

# Kapitel 13

## Grafische Zusatzanwendungen

Hier folgt eine Beschreibung aller grafischen Zusatzanwendungen, auf die über Java-Clients zugegriffen werden kann und die die Funktionalitäten von i-effect sinnvoll ergänzen.

### Keystore-Verwaltung mit dem i-effect Keymanager

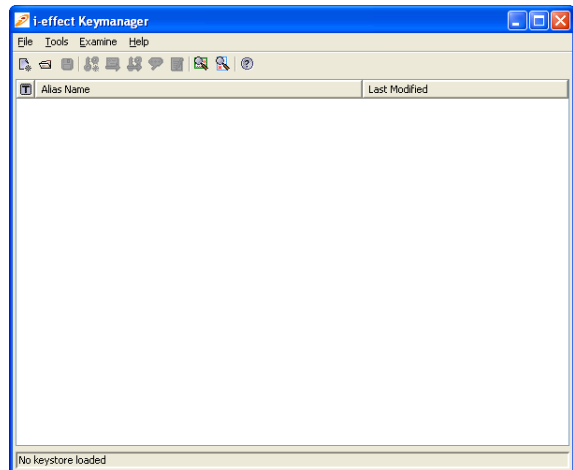
Für die Verwaltung des eigenen Schlüsselpaares und der Partnerzertifikate bietet sich das im Lieferumfang enthaltene Programm „i-effect Keymanager“ an. i-effect Keymanager ist ein Programm zur Erstellung, Verwaltung und Prüfung von Keystores, Schlüsseln, Zertifikaten, Zertifikatsanfragen und Rücknahme von Zertifikaten. Es ist ein Open Source Programm das unter der GNU GENERAL PUBLIC LICENSE steht und somit frei verwendet werden kann. Nach erfolgreicher Installation von i-effect finden Sie i-effect Keymanager im Verzeichnis `/i-effect/<version>/CRYPT/tools` unter dem Namen `,i-effectKeymanager.jar'`.

In der Version 1.1 bietet es folgende Funktionen:

- o Erstellung, Laden, Speichern, Löschen und Konvertierung von Keystores oder Keystore-Einträgen
- o Generierung von DSA und RSA Schlüsselpaaren mit eigensignierten X.509 Zertifikaten
- o Import von X.509 Zertifikaten
- o Import von Schlüsselpaaren aus PKCS#12 (Public-Key Cryptography Standards) Dateien
- o Passwortvergabe / Passwortverwaltung für Schlüsselpaare und Keystore
- o Detailanzeige von im Keystore befindlichen Zertifikaten und Schlüsselpaaren
- o Exportfunktion für Keystore Einträge in mehrere Formate  
Generierung von Zertifikats Anfragen (CSRs)

## Starten des i-effect Keymanagers

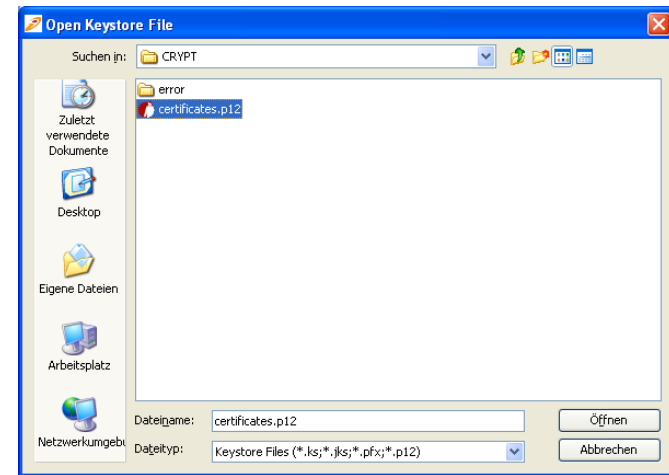
Da i-effect Keymanager ein Java-Programm ist, benötigen Sie auf dem System, von dem aus Sie i-effect Keymanager starten, eine installierte Version der Java Runtime Umgebung (JRE). Falls Sie keine Java Runtime Umgebung auf diesem System installiert haben und i-effect Keymanager als Keystore-Verwaltungsprogramm einsetzen möchten, laden Sie sich bitte von <http://java.sun.com> eine aktuelle Java Runtime herunter. Nachdem Sie diese erfolgreich installiert haben, sollte sich i-effect Keymanager durch einen Doppelklick auf die Datei *i-effectKeymanager.jar* starten lassen.



