Sichere E-Mail mit S/MIME und Microsoft Outlook[1]

Übersicht

Gegeben:

1) PC oder Laptop mit Windows 7, Windows 8.x oder Windows 10
2) Mindestens Microsoft Outlook 2007
3) Eine P12-Datei, in der das eigene S/MIME-Zertifikat und der eigene private Schlüssel vorliegen (Wie dies erzeugt wurde, wurde im ersten Dokument „Sichere Email mit S/MIME und Thunderbird“ beschrieben).

Vorgehensweise:

In vier Schritten sichere Email einrichten:

1. Schlüssel und Zertifikate im Keystore abspeichern.
2. Den Mail-Client Microsoft Outlook fürs Signieren und Verschlüsseln konfigurieren.
3. Erster Kommunikationsteilnehmer sendet eine signierte Mail, der zweite kann schon verschlüsselt antworten.
4. Fremde S/MIME-Zertifikate dauerhaft unter den Kontakten in Outlook speichern.

[1] Dies ist das 2. Dokument in der Reihe "Sichere Email mit S/MIME" (© 2016).

Inhaltsverzeichnis

SICHERE EMAIL MIT S/MIME UND OUTLOOK UNTER WINDOWS

Begriffserklärungen:

S/MIME	Protokoll für sichere Email
Outlook	Name des Microsoft Mail-Clients
Sichere Email	Mails, die signiert und verschlüsselt statt im Klartext verschickt werden.

Bemerkung: „E-Mail“ wird hier meist als „Email“ geschrieben.

Dokument-Status

Datum	19.04.2016
Version	1.1.0
Autoren	André Heller und Bernhard Esslinger Mit Unterstützung vom CrypTool-Projekt www.cryptool.org
Sprache	Deutsch
Lizenz	Keine bzw. Public-Domain bzw. GNU Free Documentation License
Aufbereitung	Schritt-für-Schritt mit vielen Bildschirmfotos (Screenshots)
Zielgruppe	Jedermann (Privat- und Heimanwender, die Ende-zu-Ende-verschlüsselt per Email kommunizieren wollen).

Dieses Dokument ist das **zweite** in der Reihe „Sichere Email mit S/MIME“. Die gesamte Reihe besteht aus den 4 Dokumenten:

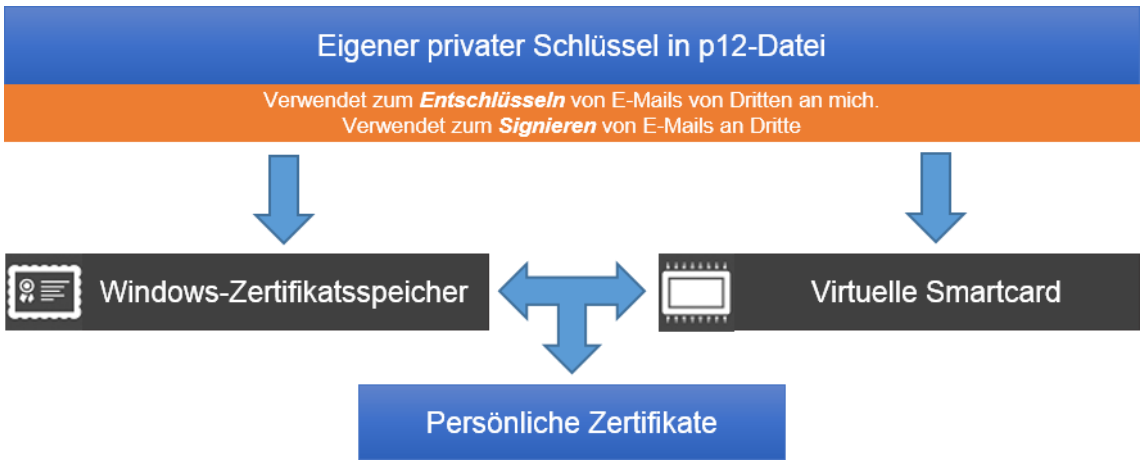
1. **Sichere Email mit S/MIME und Thunderbird (unter Windows, MAC und Linux)**
   Das 1. Dokument enthält die Theorie und erläutert, wie sichere S/MIME-Email mit dem Mailclient „Thunderbird“ funktioniert.
2. **Sichere Email mit S/MIME und Outlook unter Windows**
   Das 2. Dokument zeigt, wie es unter Windows mit dem Mailclient „Outlook“ geht (mit und ohne Virtual Smartcard).
3. **Sichere Email mit S/MIME unter Android**
   Das 3. Dokument zeigt, wie es unter Android mit dem Mailclient „SMail“ geht.
4. **Sichere Email mit S/MIME unter iOS**
   Das 4. Dokument zeigt, wie es unter iOS mit dem Mailclient „Mail“ geht.

Einleitung

Übersicht über die verwendeten Schlüssel, Zertifikate und ihre Speicherorte

Zu Beginn soll eine kurze Übersicht zeigen wie Windows, in Verbindung mit Outlook, die Zertifikate speichert und wofür die Zertifikate genutzt werden können.

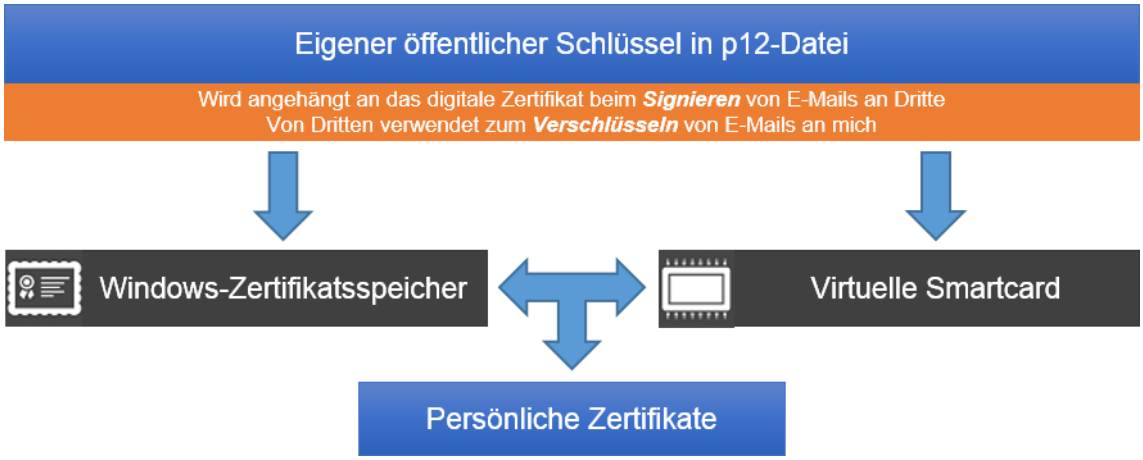
Privater Schlüssel (Private Key):



Mein privater Schlüssel wird zum **Ent**schlüsseln von Emails verwendet, die ein Dritter vorher mit meinem öffentlichen Schlüssel **ver**schlüsselt hat.

Der private Schlüssel wird entweder im Windows-Zertifikatsspeicher (Software) oder in einer virtuellen Smartcard (Hardware) gespeichert. Unabhängig davon wie der private Schlüssel gespeichert wird, ist der öffentliche Schlüssel im Windows Zertifikatsmanager (certmgr.msc) unter den persönlichen Zertifikaten sichtbar.

