup because keys need not be copied, thus section 3.2 may be skipped.

2. Instructional videos (supplemental)

These videos are optional; not required by this guide.

Link	***Outlook ver.***	***Browser ver.***	***CA***	***Scope + Notes***
▶	2007	n/a	n/a	
▶	2010	Firefox	Symantec	Demonstrates how to use Firefox and manually copy the key pair into Outlook.
▶	2013	Chrome	Comodo	
▶	2013-2016	IE?	Comodo	This comprehensive video covers every step in this entire document. It demonstrates a case where the browser automatically installs the key where Outlook can find it. The first 9 min. of the video is blather, but the link supplied skips to the relevant part.
▶	2016	IE?	Entrust	Outlook had automatic visibility to the key in this demo on Windows, so IE was likely used.

3. Prep to receive encrypted mail or to send signed mail

3.1. Get an S/MIME certificate

You have a choice in whether you want to subscribe to a Certificate Authority ("CA") or whether you want to be your own CA and generate your keys manually. The advantage of using a CA is that the recipient has less key management effort (this doesn't matter if Justin is your recipient). But CA subscriptions cost money in some situations and also limit the validity period of the key.

If you want to create your own key using Windows, install OpenSSL for Windows. Mac users will already have openssl installed. Then follow "Create a Certificate Authority to Sign A Certificate" to create your own CA and personal key pairs.

If you're opting to use a CA, then browse to one of these certificate authorities (Justin recommends Comodo via Secorio):

CA	*price* *(for non-commercial individual use)*	*validity*	*notes*
CAcert	gratis	6|24 mos.(criteria)	community driven; getting a 2yr cert requires meeting w/someone and showing state-issued proof of id
COMODO direct	gratis	1 yr	
COMODO via InstantSSL	gratis	1 yr	
COMODO via Secorio	gratis	1 yr	simple; assumed choice by this guide
Entrust	≥$20		
IdenTrust	≥$19		
StartCom®	gratis	2 yrs	distrusted by Mozilla and others, thus signed msgs will likely be seen as invalid unless recipients manually add Startcom's CA key to their keystore
~~WoSign~~	~~gratis~~ n/a	~~2 yrs~~ n/a	recently **discontinued** but still maintained in this list because they intend to return to business

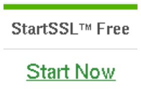
Warning: the CAs that participate in e-mail certificate verification are constantly changing. Many CAs have discontinued e-mail certification prior to this guide. Those with no intent to return to service are omitted here, but some of the above listings are likely to become obsolete as this guide ages. Consequently it might be interesting to check out the catalog of certificate authorities listed at `http://kb.mozillazine.org/Getting_an_SMIME_certificate`).

3.1.1. If you chose "Comodo via Secorio"

1. (secorio.com) If you are using the *noscript* firefox plugin, you must enable javascript for `secorio.com` and `comodo.com`.
2. (secorio.com) In the left frame, select "`S/MIME Class 2`" (even though it's *class 1* that we need), then click "`Order`".
3. (secorio.com) Scroll down to "`S/MIME Certificates`" and choose "1 year" in the pull-down to the right of the *class 1* row.
4. (comodo.com) Fill out the form that appears in a new tab. Setting a revocation password is optional (and it's a good idea).
5. (your inbox) An e-mail will arrive. If your e-mail client renders it graphically, click the button "`Click & Install Comodo Email Certificate`". For text clients, follow the instructions in the e-mail. If your mail client does not automatically use Firefox or IE to open URLs, right-click that button instead, copy the URL, and paste it in the address bar to force it to render in Firefox or IE.
6. Skip to section 3.2

3.1.2. If you chose "StartCom"

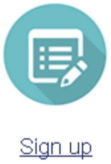
1. (startcomca.com)



StartSSL™ Free

Start Now

2. (startcomca.com)



Sign up

3. (startcomca.com) Fill out the form.
4. (your e-mail account) A verification code will arrive.
5. (startcomca.com) Enter the verification code, click "sign up".

   The system could not install the login certificate automatically, you should install it manually. Create a "Private key password" Please click here to install the issuing CA certificate into your browser first. save a *.p7b file
6. Skip to section 3.2

3.1.3. If you chose another certificate authority

Simply follow the instructions on the website of the CA. It will generally involve filling out a form and confirming an e-mail.