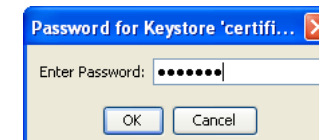
## Öffnen des i-effect-Keystore

Den Standard-Keystore von i-effect finden Sie im Verzeichnis */i-effect/<version>/crypt* unter dem Namen *certificates.p12*. Um diesen mit i-effect Keymanager zu öffnen, gehen Sie wie folgt vor:

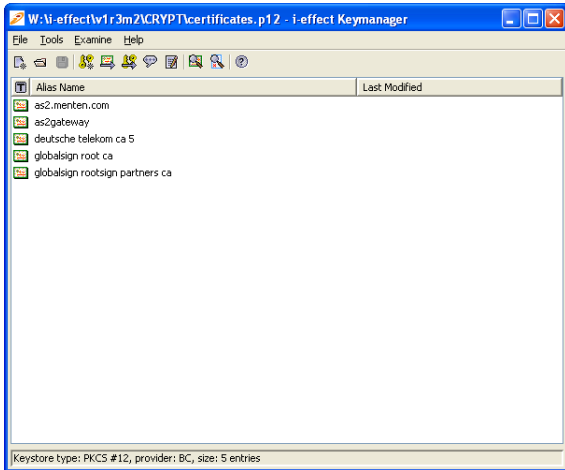
- 1) File -> Open Keystore File...(Strg+O)
- 2) In dem zur Anzeige kommenden Dialogfenster wechseln Sie bitte in das Verzeichnis */i-effect/<version>/crypt*. Wählen Sie die Datei *certificates.p12* aus und bestätigen Sie den Dialog mit *Öffnen*.



- 3) Um den Keystore öffnen zu können müssen Sie zunächst das Passwort des Keystore eingeben. Das Standardpasswort des Keystore *certificates.p12* ist: *ieffect*.



- 4) Nachdem Sie das Passwort eingegeben und den Dialog mit OK bestätigt haben, werden die im Keystore certificates.p12 enthaltenen Einträge angezeigt:



Bei den Einträgen handelt es sich um die Zertifikate von:

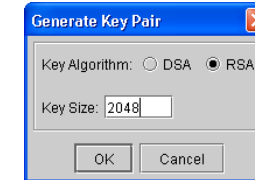
- o Cybertrust  
(Keystorealias: gte cybertrust global root )
- o Deutsche Telekom  
(Keystorealias: deutsche telekom ca 4 (gte cybertrust global root)
- o menten GmbH  
(Keystorealias: as2.menten.com)

Diese drei im Keystore enthaltenen Zertifikate bilden eine Zertifikatskette (siehe Def. 1.9).

## Erstellung eines Schlüsselpaares

- 1) Um in einem Keystore ein Schlüsselpaar anzulegen verwenden Sie bitte Tools -> Generate Key Pair...(Strg + G)

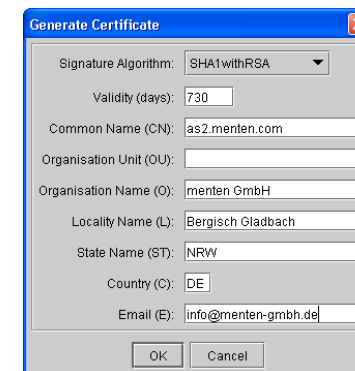
Folgendes Dialogfenster öffnet sich:



- 2) Wählen Sie RSA. aus.

- RSA Das RSA-Kryptosystem ist ein asymmetrisches Kryptosystem, d.h. es verwendet verschiedene Schlüssel zum Ver- und Entschlüsseln. Es ist nach seinen Erfindern Ronald L. Rivest, Adi Shamir und Leonard Adleman benannt.
- DSA Der Digital Signature Algorithm ist ein Standard der US-Regierung für Digitale Signatur.

Nachdem Sie mit *Ok* Ihre Eingaben bestätigt haben, öffnet sich folgendes Fenster:



- 3) Tragen Sie hier nun Ihre Zertifikatsdetails ein und bestätigen Sie mit OK. Weiter unten finden Sie eine Erläuterung der Eingabefelder. Belassen Sie die Einstellung „Signature Algorithm“ auf „SHA1 with RSA“. Im nächsten Dialog werden Sie nun dazu aufgefordert dem eben generierten Schlüsselpaar einen Aliasnamen zu vergeben.



