A-Trust Gesellschaft für Sicherheitssysteme
im elektronischen Datenverkehr GmbH
Landstraßer Hauptstraße 5
A-1030 Wien

https://www.a-trust.at
E-Mail: office@a-trust.at

a.sign RK EXE

Developer Manual

Version: 0.8
Datum: 9. Dezember 2016

Copyright

Leistungsbeschreibung

A-Trust stellt ein Executable zur Verfügung welches die Funktionen zum Zugriff auf die a.sign RK CHIP vereinfacht. Weiters werden Funktionen für den AES-Schlüssel, Base64 Kodierung und Sha256 Hash bereitgestellt, welche für die Implementierung der Registrierkassensicherheitsverordnung [Bun15] benötigt werden.

Bereitgestellte Funktionen und Programme:

- Schnittstelle zum Zugriff auf die Chipkarte.
- Funktion für JWS-Signatur der aufbereiteten Datenstruktur des Belegs
- Funktion zur Base64 und Base64-URL Kodierung von String-Werten
- Funktionen zum Generieren und Verwenden des AES-Schlüssel zur Verschlüsselung des Umsatzzählers.
- Sha256 Hash Funktion
- Funktionen zum Generieren eines QR-Codes
- Developer Handbuch mit Funktionsbeschreibung und Beispielaufrufen.

Inhaltsverzeichnis

Datum	Rev	Autor	Änderungen
06.12.2016	0.9	Patrick Hagelkruys	Fehlerkorrekturlevel für QR-Code
29.09.2016	0.8	Patrick Hagelkruys	Base64 zu Base32 Kodierung Beschreibung scalefactor, margin, dpi
13.06.2016	0.7	Patrick Hagelkruys	OCR Funktionen hinzugefügt
12.05.2016	0.6	Patrick Hagelkruys	Umbenennung der Produkte
13.04.2016	0.5	Patrick Hagelkruys	Base64 url Padding Parameter
11.04.2016	0.4	Patrick Hagelkruys	Dokumentation erweitern
30.03.2016	0.3	Patrick Hagelkruys	QR-Code Funktionen
09.03.2016	0.2	Ramin Sabet Patrick Hagelkruys	Internal Review
09.03.2016	0.1	Patrick Hagelkruys	Erste Version

Tabelle 1: Dokumentenhistorie

1 Überblick

1.1 Zusammenfassung

Ziel dieses Dokumentes ist die Beschreibung der Schnittstelle der a.sign RK EXE.

Die a.sign RK EXE kapselt die Aufrufe zur Erstellung von digitalen Signaturen, wie diese in der österreichischen Registrierkassen Sicherheitsverordnung [Bun15] benötigten werden.

Diese Dokumentation entspricht der Version 1.9.0.0 der a.sign RK EXE.

1.2 Voraussetzungen

Für die Verwendung der a.sign RK EXE sind folgende Voraussetzungen zu erfüllen:

- Windows basiertes Betriebssystem (Windows Vista oder neuer)
- a.sign Client in der Version 1.3.2.29c oder neuer
- Kartenleser
- aktivierte a.sign RK CHIP

2 Verwendung der a.sign RK EXE

Das Executeable wird über Kommandozeilenparameter gesteuert und retourniert die Ergebnisse der Verarbeitung wahlweise über die Standard Konsolen Ausgabe oder in eine Datei. Zur Ausgabe der Ergebnisse in eine Datei ist der Parameter `-outfile` zu verwenden.

```
asignRKEXE.exe --cardinfo
```

Ausgabe: Standard Konsole

```
asignRKEXE.exe --cardinfo --outfile c:\temp\test.txt
```

Ausgabe: Datei

Eine erfolgreiche Abarbeitung des Programms kann durch den Exit-Code des Programms überprüft werden. Ein Exit-Code von 0 bedeutet erfolgreiche Abarbeitung, alle anderen Wert kennzeichnen Fehler.

```
asignRKEXE.exe --cardinfo --outfile c:\temp\test.txt
echo %errorlevel%
```

Überprüfung Exit-Code des Programms

2.1 Schnittstelle Registrierkassen Karte - Methoden und Eigenschaften

2.1.1 Hilfe

Wird das Executeable ohne Parameter aufgerufen so werden die möglichen Parameter angezeigt:

```
asignRKEXE.exe
```

Programm ohne Parameter

```
Verfuegbare Parameter:
  --help                          Hilfetext ausgeben
  --check_software                Softwareinstallation ueberpruefen
  --check_card                    Karte ueberpruefen
  --sign arg                      Signatur mit Karte, Eingabe Format ...
...
```

Ausgabe der Parameterübersicht

Die komplette Ausgabe dieses Befehls ist in Anhang A abgebildet.