Eigener öffentlicher Schlüssel (Public Key):



Beim Signieren von Emails, die ich an Dritte verschicke, wird der eigene öffentliche Schlüssel (als Teil des digitalen Zertifikats) an die Mail angehängt. Dritte verwenden den angehängten öffentlichen Schlüssel zum Verschlüsseln von Emails an mich.

Der öffentliche Schlüssel wird zusammen mit dem privaten Schlüssel entweder im Windows-Zertifikatsspeicher (Software) oder in einer virtuellen Smartcard (Hardware) gespeichert. Der öffentliche Schlüssel ist im Windows Zertifikatsmanager (certmgr.msc) unter den persönlichen Zertifikaten sichtbar.

Fremder öffentlicher Schlüssel (Foreign Public Key):

Die fremden, öffentlichen Schlüssel werden zum Verschlüsseln von Emails verwendet, die ich an Dritte verschicke. Ein fremder öffentlicher Schlüssel kann z.B. in einer signierten Email empfangen werden.

Alle fremden, öffentlichen Schlüssel werden im Windows-Zertifikatsspeicher unter den vertrauenswürdigen Personen gespeichert.

Optional: Speicherort unter Windows 8.x

Normalerweise wird der private Schlüssel in einem Software-Container gespeichert (z.B. Mozilla Firefox, Windows-Zertifikatsspeicher etc.) und verwaltet. Wenn bereits Windows 8.x installiert ist und ein TPM-Chip (TPM = Trusted Platform Module) im PC verbaut ist, dann kann auch eine virtuelle Smartcard als Speicher des privaten Schlüssels gewählt werden. TPM-Chips sind Hardware-Chips, die fest auf der Platine des Rechners verbaut sind. Sie fungieren damit als eine Smartcard, die an den Rechner gebunden ist. Heutzutage sind TPM-Chips in nahezu jedem Laptop ab Werk dabei.

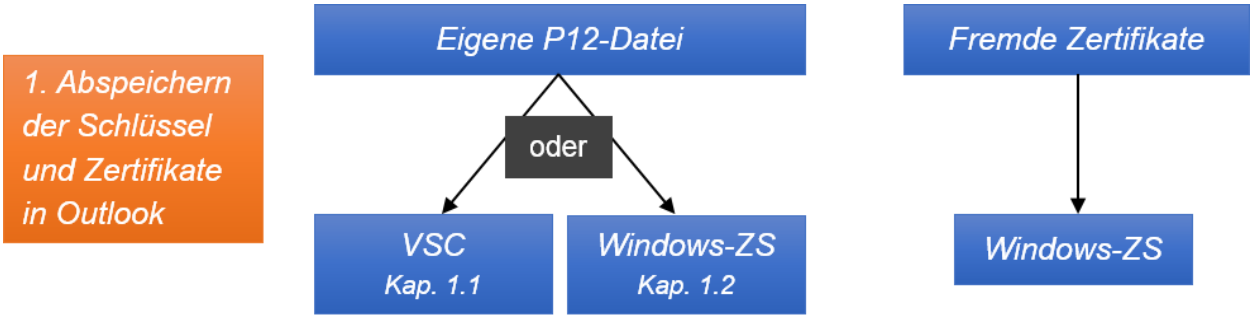
Ab Windows 8.x ist es möglich, private Schlüssel nicht nur in Software, sondern auch auf dem TPM-Chip zu speichern. Dies wird durch die kostenlos integrierte virtuelle Smartcard in Microsoft Windows ermöglicht. Dadurch ist man vor Software-seitigen Angriffen auf den privaten Schlüssel geschützt. Die virtuelle Smartcard wird durch eine PIN geschützt. Nur so ist der Zugriff auf den privaten Schlüssel möglich.

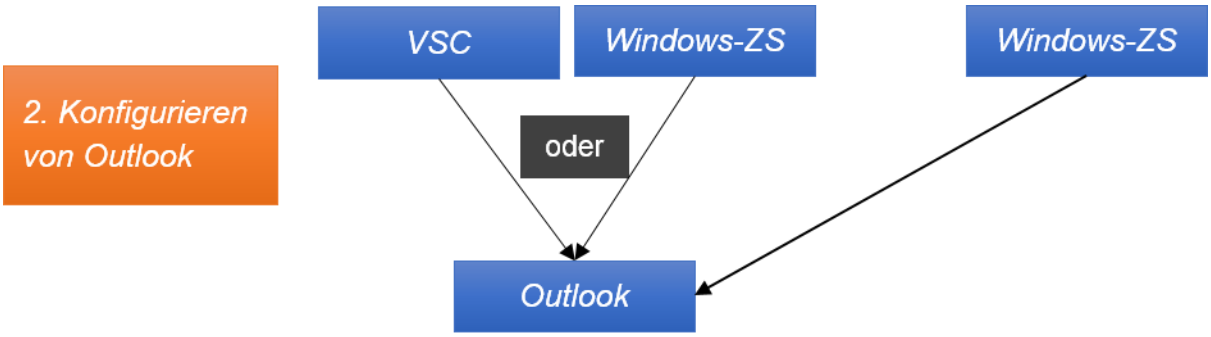
Weitere Informationen zur virtuellen Smartcard:
http://www.microsoft.com/en-us/download/details.aspx?id=29076
http://technet.microsoft.com/en-us/library/hh849637.aspx#BKMK_VSC

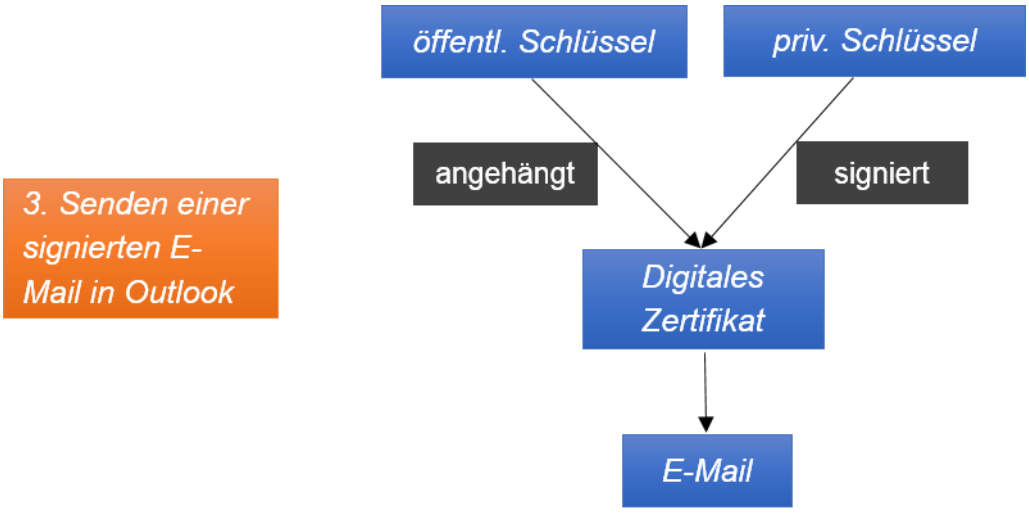
Im Folgenden wird visuell das Vorgehen erläutert, wie man sich die Schlüssel in Outlook einrichtet, wie man Outlook für sichere Email konfiguriert und wie man damit sichere Emails verschickt.

