

a.trust Empfehlungen für die Erstellung sicherer Signaturen

Die von a.trust empfohlenen Komponenten und Formate für sichere Signaturen behandeln eine qualitätsgesicherte Arbeitsumgebung der Zertifikatsinhaberin/des Zertifikatsinhabers, die/der mit einer von a.trust ausgestellten Signaturerstellungseinheit (Smartcard) eine sichere digitale Signatur erstellt.

Sichere Signaturerstellungseinheiten (Signaturkarten)

- a.sign Premium Variante 1
Smart Card mit Chip Philips Smart Card Controller P5CC036 und Betriebssystem ACOS EMV-A03V0, Configuration B und Digital Signature Application a-sign Premium
- a.sign Premium Variante 2
Smart Card mit Chip Philips Smart Card Controller P8WE5032V0G und Betriebssystem STARCOS SPK 2.3 und Digital Signature Application TrustSign
- trust|sign Karte
Smart Card mit Chip Philips Smart Card Controller P8WE5032V0G und Betriebssystem STARCOS SPK 2.3 und Digital Signature Application TrustSign
seit 23.12.2004 werden keine neuen trust|sign Karten mehr ausgestellt
- a.sign Premium WU-Karte
Prozessorchipkarte mit Smart Card IC SLE 66CX320P und Betriebssystem CardOS/M4.01 (Version C803) und Applikation für digitale Signatur Version 0.20
wird nur an Studierende und Mitarbeiter der Wirtschaftsuniversität Wien ausgegeben

Das Hauptaugenmerk der a.trust Empfehlung wurde auf die folgenden Hauptaspekte gelegt:

- PIN Eingabe
- Hashverfahren
- Vertrauenswürdige Anzeige und Dokumentenformate

Diese Liste wird stets aktuell gehalten und stellt die jeweils am Markt verfügbaren und von a.trust empfohlenen Produkte zur Erstellung sicherer Signaturen dar. Die a.trust Empfehlung umfasst die Kompatibilität der angeführten Komponenten mit den Smart Card - und Zertifikats-Produkten von a.trust, sowie die Korrektheit der Zertifizierungen und Bescheinigungen.

D. h., dass die angeführten Produkte schon im Rahmen der Evaluierungen auf das Zusammenwirken mit a.trust Produkten geprüft wurden. Um sicherzustellen, dass die unterschiedlichen Komponenten (wie z. B. Kartenleser und Viewer) miteinander kompatibel sind, müssen dazu unbedingt die Informationen der Hersteller berücksichtigt werden.

PIN Eingabe

Die PIN (Personal Identification Number) ist eine Ziffernfolge, die auch als Aktivierungsdaten für die Signaturerstellung bezeichnet wird. a.trust empfiehlt Kartenlesegeräte mit eigenem Nummern-eingabefeld für die sichere PIN-Eingabe.

Nachfolgend finden Sie die von a.trust empfohlenen Kartenlesegeräte:

Für a.sign Premium WU-Karten:

- Sign@tor Terminal Version 1.0 oder Version 2.0 von Siemens AG (Chipkartenterminal mit Tastatur und Display)

Für a.sign Premium Standardkarten Variante 1 und Variante 2 sowie für trust|sign Karten:

- Kobil KAAN Professional Firmware Version 2.0.8
- Kobil KAAN Standard Plus Firmware Version 02121852
- Cherry PC-Tastaturen mit integriertem Chipkartenterminal
G83-6700LQZxx/01
G81-7015LQZxx/01
G81-8015LQZxx/01
G81-12000LTZxx/01
- Chipkartenterminal der Familie Cherry Smartboard xx44, Firmware Version 1.04
- cyberJack e-com Chipkartenleser mit Tastatur und Display Firmware Version 2.0
- cyberJack pinpad Chipkartenleser mit Tastatur Firmware Version 2.0 und 3.0
- Towitoko CHIPDRIVE® pinpad (SCM Microsystems SPR532) Firmware Version 4.15
- Siemens AG Sign@tor Terminal Version 2.0

Für a.sign Premium Standardkarten Variante 1 und Variante 2:

- Chipkartenterminal der Familie CardMan Trust CM3621/CM3821, Firmware Version 6.00 der Omnikey GmbH

Hashverfahren

Unter Hashverfahren versteht man mathematische bzw. kryptografische Verfahren. Die angeführten Produkte wurden auf die einwandfreie Implementierung dieser Verfahren geprüft.