3.2. Installing your certificate into Outlook

If you created your key using a browser that uses the same key storage as Outlook (as indicated in section 1 Prerequisites), skip to subsection 3.3 Configuring Outlook. If you generated your key pair manually, skip to subsubsection 3.2.3 Import your key into Outlook.

3.2.1. (Chrome only) Export your key from your browser

There are multiple browsers which do not share the same key storage as Outlook, but this section presumes you are using Chrome. Follow YouTube video n3rOEpGjrc (the link of which jumps you to the relevant point in the video), or follow these steps:

1. After creating the certificate (previous section) Chrome presents a status bar under the address bar saying “Successfully stored client certificate..”. Click the “`View`” button on that bar.
2. There is a pop-up which showing basic information about your certificate. Switch to the `Details` tab.
3. Click the “`Copy to File`” button to launch an export wizard.
4. Click `Next`.
5. You are asked if you want to export the private key with the certificate. The default answer is no, but you need to click “`yes`”, then `Next`.
6. You are asked which format to use. Choose `PKCS#12`, then `Next`.
7. You will be prompted for a password for the backup file. A weak password is fine, because this backup file will not be transmitted or retained. You will import it into Outlook locally, and then you will delete the backup file.

Then skip to section 3.2.3 for steps to import the key into Outlook.

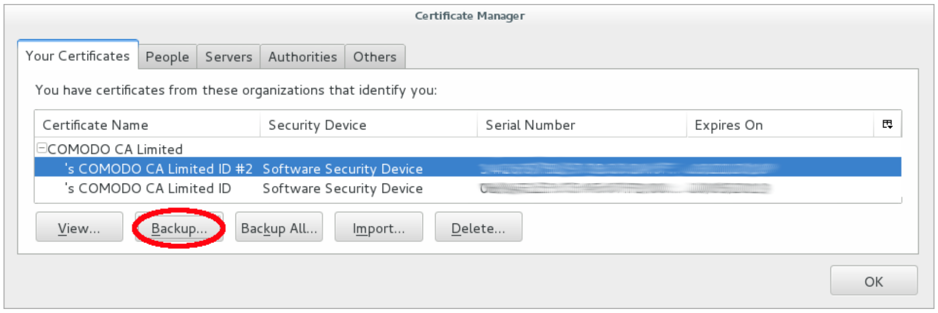
3.2.2. (Firefox only) Export your key from your browser

There are multiple browsers which do not share the same key storage as Outlook, but this section presumes you are using Firefox. Watch YouTube video wGHaB0elkaA or follow the steps below to export the key from the Firefox and then import it into Outlook.

The video demonstrates using Outlook 2010 for the mail client and Firefox (a pre-2012 version) for the browser. The first few minutes start by walking through key creation on Symantec’s website, which is useless because Symantec no longer offers the service. So the YouTube link skips that portion of the video automatically.

If not following the above-mentioned video, these are the steps to export from a more recent version of Firefox:

1. Go to: menu (≡) ≫ Options/Preferences ≫ Advanced ≫ Certificates ≫ View Certificates ≫ Your Certificates.



You have certificates from these organizations that identify you:

Certificate Name	Security Device	Serial Number	Expires On
COMODO CA Limited			
's COMODO CA Limited ID #2	Software Security Device	[redacted]	[redacted]
's COMODO CA Limited ID	Software Security Device	[redacted]	[redacted]

2. Highlight the line showing your new key. It will be under the name of the CA you chose (e.g. the line under "COMODO CA Limited" if you chose Comodo).

3. Click "`Backup...`" to export the key.

4. Save the file somewhere with a filename of your choice. It will likely be given a `.p12` extension.

5. You will be prompted for a password for the backup file. A weak password is fine, because this backup file will not be transmitted or retained for long. You will import it into Outlook locally, and then you will delete the backup file.

3.2.3. Import your key into Outlook

(on OS/X)

If you created your own CA and personal key pairs manually using openssl, then first follow this guide to import your own CA certificate.

Simply double-click the `.p12` backup file that was produced in the previous section. OS/X will then import your personal key into the "Keychain Access" tool. According to MS docs Outlook 2016 uses Keychain Access.
Older versions of Outlook may require more steps. Outlook 2011 users should read this doc.

(on Windows)

If you created your own CA and personal key pairs manually using openssl, then run `certutil -addstore Root ca.crt` first to import your CA certificate.