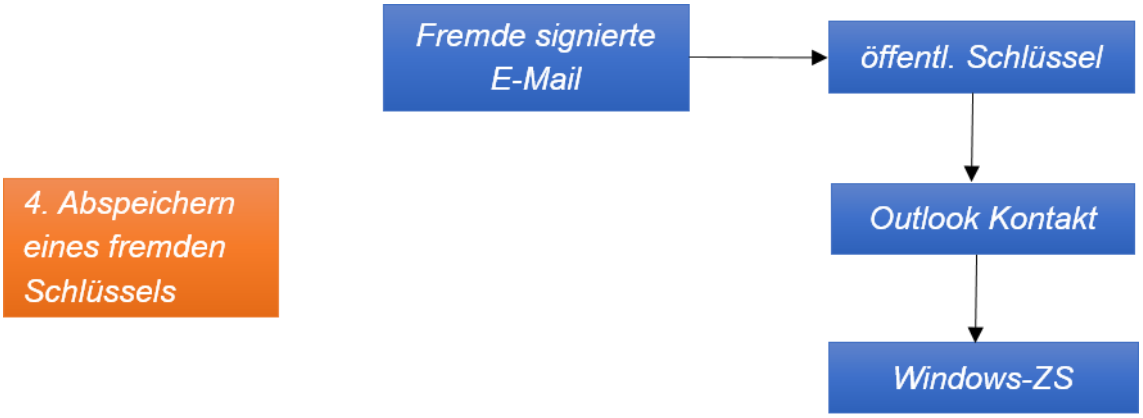
Dazu werden in den Grafiken die oben eingeführten Komponenten Zertifikat (mit öffentlichem Schlüssel), P12-Datei (mit privatem Schlüssel), Windows-Zertifikatsspeicher und Virtuelle Smartcard benutzt, so dass Sie die Komponenten in Workflows sehen können.

Grafische Darstellung der 4 Schritte








Fremde signierte E-Mail
öffentl. Schlüssel
Outlook Kontakt
Windows-ZS
4. Abspeichern eines fremden Schlüssels

Schritt 1: Schlüssel abspeichern

Zum Abspeichern des Schlüssels muss zunächst ein Schlüssel beantragt und generiert werden. Bitte hierzu Schritt 2 im Dokument „Sichere Email mit SMIME und Thunderbird unter Windows, MAC und Linux“ durchführen.

Wenn der Schlüssel auf der Festplatte als p12-Datei gesichert ist, gibt es zwei Alternativen den Schlüssel für die Verwendung mit Microsoft Outlook zu speichern:

1. In einer virtuellen Smartcard (ab Windows 8)
2. Im Windows-Zertifikatsspeicher

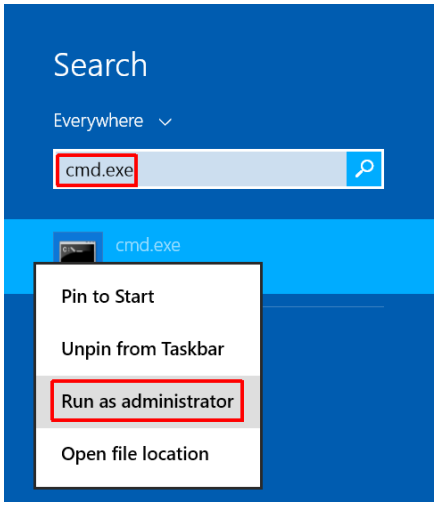
Fremde Zertifikate werden immer im Windows-Zertifikatsspeicher abgelegt.

Variante 1.1: Virtuelle Smartcard zum Abspeichern nutzen

Virtuelle Smartcard erstellen unter Windows 8.x und vorhandenes S/MIME-Zertifikat importieren.

Das Erstellen einer virtuellen Smartcard unter Windows 8.x ist relativ einfach. Es benötigt nur das Absetzen eines Kommandos über den sogenannten TPM VSC-Manager (Tpmvscmgr.exe). Das Programm bietet keine graphische Benutzeroberfläche, sondern wird über die Kommandozeile bedient.

Suchfeld öffnen (Windows-Taste) und `cmd.exe` eingeben. Diese Kommandozeile als Administrator ausführen.



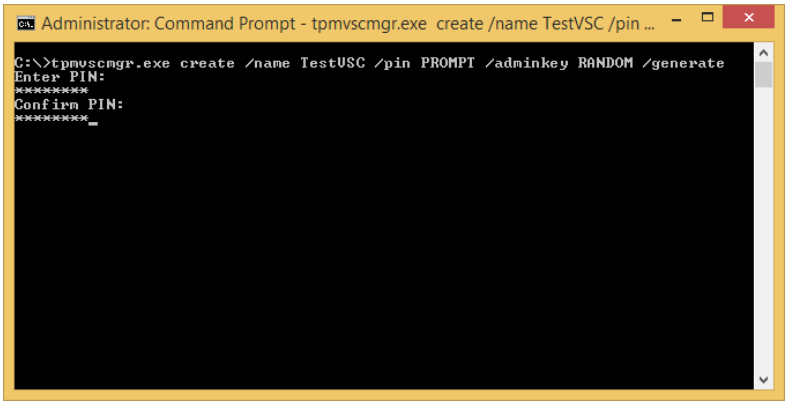
Das abzusetzende Kommando setzt sich aus folgenden Teilen zusammen:

tpmvscmgr.exe
create → erstelle eine virtuelle Smartcard
/name → frei wählbarer Name
/pin → vorgegebener Wert oder Aufforderung zum Eingeben eines Wertes
/adminkey → vorgegebener, spezifischer oder zufälliger Wert möglich
/generate → formatiere die virtuelle Smartcard

```
tpmvscmgr.exe create /name TestVSC /pin PROMPT /adminkey RANDOM
/generate
```

Das obige Kommando erstellt nun eine betriebsfertige virtuelle Smartcard mit dem Namen TestVSC. Bei der Erstellung muss eine PIN (mindestens 8 Zeichen) eingegeben werden, und der Administrator-Schlüssel* wird zufällig gewählt. Eine PUK wird nicht gesetzt: Somit ist bei vergessener PIN kein Zugriff auf die virtuelle Smartcard möglich, da es keinen Ersatz-Schlüssel gibt.

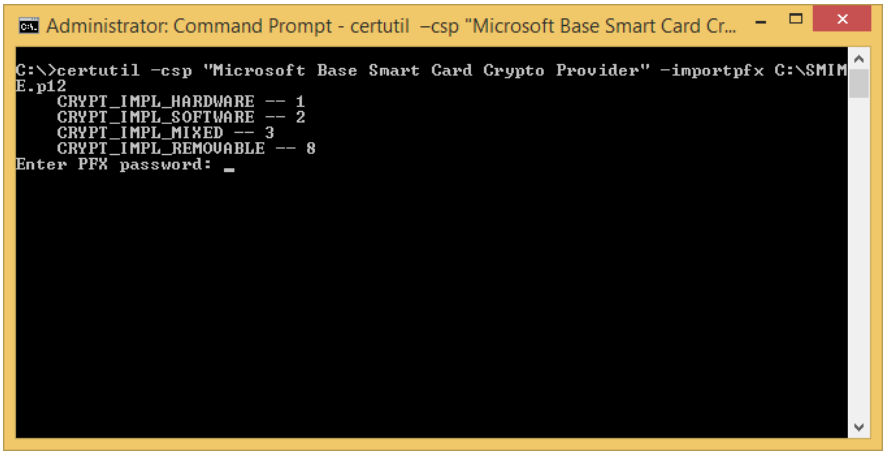
* Der Administrator Schlüssel oder auch „Admin Key“ oder „Unblock PIN“ wird zur zentralen Verwaltung von Smartcards benötigt zum Zurücksetzen der PIN und löschen von Zertifikaten. Mit diesem Schlüssel kann die Karte ohne Wissen der Nutzer-PIN konfiguriert werden.