2.1.2 Software prüfen

Dieser Befehl prüft ob die notwendige a.sign Client Software in der richtigen Version installiert ist.

```
asignRKEXE.exe --check_software
```

Software überprüfen

Programm Exit-Code:

0 OK

2 Registry Einträge des a.sign Client fehlen. Fehlerhafte Installation?

3 a.sign Client Version nicht ausreichend, bitte aktualisieren

4 a.sign Client kann nicht geladen werden. Fehlerhafte Installation?

5 Allgemeiner Fehler

2.1.3 Karte prüfen

Dieser Befehl prüft ob eine Karte im Kartenleser ist.

```
asignRKEXE.exe --check_card
```

Karte überprüfen

Programm Exit-Code:

0 OK

1 Keine aktivierte Karte gefunden

2 Keine Karte gefunden.

2.1.4 Karteninformationen lesen

Laden der Zertifikatsdaten von der Karte.

```
asignRKEXE.exe --cardinfo
```

Karteninformationen lesen

Die Ausgabe besteht aus einer mit Strichpunkt getrennten Liste welche folgende Daten enthält:

- ZDA-ID
- Zertifikatsseriennummer (dezimal)
- Zertifikat

- Ausstellerzertifikat

```
AT1;1634338;MII...pQ==;MIIF...JCw==
```

Ausgabe: der Karteninformationen

Programm Exit-Code:

0 OK

1 a.sign Client nicht initialisiert

2 Fehler in a.sign Client

2.1.5 ZDA-ID auslesen

Laden der ZDA-ID von der Karte.

```
asignRKEXE.exe --zdaid
```

ZDA-ID lesen

```
AT1
```

Ausgabe: ZDA-ID lesen

Programm Exit-Code:

0 OK

1 a.sign Client nicht initialisiert

2 Fehler in a.sign Client

2.1.6 Zertifikatsseriennummer lesen

Laden der Zertifikatsseriennummer von der Karte.

```
asignRKEXE.exe --certserial
```

Zertifikatsseriennummer lesen

```
1634338
```

Ausgabe: Zertifikatsseriennummer lesen

Programm Exit-Code:

0 OK

1 a.sign Client nicht initialisiert

2 Fehler in a.sign Client

2.1.7 Zertifikatsseriennummer lesen (hexadezimal)

Laden der Zertifikatsseriennummer im hexadezimal Format von der Karte.

```
asignRKEXE.exe --certserialhex
```

Zertifikatsseriennummer lesen

```
7684d8f2
```

Ausgabe: Zertifikatsseriennummer lesen

Programm Exit-Code:

0 OK

1 a.sign Client nicht initialisiert

2 Fehler in a.sign Client

2.1.8 Zertifikat lesen

Laden des Zertifikats von der Karte.

```
asignRKEXE.exe --certificate
```

Zertifikat lesen

```
MIIE1zCCA7...OIvPU2pQ==
```

Ausgabe: Zertifikat lesen

Programm Exit-Code:

0 OK

1 a.sign Client nicht initialisiert

2 Fehler in a.sign Client

2.1.9 Ausstellerzertifikat lesen

Laden der Ausstellerzertifikat von der Karte.

```
asignRKEXE.exe --issuer
```

Ausstellerzertifikat lesen

```
MIIF9TCCA...XtdkupJCw==
```

Ausgabe: Ausstellerzertifikat lesen

Programm Exit-Code:

0 OK

1 a.sign Client nicht initialisiert

2 Fehler in a.sign Client

2.1.10 Gültigkeit lesen

Auslesen der Gültigkeit des Zertifikates, der Rückgabewert ist ist als UTC-Zeit zu interpretieren.

```
asignRKEXE.exe --gueltigkeit
```