- 4) Die erfolgreiche Schlüsselgenerierung wird Ihnen mit einem weiteren Dialog angezeigt. Bestätigen Sie diesen mit OK um wieder zum Hauptdialog zu gelangen.




---

Wir empfehlen eine Schlüssellänge von 2048 Bit und eine Gültigkeitsdauer von 730 Tagen für das Schlüsselpaar zu verwenden.

---

Erläuterung der Eingabefelder

### Signature Algorithm

Der Algorithmus, der verwendet werden soll, um die Signatur zu errechnen.

### Validity (days)

Anzahl der Tage, die das erstellte Zertifikat gültig ist.

### Common Name (CN)

Beim Importieren wird der CN als Aliasname für das Zertifikat vorgeschlagen. Viele CAs verlangen, dass als 'Common Name' der Domainname verwendet werden muss, da dieser eindeutig ist. Falls Sie Ihr Zertifikat von einer Certificate Authority signieren lassen möchten, sollten Sie sich vor der Erstellung des Zertifikates über die Vorgaben informieren.

### Organisation Unit (OU)

Die Einheit (z. B. Filiale, Niederlassung) der Organisation (z. B. Firma, Behörde). Dieses Feld sollte allerdings nur ausgefüllt werden, falls Ihre Organisation mehr als eine Einheit besitzt.

### Organisation Name (O)

Der Name Ihrer Organisation.

### Locality Name (L)

Der Sitz Ihrer Organisation.

### State Name (ST)

Der Name des Staates/Bundesland etc. in dem der Sitz Ihrer Organisation ist.

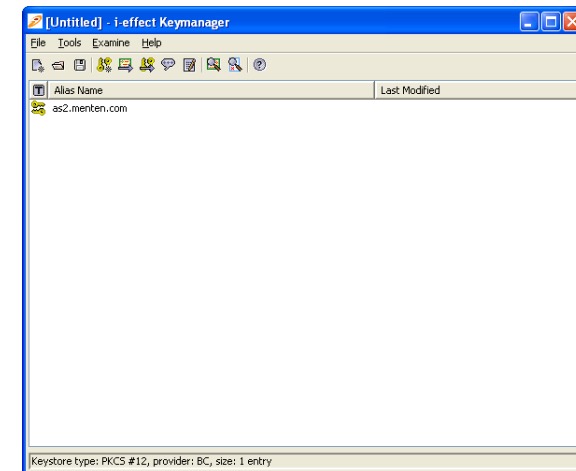
### Country (C)

Das Land, in dem Ihre Organisation ihren Sitz hat.

### Email (E)

Die eMail-Adresse Ihrer Organisation.

Ihr Keystore beinhaltet nun ein gültiges Schlüsselpaar:



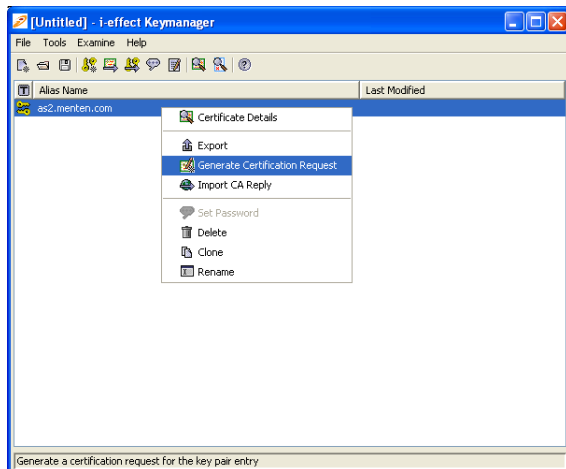
- 5) Da Sie den Keystore um ein Schlüsselpaar ergänzt haben und somit den Keystore verändert haben, ist es nötig, die Änderungen im Keystore mittels *File--> Save Keystore (Strg+S)* zu speichern.

Das erstellte Schlüsselpaar beinhaltet neben Ihrem privaten Schlüssel auch Ihren öffentlichen Schlüssel. Um mit Ihren Partnern verschlüsselte AS2-Nachrichten austauschen zu können, ist es nötig, Ihren Partnern Ihren öffentlichen Schlüssel zukommen zu lassen. Ihren öffentlichen Schlüssel können Sie in Form eines Zertifikates exportieren, was im Verlauf dieser Dokumentation noch weiter erklärt wird.