Hinweis: Diese Produkte können häufig deckungsgleich mit denen der sicheren Anzeige sein!

Nachfolgend finden Sie die von a.trust empfohlenen Produkte:

Für a.sign Premium Karten Variante 1:

- hot:PDFSign Version 1.00 (BDC EDV-Consulting GmbH)
- trustview secure viewer (IT Solution GmbH) 2.1.1 rel. 5 und rel. 6
- Signatursoftware MBS Modul zur Erstellung sicherer Signaturen Version 2.0 Release 1.3 (BDC EDV-Consulting GmbH)
- BDC hot:Sign, Signatur-Client zur Erstellung sicherer Signaturen Version 1.3 (BDC EDV-Consulting GmbH)

Für a.sign Premium Karten Variante 2:

- hot:PDFSign Version 1.00 (BDC EDV-Consulting GmbH)
- trustview secure viewer (IT Solution GmbH) 2.1.1 rel. 5 und rel. 6
- Signatursoftware MBS Modul zur Erstellung sicherer Signaturen Version 2.0 Release 1.2 und Version 2.0 Release 1.3 (BDC EDV-Consulting GmbH)
- BDC hot:Sign, Signatur-Client zur Erstellung sicherer Signaturen Version 1.2 SR4 rev.3 und rev.5 und Version 1.3 (BDC EDV-Consulting GmbH)
- SecSigner ® 2.0.0 Signaturanwendungskomponente (SecCommerce Informationssysteme GmbH, Hamburg)

Für a.sign Premium WU-Karten:

- trustview secure viewer (IT Solution GmbH) Version Version 2.1.1 rel. 5 und rel. 6
- BDC hot:Sign, Signatur-Client zur Erstellung sicherer Signaturen Version 1.3 (BDC EDV-Consulting GmbH)

Für trust|sign Karten:

- hot:PDFSign Version 1.00 (BDC EDV-Consulting GmbH)
- trustview secure viewer (IT Solution GmbH) 2.1.1 rel. 5 und rel. 6
- Signatursoftware MBS Modul zur Erstellung sicherer Signaturen Version 2.0 Release 1.2 und Version 2.0 Release 1.3 (BDC EDV-Consulting GmbH)
- BDC hot:Sign, Signatur-Client zur Erstellung sicherer Signaturen Version 1.2 SR4 rev.3 und rev.5 und Version 1.3 (BDC EDV-Consulting GmbH)
- SecSigner ® 2.0.0 Signaturanwendungskomponente (SecCommerce Informationssysteme GmbH, Hamburg)

Vertrauenswürdige Anzeige und Dokumentenformate

Unter vertrauenswürdiger Anzeige versteht man Produkte, die gewährleisten, dass nur die dem Signator dargestellten Daten auch tatsächlich signiert werden. Es werden auch die empfohlenen Dokumentenformate, die von diesen Produkten sicher angezeigt werden können, angeführt.

Nachfolgend finden Sie die von a.trust empfohlenen Produkte:

- **trustview secure viewer (IT Solution GmbH) Versionen: 2.1.1 rel. 5 und rel. 6**

Hinweis: Das genannte Produkt ist auch für die sichere Signaturprüfung geeignet!

trustview benutzt XML als Dokumentenformat. Die zu signierenden bzw. zu prüfenden Dokumente entsprechen dem SecurityLayer Viewerformat 1.2 des Bundes laut folgender Spezifikation:

<http://www.buergerkarte.at/konzept/securitylayer/spezifikation/aktuell/viewerformat/ViewerFormat.html>

oder folgender Spezifikation:

```
<?xml version="1.0" encoding="UTF-8" ?>
<Document Height="520" Width="640">
  <Data Id="SignedData">
    <Text X="10" Y="10">Bestätigung</Text>
    <Text X="10" Y="58">Ich bestätige hiermit ... Angaben:</Text>
    <Vorname X="10" Y="90">Rainer</Vorname>
    <Nachname X="10" Y="122">Gundacker</Nachname>
    <Datum X="10" Y="170">31.01.2002 09:28.41 GMT+00</Datum>
  </Data>
  <Signature>
    <SignedInfo>
      <SignatureMethod Algorithm="rsa-sha1" />
      <Reference URI="#SignedData">
        <DigestMethod Algorithm="sha1" />
        <DigestValue>ErcBymw90D ... W1wlulQ=</DigestValue>
      </Reference>
    </SignedInfo>
    <SignatureValue>HIX2 ... JKHsnbYlenyKQ=</SignatureValue>
    <KeyInfo>
      <X509Data>
        <X509Certificate> +iEtCIZwj ... e7Hoqh</X509Certificate>
      </X509Data>
    </KeyInfo>
  </Signature>
</Document>
```