Gültigkeit lesen

```
2020-12-30T22:00:00
```

Ausgabe: Gültigkeit lesen

Programm Exit-Code:

0 OK

1 a.sign Client nicht initialisiert

2 Fehler in a.sign Client

2.1.11 SignJWS

Durchführen einer Signatur auf der Karte. Die Funktion bereitet die eingegebenen Daten nach dem JWS Standard auf, d.h. es wird der entsprechende JWS-Header mit dem Algorithmus erzeugt und sowohl Daten als auch Header Base64-URL kodiert. Der zurückgegebene Wert entspricht der JWS Signatur bestehend aus Protected Header, Payload und Signatur jeweils Base64-URL kodiert und durch Punkt getrennt.
Die Eingabedaten müssen entsprechende der österreichischen Registrierkassensicherheitsverordnung [Bun15, Detailspezifikation Kapitel 5] formatiert werden. Beispiele, Testdatensätze und Prüftools sind unter [A-S16a] verfügbar.

```
asignRKEXE.exe --signjws _R1-AT1_1_1_2016-0...XdnO+I=
```

Signatur JWS durchführen

```
eyJhbGciOiJFUzI1NiJ9.X1IxLUFUMV8x...F9xZFhvb1hkbk8rST0=.gON2gQR...cpQZnyF0Kw==
```

Ausgabe: Signatur JWS durchführen

Programm Exit-Code:

0 OK

1 a.sign Client nicht initialisiert

2 Fehler beim Signieren

2.1.12 Sign

Es wird empfohlen die Funktion SignJWS aus Kapitel 2.1.11 zu verwenden. Für die Verwendung der hier angeführten Funktion müssen die zu signierenden Daten selbst aufbereitet werden.
Durchführen einer Signatur auf der Karte, der zurückgegebenen Wert ist bereits Base64-URL kodiert. Die Eingabedaten müssen bereits die Form `[protected header].[payload]` entsprechend dem JWS-Standard [Jon15] haben.

```
asignRKEXE.exe --sign eyJhbGciOiJFUzI1NiJ9.X1IxLU...rST0
```

Signatur durchführen

```
vxQbHXIuMX...lLreEQ==
```

Ausgabe: Signatur durchführen

Programm Exit-Code:

0 OK

1 a.sign Client nicht initialisiert

2 Fehler beim Signieren

2.2 Schnittstelle AES ICM - Methoden und Eigenschaften

2.2.1 GenerateKey

Generieren eines AES Schlüssel. Dieser Befehl muss nur einmal pro Kasse durchgeführt werden und das Ergebnis durch den Aufrufenden gespeichert werden.

```
asignRKEXE.exe --aes_generate
```

AES Schlüssel generieren

```
qyDC9hlN5MAvum2K/MUu+eVSxB6OzfIShT4o3vTcWtg=
```

Ausgabe: AES Schlüssel generieren

Programm Exit-Code:

0 OK

1 Fehler

2.2.2 Verschlüsseln

Verschlüsselung des Umsatzzählers. Der Umsatz muss in Euro-Cent Beträgen angegeben werden, weitere Informationen dazu unter [A-S16b, Brutto vs. Netto].

```
asignRKEXE.exe --aes_encrypt
      --aes_key qyDC9hlN5MAvum2K/MUu+eVSxB6OzfIShT4o3vTcWtg=
      --umsatz 123412
      --kassenid cash-reg-1
      --belegnummer adsf51
```

AES verschlüsseln

```
FIKrRXdGO6f76zg6zneiQg==
```

Ausgabe: AES verschlüsseln

Programm Exit-Code:

0 OK

1 Fehler

2.2.3 Entschlüsseln

Entschlüsselung des Umsatzzählers. Diese Funktion wird im Regelfall nicht benötigt.

```
asignRKEXE.exe --aes_decrypt
      --aes_key qyDC9hlN5MAvum2K/MUu+eVSxB6OzfIShT4o3vTcWtg=
      --encrypted_data FIKrRXdGO6f76zg6zneiQg==
      --kassenid cash-reg-1
      --belegnummer adsf51
```