Direkt nach dem Erstellen der virtuellen Smartcard kann das vorhandene S/MIME-Zertifikat aus der P12-Datei importiert werden. Dazu muss lediglich dieser Befehl in der Kommandozeile abgesetzt werden:

```
certutil –csp "Microsoft Base Smart Card Crypto Provider" –importpfx
PATH\FILENAME.p12
```

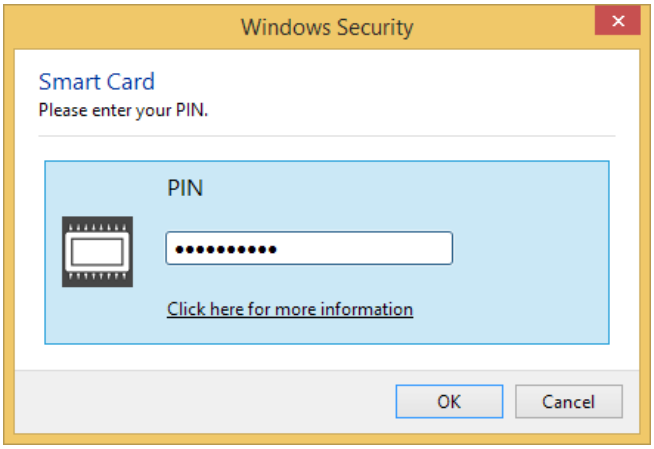
Dazu muss das Passwort zum Öffnen der P12-Datei eingeben werden.



Die vier im Screenshot zu sehenden „CRYPT_...“ Zeilen geben verschiedene Attribute des PP_IMPTYPE-Wertes des verwendeten Cryptographic Service Providers (CSP) wider. Sie geben an, wie der CSP implementiert ist.

`CRYPT_IMPL_HARDWARE 1:` *In Hardware implementiert (z.B. TPM –Chip)*
`CRYPT_IMPL_SOFTWARE 2:` *In Software implementiert (z.B. Windows-Zertifikatsspeicher)*
`CRYPT_IMPL_MIXED 3:` *Teilweise in Hardware implementiert (z.B. privater Schlüssel), teilweise in Software implementiert (z.B. Hash etc.).*
`CRYPT_IMPL_REMOVABLE 8:` *In Wechselmedien implementiert*

Danach die PIN der eben erstellten virtuellen Smartcard einzugeben, um sie zu öffnen.

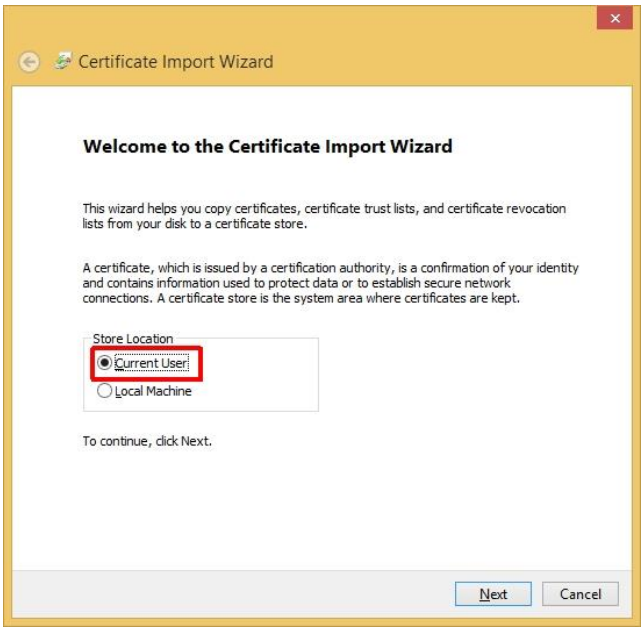


Nach erfolgreichem Import aus der P12-Datei kann diese entweder vom System gelöscht oder an einem sicheren Ort abgespeichert werden.

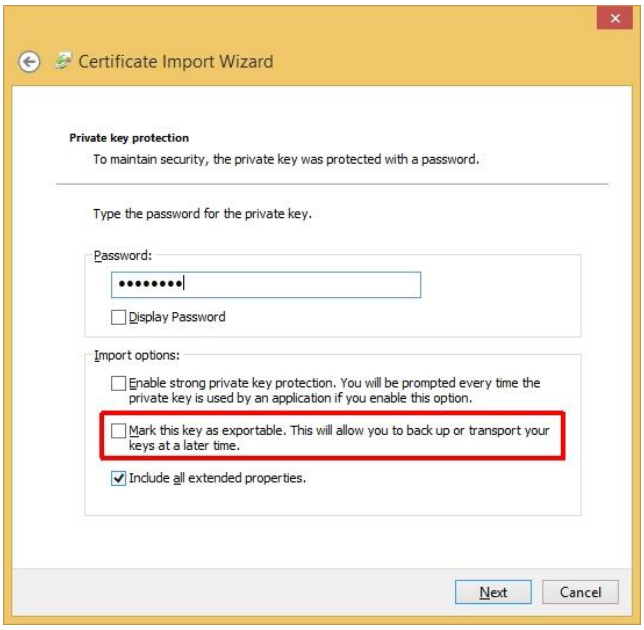
Variante 1.2: Windows-Zertifikatsspeicher zum Abspeichern nutzen

Alternativ zur virtuellen Smartcard kann der private Schlüssel auch im Windows-Zertifikatsspeicher abgelegt werden.

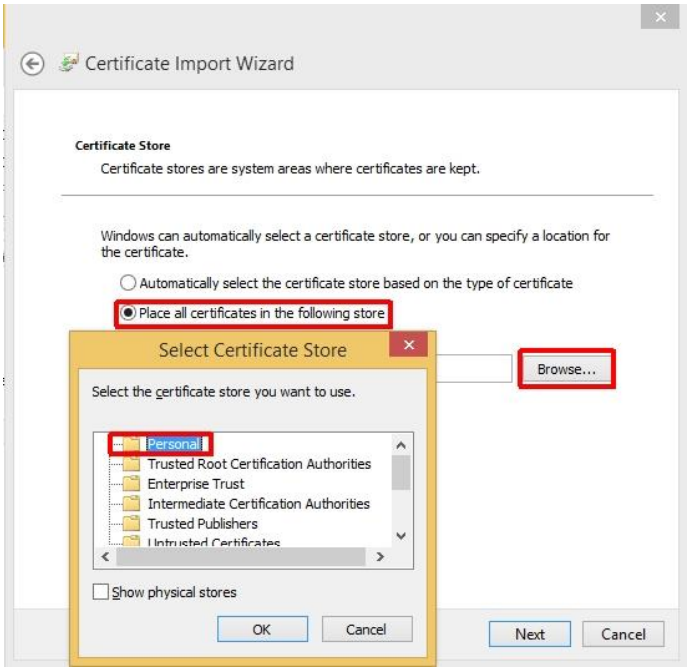
Doppelklick auf die P12-Datei und den Benutzerspeicher auswählen.



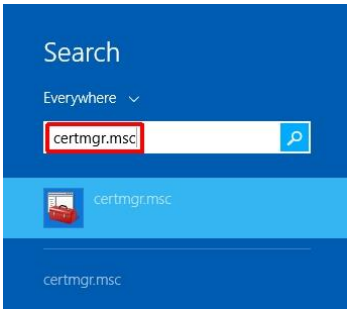
Bei „Exportable“ darf kein Haken gesetzt werden. Der Schlüssel soll nicht exportierbar sein.