Das XML Dokument selbst kann vier verschiedene Arten von XML Tags enthalten:

- Text Tags mit Positionierungsattributen (X/Y Koordinate)
Beispiel: <Text X="10" Y="10">**Bestätigung**</Text>
- Daten Tags mit Positionierungsattributen (X/Y Koordinate)
Beispiel: <Vorname X="10" Y="90">**Rainer**</Vorname>
- Image Tags mit Positionierungsattributen (X/Y Koordinate)
Beispiel: <Image X="450" Y="20">**Qk3EJwJk ... AAA==**</Image>
Image Tags enthalten immer eine Bitmap (Windows BMP Format) als Daten
- einen Datum Tag mit Positionierungsattributen (X/Y Koordinate)
Beispiel: <Datum X="10" Y="170">**31.01.2002 09:28.41 GMT+00**</Datum>

- **Signatursoftware MBS Modul zur Erstellung sicherer Signaturen Version 2.0 Release 1.2 und Version 2.0 Release 1.3 (BDC EDV-Consulting GmbH)**

Character set

Das MBS Modul zur Erstellung sicherer Signaturen benutzt als Character set ein eingeschränktes ISO 8859-1.

Als erlaubte Zeichen sind folgende spezifiziert:

Zeichen	Hexwert
Linefeed	0x0a
Space	0x20
#	0x23
-	0x2d
.	0x2e
0-9	0x30-0x39
A-Z	0x41-0x5a
a-z	0x61-0x7a
Ä	0xc4
Ö	0xd6
Ü	0xdc
ß	0xdf
ä	0xe4
ö	0xf6
ü	0xfc

- **BDC hot:Sign, Signatur-Client zur Erstellung sicherer Signaturen, Version 1.2 SR4 rev.3 und rev.5 sowie Version 1.3 (BDC EDV-Consulting GmbH)**

Die folgenden Datenformate werden von hot:Sign unterstützt:

text/plain

Die Daten werden ohne Interpretation der Formatierung angezeigt. Bei der Interpretation des Inhalts ist das Character-Set relevant, in dem die Daten vorliegen, dazu kann das Character-Encoding als zusätzlicher Parameter übergeben werden. Wird kein Character-Encoding angegeben, werden die Daten als ISO-8859-1 interpretiert.

Liste aller unterstützten Character-Sets:

ANSI_X3.4-1968 (US-ASCII)	Cp869	Cp1124
Big5	Cp870	Cp1258
Big5-HKSCS	Cp871	Cp1381
Cp037	Cp874	Cp1383
Cp273	Cp875	Cp33722
Cp277	Cp918	EUC_CN
Cp278	Cp921	EUC_JP
Cp280	Cp922	EUC_JP_LINUX
Cp284	Cp930	EUC_KR
Cp285	Cp933	EUC_TW
Cp297	Cp935	ISO2022JP
Cp420	Cp937	ISO8859_1
Cp424	Cp939	ISO8859_2
Cp437	Cp942	ISO8859_3
Cp500	Cp942C	ISO8859_4
Cp737	Cp943	ISO8859_5
Cp775	Cp943C	ISO8859_6
Cp838	Cp948	ISO8859_7
Cp850	Cp949	ISO8859_8
Cp852	Cp949C	ISO8859_1
Cp855	Cp950	ISO8859_9
Cp856	Cp964	ISO8859_13
Cp857	Cp970	ISO8859_15_FDIS
Cp860	Cp1006	MS874
Cp861	Cp1025	MS932
Cp862	Cp1026	SJIS
Cp863	Cp1097	TIS620
Cp864	Cp1098	UTF8
Cp865	Cp1112	UnicodeLittleUnmarked
Cp866	Cp1122	
Cp868	Cp1123	

text/xml

Die Daten werden als XML-Daten interpretiert und die XML-Elemente werden hierarchisch dargestellt. Als Einrückungszeichen werden jeweils vier Leerzeichen verwendet.