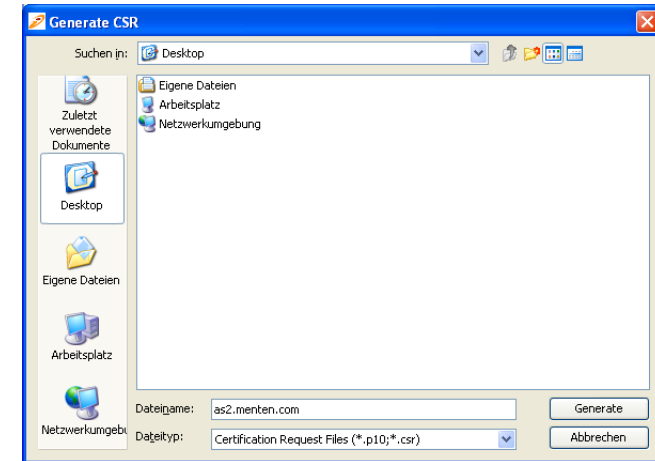
## Erstellung einer Zertifikatsanfrage (Certificate Request)

Um Ihr Zertifikat von einer höheren Instanz, einer sogenannten Zertifizierungsstelle (CA), signieren zu lassen, also damit Ihr eigenes Zertifikat beglaubigen zu lassen, müssen Sie eine Certificate request erstellen. Um dies zu tun, gehen Sie bitte folgendermaßen vor:

- 1) Bewegen Sie den Mauszeiger über den Eintrag Ihres Schlüsselpaars und drücken Sie die rechte Maustaste. In dem unten abgebildeten Kontextmenü wählen Sie den Menüpunkt *Generate Certification Request*.

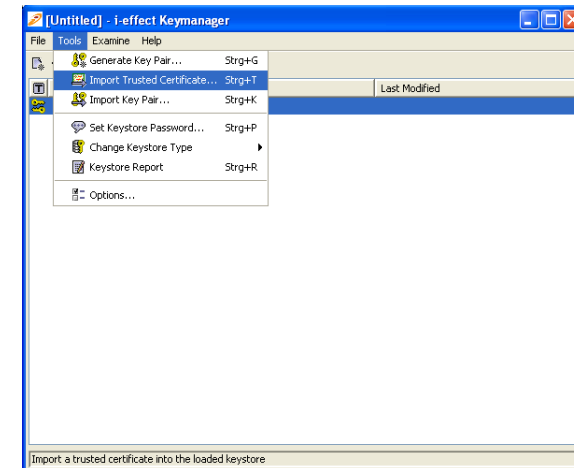


Es öffnet sich der unten abgebildete Speicherdialog.



- 2) Wählen Sie einen Ordner aus und vergeben Sie einen Namen für Ihr Certificate Request. Bitte beachten Sie dabei, dass Sie eine der beiden Dateierweiterungen mit angeben, die Ihnen unter Dateityp vorgeschlagen werden (\*.p10 ; \*.csr). Danach bestätigen Sie den Speicherdialog mit *Generate*.

Um Ihr Zertifikat von einer Certificate Authority signieren zu lassen, müssen Sie abschließend die eben erstellte Datei an die Certificate Authority Ihrer Wahl übertragen. Die Certificate Authority wird Ihnen nach erfolgreicher Validierung Ihrer Daten ein signiertes Zertifikat zurückschicken. Das von der CA erhaltene Zertifikat muss nun wieder in Ihren Keystore importiert werden.



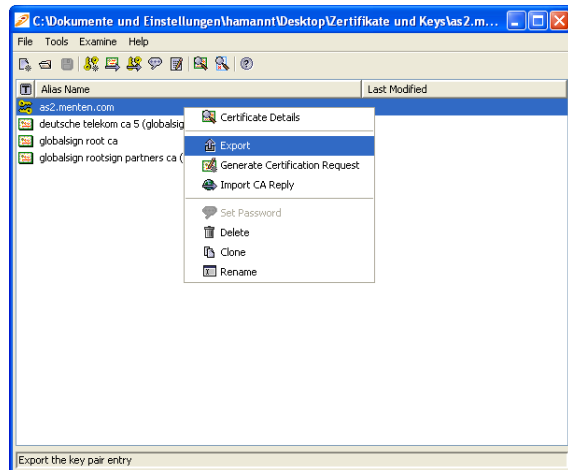
Ihr Zertifikat können Sie nun zusammen mit dem Zertifikat der Certificate Authority an Ihre Partner versenden. Ihr Zertifikat gilt somit als vertrauenswürdig, allerdings muss dazu das Zertifikat der Certificate Authority, die Ihr Zertifikat signiert hat, im Keystore Ihres Partners vorhanden sein, bevor Ihr Zertifikat importiert wird.