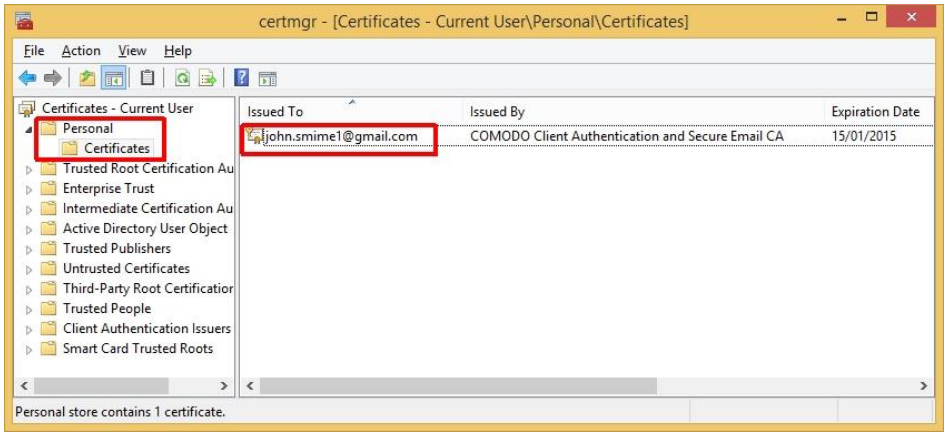
Persönlichen Speicher auswählen



Um sicher zu stellen, dass das eigene Zertifikat korrekt importiert wurde, das Suchfeld öffnen (Windows-Taste) und certmgr.msc öffnen (Zertifikatsmanager von Windows).



Unter den persönlichen Zertifikaten sollte nun das eigene Zertifikat angezeigt werden.

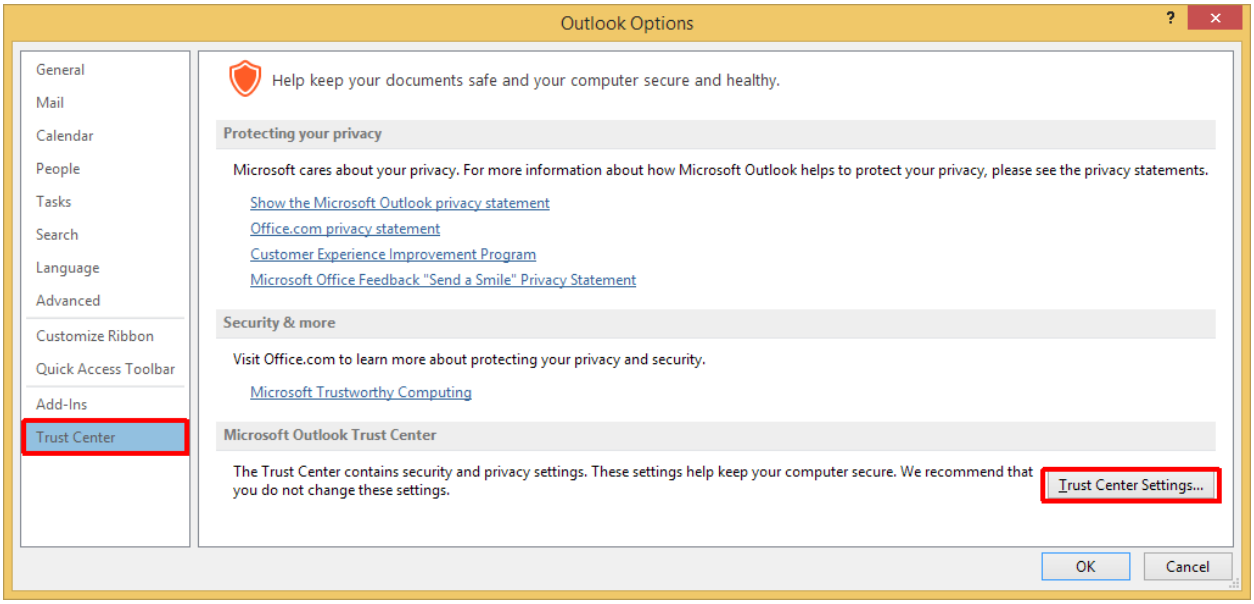


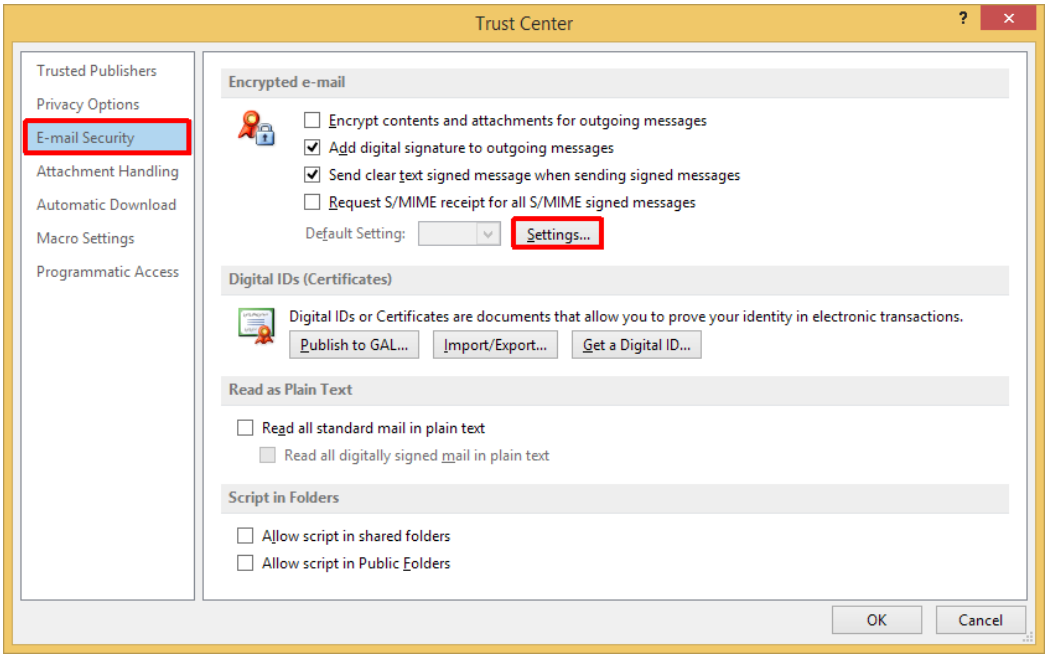
Nach erfolgreichem Import aus der P12-Datei kann diese entweder vom System gelöscht oder an einem sicheren Ort abgespeichert werden.