Watch YouTube video n3rOEpGjrc (the link of which jumps you to the relevant point in the video) or follow the steps below to import your key into Outlook 2013.

1. Go to File ≫ Options ≫ Trust Center (left pane) ≫ Trust Center Settings..

2. Go to Email Security (left pane) ≫ Digital IDs (Certificates) ≫ "`Import/Export`" (button)

3. Click "`Browse..`" (button)

4. Select the backup file that was produced in the previous section.

5. In the "`Password`" field enter the password that was chosen in final step of the previous.

6. Click "`OK`" in the pop-up dialog and the next window.

7. After the key is imported into Outlook, you should delete the backup file. (You can always create a new backup file from Firefox if needed).

3.3. Configuring Outlook

(on OS/X)

1. On the Tools menu, click Accounts.
2. Click the account that you want to send a digitally signed message from, click Advanced, and then click the Security tab.
3. Under Digital signing, on the Certificate pop-up menu, click the certificate that you want to use.

(on Windows)

Follow this document. That link skips the top portion and takes you straight to "How to set up your e-mail certificate in Outlook" because you already have a "digital ID". That will configure Outlook for using your certificate.

3.4. Distribute your S/MIME certificate (aka public key)

In short: simply send an e-mail to the recipient using Outlook, and sign the message. If you created your own CA key pair instead of using your browser to subscribe to a CA, then you must also attach the `ca.crt` file that you created.

Detailed explanation: You have a pair of keys (these were created in subsection 3.1 Get an S/MIME certificate). One is a public key and the other is a private key. The public key must be sent to those who will send you encrypted e-mail. They will use your public key to encrypt messages to you. Your public key is automatically contained in the signature of all messages you sign.

So to distribute your public key, simply send the other party an signed e-mail from Outlook. Encrypting this key distribution message is optional, but it must be signed. They can then extract your public key from your signature.

The `ca.crt` also must be sent to your recipient if you created one, because the application used by the recipient is unlikely to accept certificates without a chain of trust.

4. Prep to send encrypted mail

Before you can send someone an encrypted message, you need the recipients S/MIME certificate (public key). This will normally come to you when they send you a signed message, at which point you can extract the certificate. The certificate must then be associated to that person in your address book. Your Outlook Address Book has a *certificates* tab for each record, where you can verify and control the association between that person and their certificate.

Appendices

A. The S/MIME alternative: PGP

Pretty Good Privacy (“PGP”) is a public key infrastructure (“PKI”) just as S/MIME is, but it’s more popular among individuals outside of the enterprise context.

Outlook does not natively support PGP, so extra installations are required. For OS/X and Windows there is a proprietary product from Symantec. There is also a free software option for Windows users, where the typical PGP implementation is GPG4Win, which includes the MS Outlook plugin “GPGOL”. Historically Outlook has not worked well with PGP, but “GPGOL” has been overhauled in GPG4Win 3.0. This video shows GPGOL in action (but note that that video shows the old version).

This is a comparison between PGP and S/MIME on Outlook:

<table>
<tr><th></th><th>PGP</th><th>S/MIME
(without CA)</th><th>S/MIME
(with CA)</th></tr>
<tr><td>typical application</td><td>p2p</td><td>internal</td><td>b2b</td></tr>
<tr><td>ease of Outlook installation</td><td>3rd-party app (GPG) and plugin needed</td><td>3rd-party app needed (openssl), and extra setup effort for you and your correspondent</td><td>Supported natively.</td></tr>
<tr><td>key cost</td><td colspan="2">Keys are gratis for everyone (individuals and corporations)</td><td>Keys are gratis only for individuals and only from some CAs.</td></tr>
<tr><td>key lifetime</td><td colspan="2">Keys expire at any arbitrary date of your choosing, or never.</td><td>Keys expire in 6-24 months (approx).</td></tr>
<tr><td>flexibility</td><td>complete flexibility: 0, 1 or many addresses can be on one key, and many keys can have the same address</td><td colspan="2">one address per key (and also one key per address if the recipient uses iOS)</td></tr>
<tr><td>typical key distribution</td><td>publish your public key on a gratis public keyring or attach it to an e-mail</td><td colspan="2">public keyrings virtually non-existent; recipient cannot send you an encrypted message until you send them a signed email</td></tr>
</table>