## Export des eigenen Zertifikats

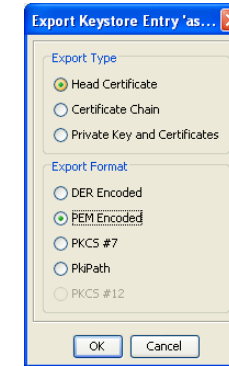
Damit Sie mit Ihren Partnern unter Anwendung aller in AS2 zur Verfügung stehenden Sicherheitsmechanismen erfolgreich Daten austauschen können, ist es notwendig, Ihren Partnern Ihren öffentlichen Schlüssel in Form eines Zertifikates zukommen zu lassen. Um dieses Zertifikat mit Ihrem öffentlichen Schlüssel zu erhalten, müssen Sie es aus Ihrem Schlüsselpaar wie folgt exportieren:

- 1) Bewegen Sie den Mauszeiger über Ihr Schlüsselpaar und betätigen Sie die rechte Maustaste.

Folgendes Kontextmenü kommt zur Anzeige:

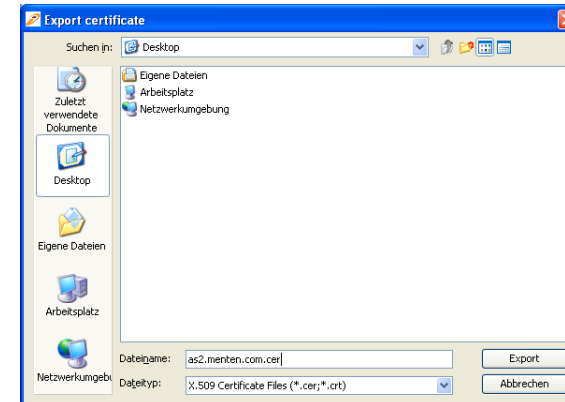


- 2) Wählen Sie in diesem Kontextmenü bitte den Menüpunkt *Export*. Es öffnet sich folgendes Dialogfenster:



- 3) Ändern Sie das Export-Format von DER Encoded nach PEM Encoded und bestätigen Sie mit OK, um Ihren öffentlichen Schlüssel als Zertifikat zu exportieren.

Es öffnet sich ein Dialogfenster zum Speichern eines Zertifikates.



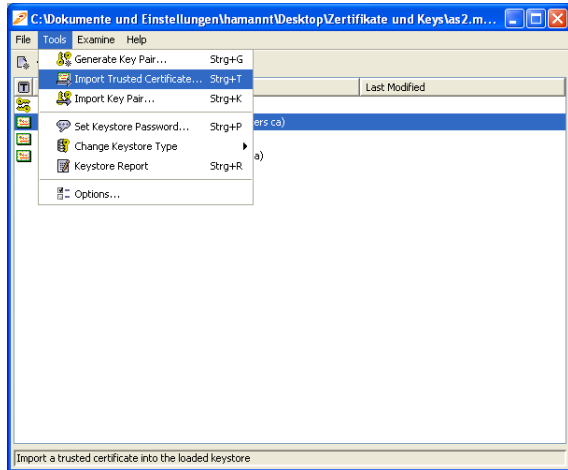
- 4) Wählen Sie einen Ordner aus und vergeben Sie einen Namen für Ihr Zertifikat. Bitte beachten Sie dabei, dass Sie eine der beiden Dateiendungen mit angeben, die Ihnen unter Dateityp vorgeschlagen werden (\*.cer ; \*.crt). Danach bestätigen Sie den Speicherdialog mit Export.