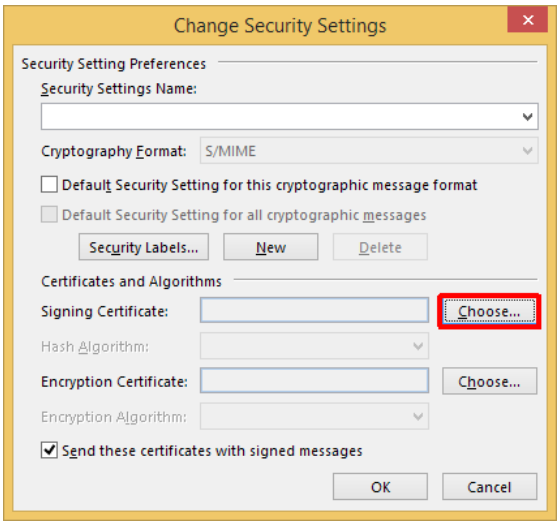
Schritt 2: Microsoft Outlook konfigurieren

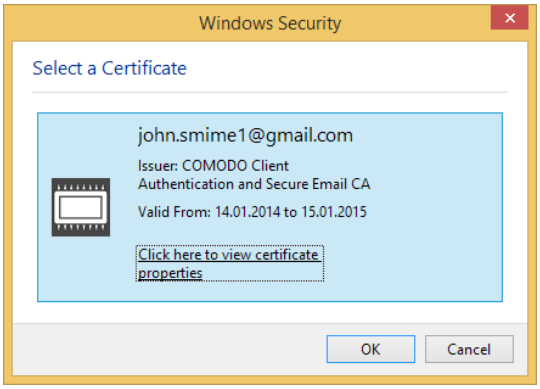
In diesem Schritt wird Microsoft Outlook so konfiguriert, dass das Signieren einer Email entweder mit Hilfe der eben erstellten virtuellen Smartcard oder mit Hilfe des Windows-Zertifikatsspeichers möglich ist.

Microsoft Outlook öffnen und auf „Datei“ → „Optionen“ gehen; dann auf „Trust Center“ und „E-mail Security“.



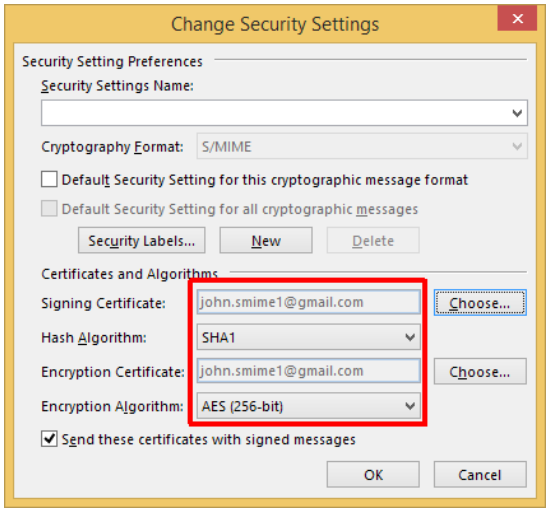




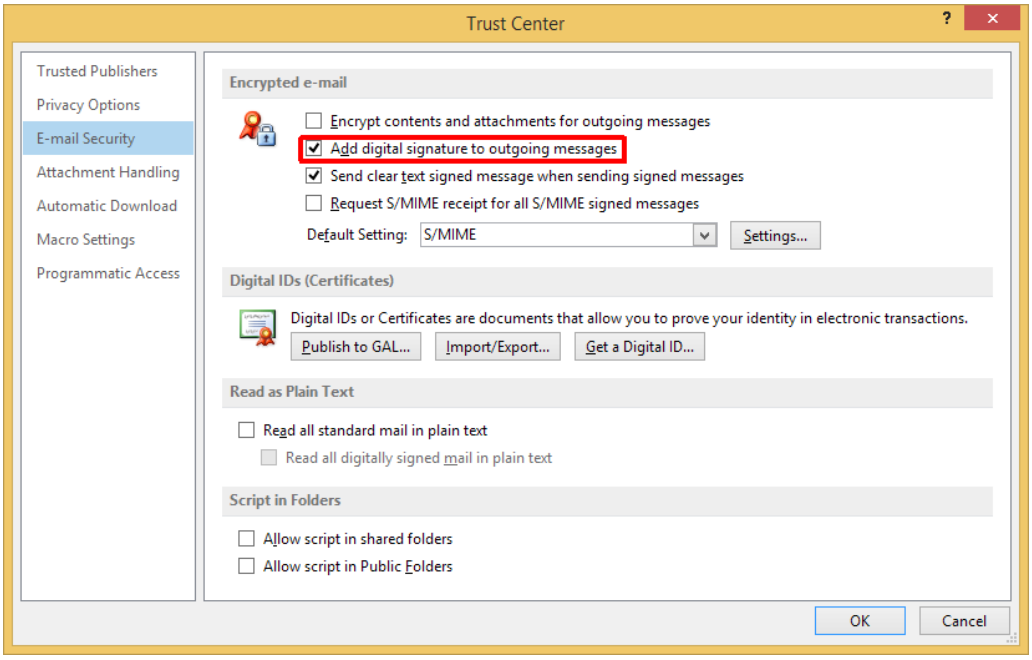


Ist das Zertifikat im Windows-Zertifikatsspeicher abgelegt, wird dieses Symbol  neben dem Zertifikat angezeigt.

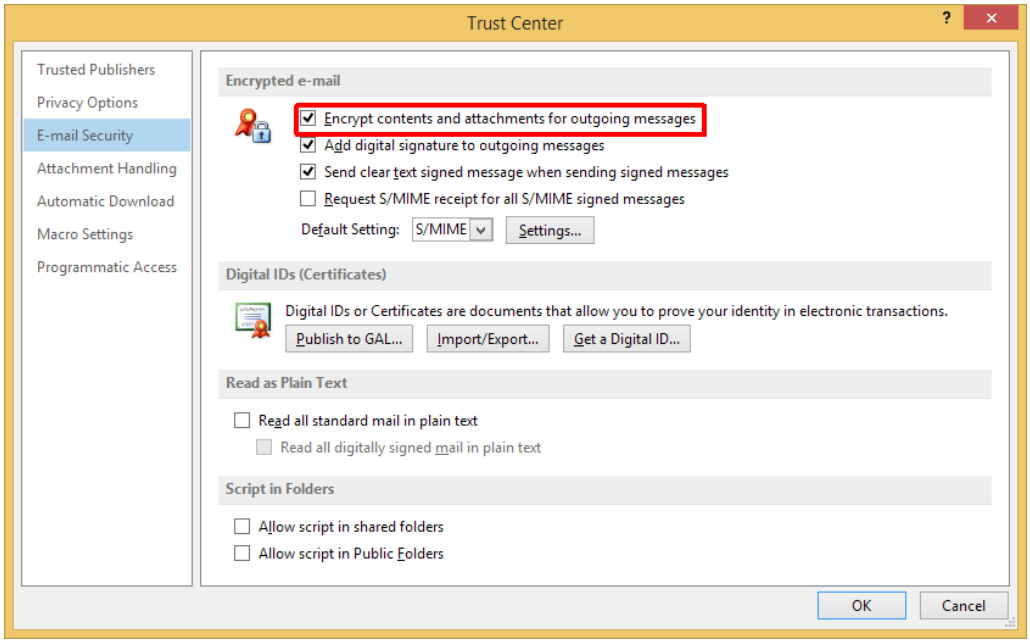
Sobald man sein S/MIME-Zertifikat ausgewählt hat, werden die unten stehenden Einstellungen automatisch anhand des jeweiligen Zertifikats ergänzt.



Bemerkung 1: Möchte man jede ausgehende Email signieren, dann muss hier ein Haken gesetzt werden:



Bemerkung 2: Möchte man zukünftig alle Emails und auch deren Anhänge direkt verschlüsseln, muss auch hier ein Haken gesetzt werden (Dies entspricht teilweise der Funktion der Thunderbird-Erweiterung „Encrypt-if-Possible“):



Bitte beachten: Wenn diese Option aktiviert ist, muss ich das öffentliche Zertifikat des Empfängers bereits empfangen haben. Nur dann kann ich meine Emails standardmäßig verschlüsseln. Sollte ich das öffentliche Zertifikat des Empfängers nicht haben, erscheint in Outlook eine Fehlermeldung, die mich darauf hinweist, dass Outlook kein Zertifikat zum Verschlüsseln der Email finden kann.