AES entschlüsseln

```
123412
```

Ausgabe: AES entschlüsseln

Programm Exit-Code:

0 OK

1 Fehler

2.3 Schnittstelle Base64 - Methoden und Eigenschaften

2.3.1 Encode

Base64 Encoding eines String, mit dem optionalen Parameter `-padding` kann das Padding eingestellt werden.

```
asignRKEXE.exe --padding 1 --base64_encode test
```

Base64 kodieren eines Strings

```
dGVzdA==
```

Ausgabe: Base64 kodieren eines Strings

Programm Exit-Code:

0 OK

1 Fehler

2.3.2 EncodeUrl

Base64-URL Encoding eines String, mit dem optionalen Parameter `-padding` kann das Padding eingestellt werden.

```
asignRKEXE.exe --padding 0 --base64url_encode test
```

Base64Url kodieren eines Strings

```
dGVzdA
```

Ausgabe: Base64Url kodieren eines Strings

Programm Exit-Code:

0 OK

1 Fehler

2.3.3 Reencode Base64-URL to Base64

Decodiert einen Base64-URL kodierten String und kodiert diesen neu als Base64 (Normal). Mit dem optionalen Parameter `-padding` kann das Padding eingestellt werden.

```
asignRKEXE.exe --padding 1 --base64_url_to_normal_reencode dGVzdA==
```

Base64url kodieren zu Base64

```
dGVzdA==
```

Ausgabe: Base64url kodieren zu Base64

Programm Exit-Code:

0 OK

1 Fehler

2.3.4 Reencode Base64 to Base64-URL

Decodiert einen Base64 (Normal) kodierten String und kodiert diesen neu als Base64-URL. Mit dem optionalen Parameter `-padding` kann das Padding eingestellt werden.

```
asignRKEXE.exe --padding 0 --base64_normal_to_url_reencode dGVzdA==
```

Base64 kodieren zu Base64url

```
dGVzdA
```

Ausgabe: Base64 kodieren zu Base64url

Programm Exit-Code:

0 OK

1 Fehler

2.3.5 Reencode Bease64 to Base32

Decodiert einen Base64 (Normal) kodierten String und kodiert diesen neu als Base32.

```
asignRKEXE.exe --base64_normal_to_base32_reencode dGVzdA==
```

Base64 kodieren zu Base32

```
ORSXG5A=
```

Ausgabe: Base64 kodieren zu Base32

Programm Exit-Code:

0 OK

1 Fehler

2.3.6 Reencode Bease64-URL to Base32

Decodiert einen Base64 (URL) kodierten String und kodiert diesen neu als Base32.

```
asignRKEXE.exe --base64_url_to_base32_reencode dGVzdA==
```

Base64-URL kodieren zu Base32

```
ORSXG5A=
```

Ausgabe: Base64-URL kodieren zu Base32

Programm Exit-Code:

0 OK

1 Fehler

2.4 Schnittstelle Sha256 - Methoden und Eigenschaften

2.4.1 HashString

Sha256 eines String, Ausgabe ist bereits Base64 kodiert

```
asignRKEXE.exe --hash_string test
```

Sha256 Hash eines Strings

```
n4bQgYhMfWWaL+qgxVrQFaO/TxsrC4Is0V1sFbDwCgg=
```

Ausgabe: Sha256 Hash eines Strings

Programm Exit-Code:

0 OK

1 Fehler

2.4.2 HashSigVorigerBeleg

Sha256 des vorigen Belegs wie in [Bun15, Z4, Sig-Voriger-Beleg] verlangt

```
asignRKEXE.exe --hash_sig_voriger_beleg test
```

Hash Signatur voriger Beleg

Zusätzlich kann die Anzahl der zu extrahierenden angegeben werden

```
asignRKEXE.exe --hash_sig_voriger_beleg test --bytes_extrahiert 8
```

Hash Signatur voriger Beleg mit Angabe der zu extrahierenden Bytes

```
n4bQgYhMfWU=
```

Ausgabe: Hash Signatur voriger Beleg

Programm Exit-Code:

0 OK

1 Fehler

2.5 QR-Code - Methoden und Eigenschaften

2.5.1 QR-Code erstellen aus Belegzeile

```
asignRKEXE.exe --qrcode --qrdata "R1-AT1_DEMO-CA...Q2===" --scalefactor 2 --margin 3
    --dpi 24 --errorcorrection H --outfile qr1.bmp
```

QR-Code erstellen aus Buchungszeile

Abbildung 1: QR-Code Ausgabe (qr1.bmp)

Programm Exit-Code:

0 OK

1 Fehler

2.5.2 QR-Code erstellen aus JWS Zeile

```
asignRKEXE.exe --qrcode --qrdata_jws "eyJhbGciOi...CXH" --scalefactor 3 --margin 10
    --dpi 24 --errorcorrection M --outfile qr2.bmp
```

QR-Code erstellen aus JWS Zeile

Programm Exit-Code:

0 OK

1 Fehler

Abbildung 2: QR-Code Ausgabe (qr2.bmp)

2.5.3 Beschreibung der Parameter scalefactor, margin, dpi

Bei den Befehlsaufrufen zur QR-Code Erstellung kann mittels der Paramter scalefactor, margin und dpi die erstellte Bilddatei beeinflusst werden.

scalefactor: Skalierungsfaktor für QR-Code. Der QR-Code wird bei einem Skalierungsfaktor von 1 als 77x77 Pixel ausgegeben und entsprechend des Faktors vergrößert.

maring: Rand für QR-Code. Entsprechend dem übergebenen Wert werden weiße Pixel an allen Seiten eingefügt.

dpi: Farbtiefe für den QR-Code in bit. Mögliche Werte sind 1,4,8,16,24,32.

errorcorrection: Fehlerkorrekturlevel für den QR-Code. Mögliche Werte sind L,M,Q,H.

Level L (Low): ca. 7% der Daten können wiederhergestellt werden

Level M (Medium): ca. 15% der Daten können wiederhergestellt werden

Level Q (Quartile): ca. 25% der Daten können wiederhergestellt werden

Level H (High): ca. 30% der Daten können wiederhergestellt werden

2.6 OCR-Code - Methoden und Eigenschaften

Für den OCR-Code ist in der RKSV [Bun15, Detailspezifikation Kapitel 14] beschreiben, dass die Base64 Werte im Base32 Format kodiert werden müssen.

2.6.1 OCR-Code erstellen aus Belegzeile

```
asignRKEXE.exe --ocrcode --ocrdata "_R1-AT1_DEMO-CA...Q2==="
```

OCR-Code erstellen aus Buchungszeile

```
_R1-AT1_DEMO-CASHBOX_r1_2016-03-11T04:24:43_0,00_0,00_0,00_0,00_0,00_4K6WEIOWIZ4AI===_...
```

Ausgabe: OCR-Code

Programm Exit-Code:

0 OK

1 Fehler

2.6.2 OCR-Code erstellen aus JWS Zeile

```
asignRKEXE.exe --ocrrcode --ocrdata_jws "eyJhbGciOi...CXH"
```

OCR-Code erstellen aus JWS Zeile

```
_R1-AT1_DEMO-CASHBOX_r1_2016-03-11T04:24:43_0,00_0,00_0,00_0,00_0,00_4K6WEIOWIZ4AI===_...
```

Ausgabe: OCR-Code

Programm Exit-Code:

0 OK

1 Fehler

2.7 Logging

Zur Fehleranalyse kann das Logging des Executeables aktiviert werden, dazu müssen in der Registry die entsprechenden Werte eingetragen werden.

```
HKEY_LOCAL_MACHINE\SOFTWARE\A-Trust GmbH\ATrustRegistrierkasseCom
```

Registry Pfad für 32-bit Systeme

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\A-Trust GmbH\ATrustRegistrierkasseCom
```

Registry Pfad für 64-bit Systeme

Nachfolgend die Werte für das Aktivieren des Logging.

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\A-Trust GmbH\ATrustRegistrierkasseCom]
"LogPath"="c:\\temp\\asignrkexe.log"
"Log"=dword:00000001
```