## Import von Partnerzertifikaten

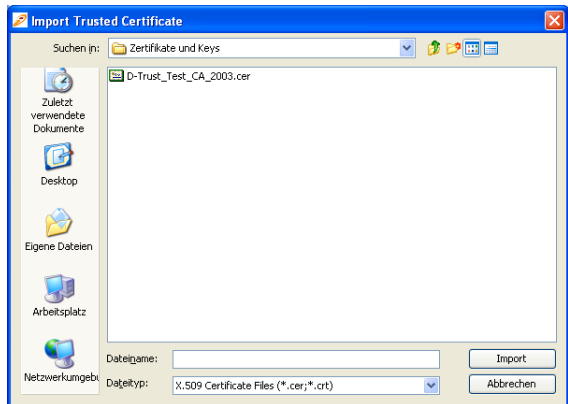
Um die Daten von AS2-Nachrichten, die Sie an Ihre Partner schicken wollen, verschlüsseln zu können, ist es notwendig, dass die Zertifikate, die den öffentlichen Schlüssel Ihres Partners enthalten, im Keystore enthalten sind. Um die Zertifikate Ihrer Partner in den Keystore zu importieren, gehen Sie bitte wie folgt vor.:

- 1) Tools -> Import Trusted Certificate...(Strg+T)

Folgendes Kontextmenü kommt zur Anzeige:



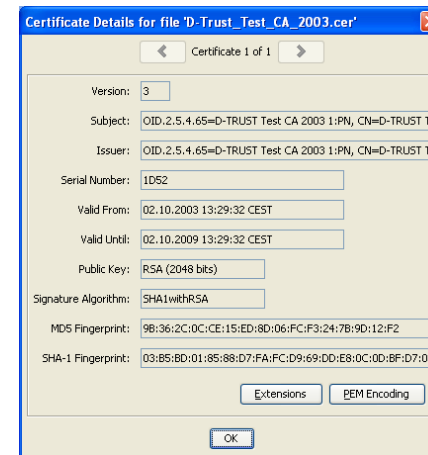
- 2) In dem folgenden Dialogfenster wechseln Sie bitte in das Verzeichnis, in dem Sie das Zertifikat Ihres Partners abgelegt haben. Wählen Sie es aus und bestätigen Sie das Dialogfenster mit *Import*.



Bei der Importierung von Zertifikaten können Ihnen zwei Situationen begegnen: Es besteht zum einen die Möglichkeit, dass das Zertifikat, welches Sie importieren wollen, zwar Teil einer Zertifikatskette ist, das root-Zertifikat (siehe Def. 1.9), oder von dem Zertifikat abhängige Zertifikate, nicht in Ihrem Keystore vorhanden sind. In diesem Fall, sowie wenn es sich um ein selbstsigniertes (ohne CA Signatur) Zertifikat handelt, bekommen Sie folgenden Hinweis von i-effect Keymanager angezeigt:



Dieser Hinweis beschreibt den oben erläuterten Sachverhalt ebenfalls. Um das Zertifikat Ihres Partners selbst zu validieren werden Ihnen, nachdem Sie den Dialog mit OK bestätigt haben, die Zertifikatsdetails angezeigt. Der folgende Screenshot zeigt eine solche Detailansicht.



Anhand dieser Detailansicht können Sie das Zertifikat Ihres Partners selbst validieren, um später nach Bestätigung des Dialoges mit OK zu entscheiden, ob Sie diesem vertrauen oder nicht.



Wenn Sie dabei nun das oben abgebildete Dialogfenster mit Ja bestätigen, wird das Zertifikat, nachdem Sie im nächsten erscheinenden Dialogfenster einen Aliasnamen für das Zertifikat vergeben haben, im Keystore gespeichert. i-effect Keymanager wird Ihnen automatisch den im Zertifikat unter CN (Common Name) eingetragenen Namen als Aliasnamen für das Zertifikat vorschlagen. Sie können natürlich das Zertifikat unter dem vorgeschlagenen Namen oder einem selbst gewählten Namen speichern. Den Aliasnamen können Sie jederzeit ohne großen Aufwand ändern.