Ich habe dann die Möglichkeit, die Email unverschlüsselt zu senden oder den Vorgang abzubrechen und die Email nicht zu senden.

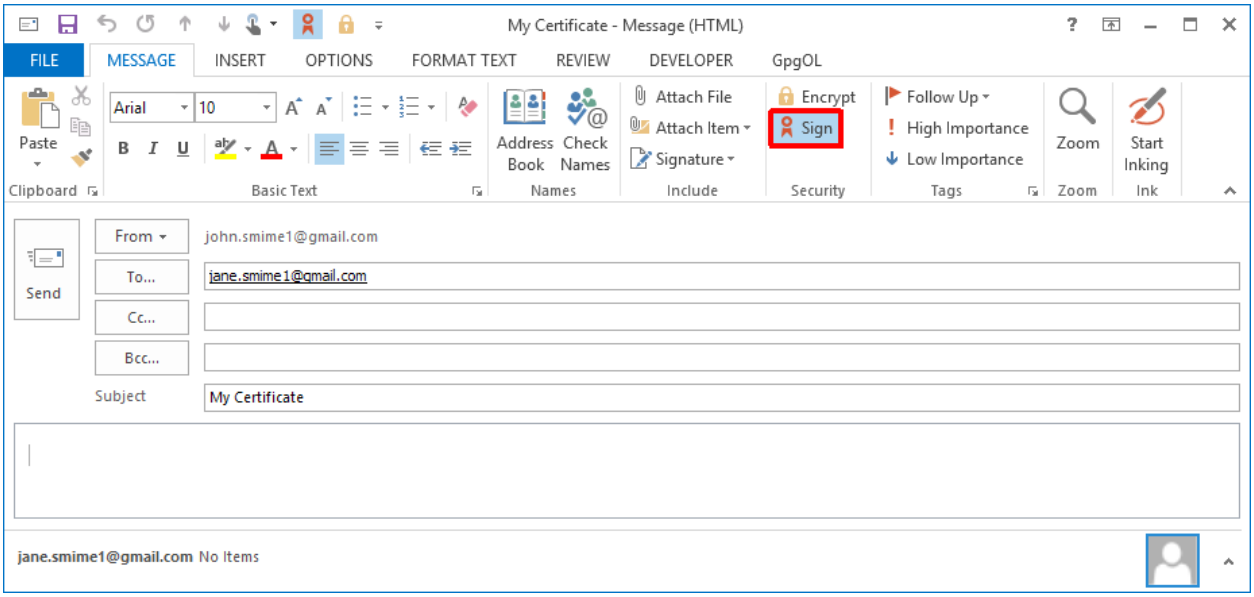
Nach diesem Schritt ist Outlook konfiguriert und bereit zum Versenden von signierten und verschlüsselten Emails.

Schritt 3: Eine signierte Email versenden

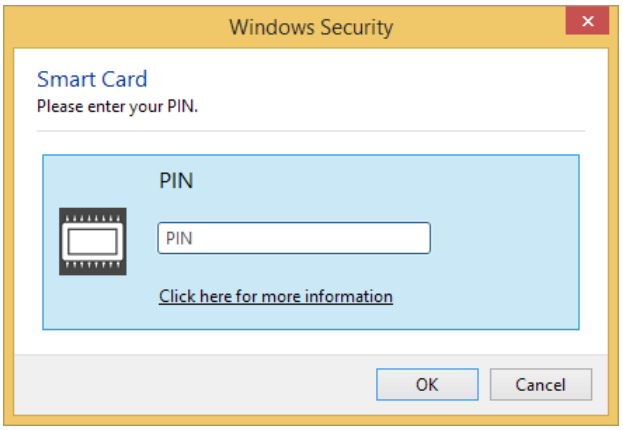
Erster Kommunikationsteilnehmer sendet eine signierte Mail, der zweite kann sofort verschlüsselt antworten.

Gehe in Microsoft Outlook im Hauptfenster auf „Neue E-Mail“.

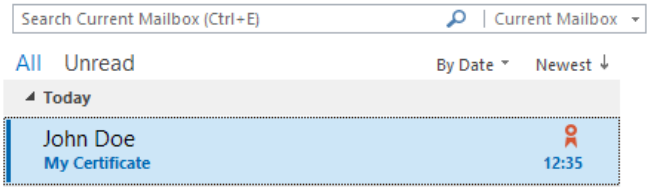
Vor dem Versenden der neuen Email-Nachricht nun unter dem Reiter „Sicherheit“ den Eintrag „Signieren“ auswählen.



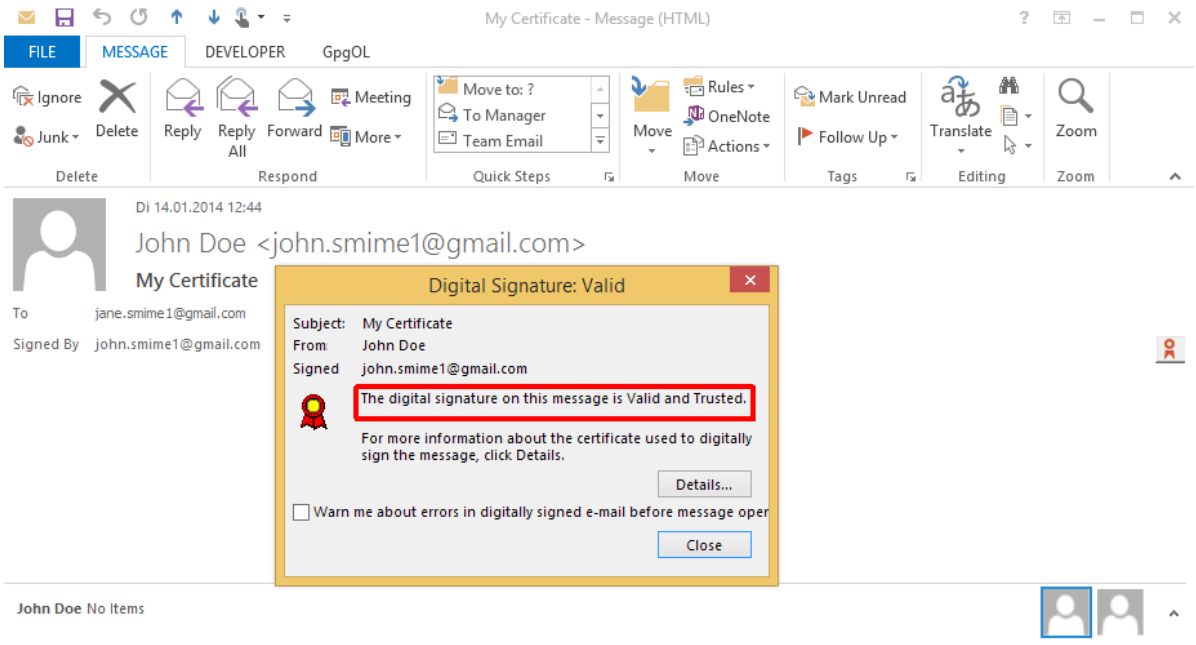
Bevor die Nachricht versandt wird, muss man seine PIN für die virtuelle Smartcard oder sein Passwort für den Zertifikatsmanager eingeben. Die signierte Email wird erst nach Eingabe der korrekten PIN versandt.