Liste der zulässigen Character-Sets:

ANSI_X3.4-1968 (US-ASCII)	EBCDIC-CP-IT	ISO-8859-1
Big5	EBCDIC-CP-NL	ISO-8859-2
EBCDIC-CP-AR1	EBCDIC-CP-NO	ISO-8859-3
EBCDIC-CP-AR2	EBCDIC-CP-ROECE	ISO-8859-4
EBCDIC-CP-CA	EBCDIC-CP-SE	ISO-8859-5
EBCDIC-CP-CH	EBCDIC-CP-US	ISO-8859-6
EBCDIC-CP-DK	EBCDIC-CP-YU	ISO-8859-7
EBCDIC-CP-ES	EUC-JP	ISO-8859-8
EBCDIC-CP-FI	EUC-KR	ISO-8859-9
EBCDIC-CP-FR	GB2312	Shift_JIS
EBCDIC-CP-GB	KOI8-R	UTF-8
EBCDIC-CP-HE	ISO-2022-JP	Windows-31J
EBCDIC-CP-IS	ISO-2022-KR	

text/html

Die Daten werden nach dem HTML-Standard interpretiert und angezeigt, die zulässigen Character-Encodings entsprechen den Character-Sets für „text/xml“. Der HTML-Sprachumfang ist eingeschränkt und die zulässigen HTML-Elemente in einem XML-Schema festgelegt. Wenn die vorliegenden Daten nicht dem spezifiziertem HTML-Sprachumfang entsprechen, wird der Signaturvorgang abgebrochen. Zusätzlich werden die verwendbaren Text- und Hintergrundfarben eingeschränkt.