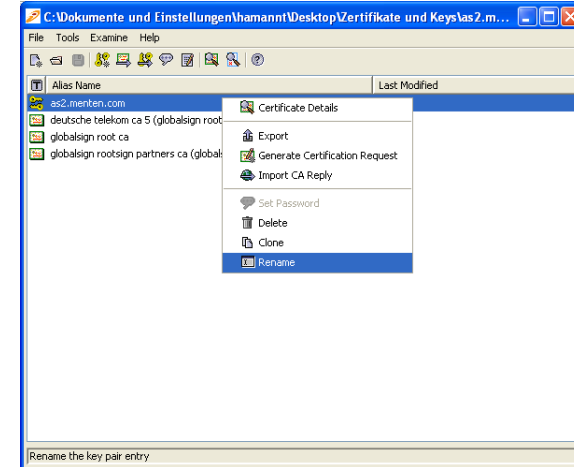
Die zweite Situation, die Ihnen beim Importieren von Zertifikaten begegnen kann, ist folgende: Falls das zu importierende Zertifikat von einer CA signiert worden ist und somit Teil einer Zertifikatskette werden Sie zu keiner Validierung der Zertifikatsdaten aufgefordert, wenn das root-Zertifikat der CA und alle davon abhängigen Zertifikate in Ihrem Keystore enthalten sind. Den oben abgebildeten Bestätigungsdiallog (Import Trusted Certificate) bekommen Sie ebenfalls nicht angezeigt, da durch das Vorhandensein abhängiger Zertifikate in Ihrem Keystore die Vertrauenskette (Certificate Chain) geschlossen ist und das Zertifikat somit automatisch als vertrauenswürdig eingestuft wird.

## Ändern eines Aliasnamens

- Um einen Aliasnamen eines im Keystore gespeicherten Zertifikates zu ändern gehen Sie wie folgt vor:

Bewegen Sie den Mauszeiger über das Zertifikat, dessen Aliasnamen Sie ändern möchten und betätigen Sie die rechte Maustaste.

Folgendes Kontextmenü kommt zur Anzeige:

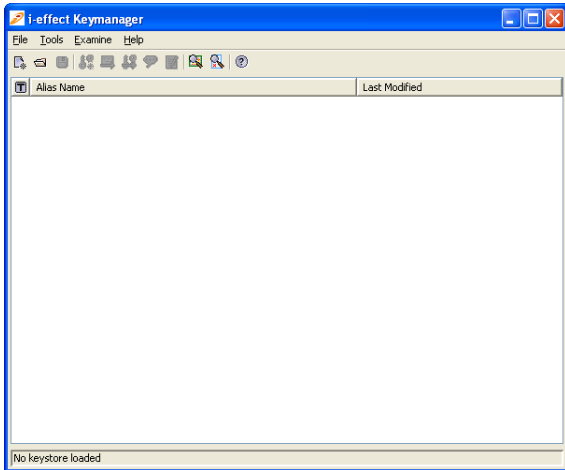


- Nachdem Sie nun den Menüpunkt Rename ausgewählt haben, wird Ihnen das unten abgebildete Dialogfenster angezeigt. Tragen Sie hier bitte den neuen Aliasnamen ein und bestätigen Sie das Dialogfenster mit *Ok*.



## Erstellung eines neuen Keystores

- 1) Starten Sie i-effect Keymanager.  
Das unten abgebildete Fenster kommt zur Anzeige:



- 2) Um einen neuen Keystore anzulegen, gehen Sie wie folgt vor:  
*File* -> *New Keystore (Strg + N)*  
Folgendes Dialogfenster öffnet sich:



Folgende Keystore-Formate werden unterstützt:

- o JKS: Java Keystore (Sun Keystore Format)
  - o PKCS#12: Public-Key Cryptography Standards #12 Keystore (RSA's Personal Information Exchange Syntax Standard)
  - o JCEKS: Java Cryptography Extension Keystore (More secure version of JKS)
  - o BKS: Bouncy Castle Keystore (Bouncy Castle's version of JKS)
  - o UBER: Bouncy Castle UBER Keystore (More secure version of BKS)
- 3) Wählen Sie hier bitte das Keystore Format PKCS#12 aus und bestätigen Sie mit OK. Sie kommen zurück zu dem Hauptfenster. Nun können Sie den neu angelegten Keystore speichern, indem Sie folgendermaßen vorgehen:

*File* -> *Save Keystore / Save Keystore (Strg + S)*

Sie werden dazu aufgefordert, ein Passwort für den neu angelegten Keystore zu vergeben.



- 4) Nachdem Sie hier das Passwort eingegeben und bestätigt haben, können Sie den angelegten Keystore im Dateisystem Ihrer iSeries speichern (Vergessen Sie hier bitte nicht, die Dateiendung entsprechend dem gewählten Keystore-Format mit anzugeben). Nach dem Speichern kehren Sie automatisch zum Hauptfenster zurück.