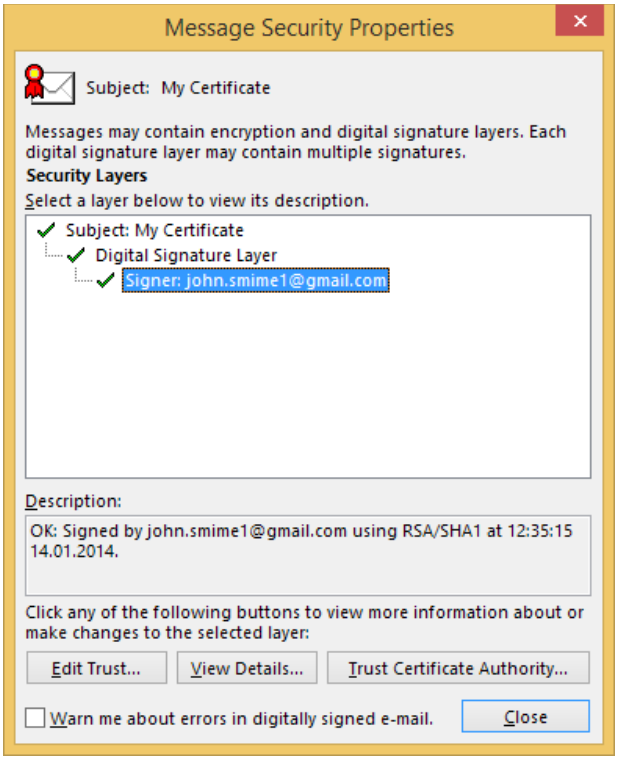
Der Empfänger der Email erhält nun – zusätzlich zur Signatur und angehängt an die Email – das S/MIME-Zertifikat des Absenders.
Dieses wird bei Outlook durch ein rotes Fähnchen in der Email Nachricht angezeigt.



Nach dem Öffnen der Email und einem Klick auf das rote Fähnchen wird angezeigt, von wem die Email signiert wurde und ob das Zertifikat gültig ist.



Nach Klick auf „Details“ wird die Zertifikatskette angezeigt.



Ist die Signatur gültig, wird beim Empfänger das Zertifikat des Senders automatisch im Windows-Zertifikatsspeicher gespeichert. Im Zertifikatsspeicher verbleibt es nur solange, wie man eine signierte Email des Senders hat oder wenn der Sender nach Erhalt einer signierten Email in Outlook als Kontakt gepflegt wird.

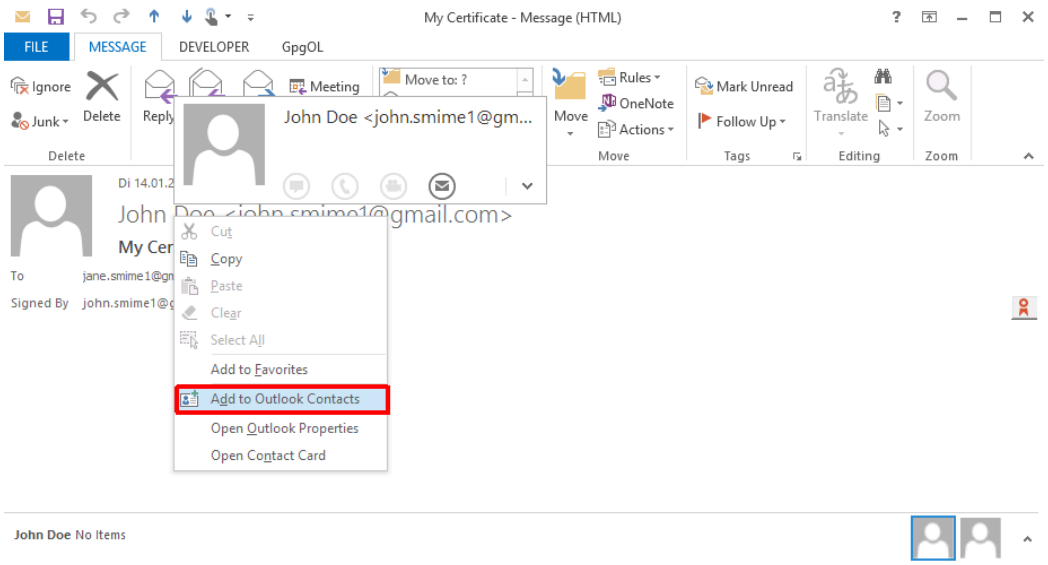
Der Empfänger kann nun entweder direkt auf die Email antworten: Diese Antwort kann er nicht nur (mit seinem privaten Schlüssel) signieren, sondern auch direkt verschlüsseln (dazu wird das öffentliche Zertifikat des Empfängers benutzt, das aus dem Windows-Zertifikatsspeicher gelesen wird).

Schritt 4: Fremde S/MIME-Zertifikate dauerhaft in Outlook speichern

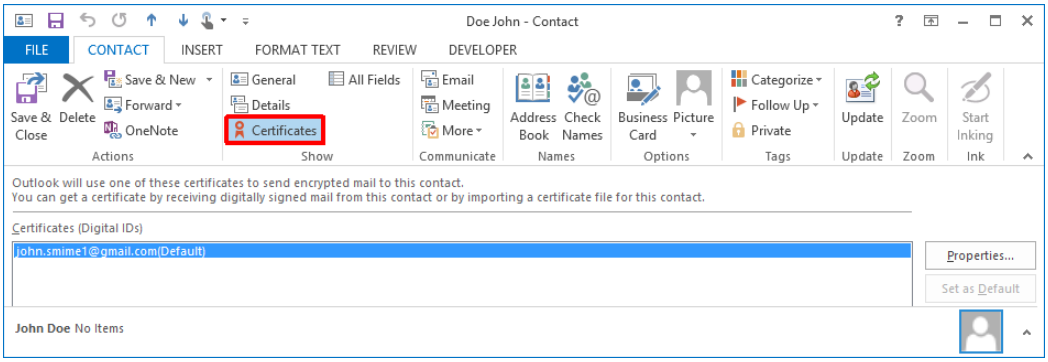
Zertifikate enthalten öffentliche Informationen, z.B. den öffentlichen Schlüssel des Senders. Das Zertifikat des Senders wird normalerweise (im Klartext) mit gesendet – sowohl bei signierten als auch bei verschlüsselten Emails. Leider werden die erhaltenen S/MIME-Zertifikate nach Erhalt einer signierten Email nicht dauerhaft in Microsoft Outlook gespeichert: Sobald alle signierten Emails eines Kommunikationspartners gelöscht sind, ist auch dessen öffentlicher Schlüssel verloren. Um weiterhin Emails an diese Person zu verschlüsseln, muss diese mir erst eine neue signierte Email senden.

Möchte man unabhängig von alten Konversationen Emails verschlüsseln, muss die Person zu den Outlook-Kontakten hinzugefügt werden. Das öffentliche S/MIME-Zertifikat wird dann automatisch mit gespeichert.

Dazu eine signierte Email öffnen, per Rechtsklick auf die Adresse des Versenders klicken und „Hinzufügen zu Outlook-Kontakten“ auswählen.



Wenn danach der Kontakt geöffnet wird, können unter „Zertifikate“ alle zu dieser Person gehörenden Zertifikate angezeigt werden.



Anhang

Für die Tests und Screenshots wurden zwei Google Email-Adressen angelegt und für diese jeweils ein Comodo-Zertifikat beantragt.

- John.smime1@gmail.com
- Jane.smime1@gmail.com

Diese Adressen werden ausschließlich für Verschlüsselungstests genutzt und dienen nur zur Veranschaulichung.