XML-Schema für den HTML-Sprachumfang:

```
<?xml version="1.0" encoding="UTF-8" ?>
<:xs:schema targetNamespace="http://www.w3.org/1999/xhtml"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="http://www.w3.org/1999/xhtml">
  <:xs:group name="Heading.class">
    <:xs:choice>
      <:xs:choice minOccurs="0" maxOccurs="unbounded">
        <xs:element ref="h1" />
        <xs:element ref="h2" />
        <xs:element ref="h3" />
        <xs:element ref="h4" />
        <xs:element ref="h5" />
        <xs:element ref="h6" />
      </xs:choice>
    </xs:choice>
  </xs:group>
  <:xs:group name="List.class">
    <:xs:choice>
      <:xs:choice minOccurs="0" maxOccurs="unbounded">
        <xs:element ref="ul" />
        <xs:element ref="ol" />
        <xs:element ref="dl" />
      </xs:choice>
    </xs:choice>
  </xs:group>
  <:xs:group name="Block.class">
    <:xs:choice>
      <:xs:choice minOccurs="0" maxOccurs="unbounded">
```

```
<xs:element ref="p" />
<xs:element ref="blockquote" />
<xs:element ref="table" />
<xs:element ref="hr" minOccurs="0" maxOccurs="unbounded" />
<xs:group ref="List.class" />
</xs:choice>
</xs:choice>
</xs:group>
= <xs:group name="Block.mix">
  = <xs:choice>
    = <xs:choice minOccurs="0" maxOccurs="unbounded">
      <xs:group ref="Heading.class" />
      <xs:group ref="Text.class" />
      <xs:group ref="Block.class" />
    </xs:choice>
  </xs:choice>
</xs:group>
= <xs:group name="Flow.mix">
  = <xs:choice>
    = <xs:choice minOccurs="0" maxOccurs="unbounded">
      <xs:group ref="Block.class" />
      <xs:group ref="Text.class" />
    </xs:choice>
  </xs:choice>
</xs:group>
= <xs:group name="Text.class">
  = <xs:choice>
    = <xs:choice minOccurs="0" maxOccurs="unbounded">
      <xs:element ref="font" />
      <xs:element ref="br" />
      <xs:element ref="tt" />
      <xs:element ref="i" />
      <xs:element ref="b" />
      <xs:element ref="big" />
      <xs:element ref="sub" />
      <xs:element ref="sup" />
    </xs:choice>
  </xs:choice>
</xs:group>
= <xs:simpleType name="Color">
  = <xs:restriction base="xs:string">
    <xs:enumeration value="Black" />
    <xs:enumeration value="black" />
    <xs:enumeration value="Silver" />
    <xs:enumeration value="silver" />
    <xs:enumeration value="Gray" />
    <xs:enumeration value="gray" />
    <xs:enumeration value="White" />
    <xs:enumeration value="white" />
    <xs:enumeration value="Maroon" />
    <xs:enumeration value="maroon" />
    <xs:enumeration value="Red" />
    <xs:enumeration value="red" />
    <xs:enumeration value="Purple" />
    <xs:enumeration value="purple" />
    <xs:enumeration value="Fuchsia" />
    <xs:enumeration value="fuchsia" />
    <xs:enumeration value="Green" />
    <xs:enumeration value="green" />
  </xs:restriction>
</xs:simpleType>
```



```
<xs:enumeration value="Lime" />
<xs:enumeration value="lime" />
<xs:enumeration value="Olive" />
<xs:enumeration value="olive" />
<xs:enumeration value="Yellow" />
<xs:enumeration value="yellow" />
<xs:enumeration value="Navy" />
<xs:enumeration value="navy" />
<xs:enumeration value="Blue" />
<xs:enumeration value="blue" />
<xs:enumeration value="Teal" />
<xs:enumeration value="teal" />
<xs:enumeration value="Aqua" />
<xs:enumeration value="aqua" />
</xs:restriction>
</xs:simpleType>
= <xs:simpleType name="FontSize">
  = <xs:restriction base="xs:string">
    <xs:enumeration value="-1" />
    <xs:enumeration value="+1" />
    <xs:enumeration value="+2" />
    <xs:enumeration value="+3" />
    <xs:enumeration value="+4" />
    <xs:enumeration value="2" />
    <xs:enumeration value="3" />
    <xs:enumeration value="4" />
    <xs:enumeration value="5" />
    <xs:enumeration value="6" />
    <xs:enumeration value="7" />
  </xs:restriction>
</xs:simpleType>
= <xs:simpleType name="FontFace">
  = <xs:restriction base="xs:string">
    <xs:enumeration value="Arial" />
    <xs:enumeration value="arial" />
    <xs:enumeration value="Times New Roman" />
    <xs:enumeration value="times new roman" />
    <xs:enumeration value="Verdana" />
    <xs:enumeration value="verdana" />
    <xs:enumeration value="courier new" />
    <xs:enumeration value="Courier New" />
  </xs:restriction>
</xs:simpleType>
= <xs:simpleType name="Length">
  = <xs:restriction base="xs:integer">
    <xs:minInclusive value="0" />
    <xs:maxInclusive value="1000" />
  </xs:restriction>
</xs:simpleType>
= <xs:simpleType name="restrictedLength">
  = <xs:restriction base="xs:string">
    <xs:pattern value="\d{2}%" />
    <xs:pattern value="\d{1}%" />
    <xs:pattern value="100%" />
    <xs:pattern value="\d{1}" />
    <xs:pattern value="\d{2}" />
    <xs:pattern value="\d{3}" />
    <xs:pattern value="\d{4}" />
    <xs:pattern value="10000" />
  </xs:restriction>
</xs:simpleType>
```

```
<xs:pattern value="\d{1}px" />
<xs:pattern value="\d{2}px" />
<xs:pattern value="\d{3}px" />
<xs:pattern value="\d{4}px" />
<xs:pattern value="10000px" />
</xs:restriction>
</xs:simpleType>
<xs:simpleType name="Pixels">
  <xs:restriction base="xs:integer">
    <xs:minInclusive value="0" />
    <xs:maxInclusive value="1000" />
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="Number">
  <xs:restriction base="xs:nonNegativeInteger" />
</xs:simpleType>
<xs:complexType name="empty.type" />
<xs:attributeGroup name="CellVAlign.attrib">
  <xs:attribute name="valign">
    <xs:simpleType>
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="top" />
        <xs:enumeration value="middle" />
        <xs:enumeration value="bottom" />
      </xs:restriction>
    </xs:simpleType>
  </xs:attribute>
</xs:attributeGroup>
<xs:attributeGroup name="CellHAlign.attrib">
  <xs:attribute name="align">
    <xs:simpleType>
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="left" />
        <xs:enumeration value="center" />
        <xs:enumeration value="right" />
      </xs:restriction>
    </xs:simpleType>
  </xs:attribute>
</xs:attributeGroup>
<xs:complexType name="body.type" mixed="true">
  <xs:group ref="Block.mix" />
  <xs:attribute name="bgcolor" type="Color" />
  <xs:attribute name="text" type="Color" />
</xs:complexType>
<xs:element name="body" type="body.type" />
<xs:simpleType name="FPI">
  <xs:restriction base="xs:normalizedString" />
</xs:simpleType>
<xs:complexType name="html.type">
  <xs:sequence>
    <xs:element ref="head" minOccurs="0" />
    <xs:element ref="body" />
  </xs:sequence>
  <xs:attribute name="version" type="FPI" fixed="XHTML1.1" />
</xs:complexType>
<xs:element name="html" type="html.type" />
<xs:element name="title" type="xs:string" />
<xs:group name="head.content">
  <xs:sequence>
```

```

    <xs:element ref="title" minOccurs="0" />
  </xs:sequence>
</xs:group>
- <xs:complexType name="head.type">
  <xs:group ref="head.content" />
</xs:complexType>
<xs:element name="head" type="head.type" />
- <xs:complexType name="blockquote.type" mixed="true">
  <xs:group ref="Block.mix" />
</xs:complexType>
<xs:element name="blockquote" type="blockquote.type" />
- <xs:complexType name="heading.type" mixed="true">
  <xs:group ref="Text.class" />
  <xs:attributeGroup ref="CellHAlign.attrib" />
</xs:complexType>
<xs:element name="h1" type="heading.type" />
<xs:element name="h2" type="heading.type" />
<xs:element name="h3" type="heading.type" />
<xs:element name="h4" type="heading.type" />
<xs:element name="h5" type="heading.type" />
<xs:element name="h6" type="heading.type" />
- <xs:complexType name="p.type" mixed="true">
  <xs:group ref="Text.class" />
  <xs:attributeGroup ref="CellHAlign.attrib" />
</xs:complexType>
<xs:element name="p" type="p.type" />
- <xs:complexType name="dt.type" mixed="true">
  <xs:group ref="Text.class" />
</xs:complexType>
<xs:element name="dt" type="dt.type" />
- <xs:complexType name="dd.type" mixed="true">
  <xs:group ref="Flow.mix" />
</xs:complexType>
<xs:element name="dd" type="dd.type" />
- <xs:group name="dl.content">
  - <xs:choice>
    <xs:element ref="dt" />
    <xs:element ref="dd" />
  </xs:choice>
</xs:group>
- <xs:complexType name="dl.type" mixed="true">
  <xs:group ref="dl.content" minOccurs="1" maxOccurs="unbounded" />
</xs:complexType>
<xs:element name="dl" type="dl.type" />
- <xs:complexType name="li.type" mixed="true">
  <xs:group ref="Flow.mix" />
</xs:complexType>
<xs:element name="li" type="li.type" />
- <xs:complexType name="ol.type">
  - <xs:sequence>
    <xs:element ref="li" minOccurs="1" maxOccurs="unbounded" />
  </xs:sequence>
</xs:complexType>
<xs:element name="ol" type="ol.type" />
- <xs:complexType name="ul.type">
  - <xs:sequence>
    <xs:element ref="li" minOccurs="1" maxOccurs="unbounded" />
  </xs:sequence>
</xs:complexType>