Registry Werte

A Ausgabe der Hilfe

```
Verfuegbare Parameter:
  --help                                Hilfetext ausgeben
  --check_software                      Softwareinstallation ueberpruefen
  --check_card                          Karte ueberpruefen
  --sign arg                            Signatur mit Karte, Eingabe Format [JWS
                                        protected Header].[Payload], Ausgabe
                                        Format Base64
  --signjws arg                         Signatur JWS mit Karte, Eingabe Format
                                        Payload, Ausgabe Format [JWS protected
                                        Header].[Payload].[Signatur]
  --cardinfo                            Karteninformationen auslesen,
                                        Ausgabeformat ZdaId;Seriennummer;Signatur
                                        zertifikat;Ausstellerzertifikat
  --zdaid                               Karteninformationen auslesen,
                                        Ausgabeformat ZdaId
  --certserial                          Karteninformationen auslesen,
                                        Ausgabeformat Seriennummer
  --certserialhex                       Karteninformationen auslesen,
                                        Ausgabeformat Seriennummer (HEX)
  --certificate                         Karteninformationen auslesen,
                                        Ausgabeformat Signaturzertifikat
  --issuer                              Karteninformationen auslesen,
                                        Ausgabeformat Ausstellerzertifikat
  --gueltigkeit                         Gueltigkeit des Zertifikates
  --aes_generate                        AES Schluessel generieren, Ausgabeformat
                                        AES-Schluessel in Base64
  --aes_encrypt                         AES Verschluesselung
  --aes_decrypt                         AES Entschluesseln
  --aes_key arg                         AES Schluessel
  --umsatz arg                          Umsatz
  --kassenid arg                        KassenId
  --belegnummer arg                     Belegnummer
  --encrypted_data arg                  Verschluesselter Umsatzzaehler
  --base64_encode arg                   Base64 Encode
  --base64url_encode arg                Base64URL Encode
  --padding arg                         padding fuer base64
  --base64_normal_to_url_reencode arg   Base64 Normal zu URL umkodieren
  --base64_url_to_normal_reencode arg   Base64 URL zu Normal umkodieren
  --hash_string arg                     Sha256 der Eingabedaten, Ausgabe in
                                        Base64 Format
  --hash_sig_voriger_beleg arg          Sha256 der Signatur des Vorigen Beleges,
                                        Ausgabe in Base64 Format
  --bytes_extrahiert arg                Zu extrahierende Byteanzahl aus Signatur
                                        des Vorigen Beleges
  --outfile arg                         Ausgabe in eine Datei anstelle der
                                        Konsole
  --qrcode                              QR-Code erstellen, outfile muss angegeben
                                        werden
  --qrdata arg                          QR-Daten
  --qrdata_jws arg                      QR-Daten im JWS Format
  --scalefactor arg                     Skalierungsfaktor fuer QR-Code
  --margin arg                          Margin fuer QR-Code
  --ocrcode                             OCR-Code vorberieten
  --ocrdata arg                         OCR-Daten
  --ocrdata_jws arg                     OCR-Daten im JWS Format
```

Ausgabe der Parameterübersicht

Literatur

[A-S16a] A-SIT Plus GmbH: *a-sit-plus/at-registrierkassen-mustercode*, 2016. https://github.com/a-sit-plus/at-registrierkassen-mustercode/, besucht: 2016-03-09.

[A-S16b] A-SIT Plus GmbH: *Erläuterungen FAQ - a-sit-plus/at-registrierkassen-mustercode*, 2016. https://github.com/a-sit-plus/at-registrierkassen-mustercode/wiki/Erl%C3%A4uterungen-FAQ, besucht: 2016-04-11.

[Bun15] Bundesministers für Finanzen: *Verordnung des Bundesministers für Finanzen über die technischen Einzelheiten für Sicherheitseinrichtungen in den Registrierkassen und andere, der Datensicherheit dienende Maßnahmen (Registrierkassensicherheitsverordnung, RKSV)*, 2015. https://www.bmf.gv.at/steuern/RKSV.pdf, besucht: 2015-11-16.

[Jon15] Jones, M.: *JSON Web Algorithms (JWA)*. RFC 7518, May 2015. https://tools.ietf.org/html/rfc7518, besucht: 2015-11-25.