```

```
<xs:element name="ul" type="ul.type" />
= <xs:attributeGroup name="table.attlist">
  <xs:attribute name="width" type="restrictedLength" />
  <xs:attribute name="border" type="Pixels" />
  <xs:attribute name="cellspacing" type="Length" />
  <xs:attribute name="cellpadding" type="Length" />
  <xs:attribute name="bgcolor" type="Color" />
= <xs:attribute name="align">
  = <xs:simpleType>
    = <xs:restriction base="xs:NMTOKEN">
      <xs:enumeration value="left" />
      <xs:enumeration value="center" />
      <xs:enumeration value="right" />
    </xs:restriction>
  </xs:simpleType>
</xs:attribute>
</xs:attributeGroup>
= <xs:group name="table.content">
  = <xs:choice>
    <xs:element ref="tr" minOccurs="1" maxOccurs="unbounded" />
  </xs:choice>
</xs:group>
= <xs:complexType name="table.type" mixed="true">
  <xs:group ref="table.content" />
  <xs:attributeGroup ref="table.attlist" />
</xs:complexType>
<xs:element name="table" type="table.type" />
= <xs:attributeGroup name="tr.attlist">
  <xs:attribute name="bgcolor" type="Color" />
  <xs:attributeGroup ref="CellHAlign.attrib" />
  <xs:attributeGroup ref="CellVAlign.attrib" />
</xs:attributeGroup>
= <xs:group name="tr.content">
  = <xs:choice>
    <xs:element ref="th" minOccurs="0" maxOccurs="unbounded" />
    <xs:element ref="td" minOccurs="0" maxOccurs="unbounded" />
  </xs:choice>
</xs:group>
= <xs:complexType name="tr.type">
  <xs:group ref="tr.content" />
  <xs:attributeGroup ref="tr.attlist" />
</xs:complexType>
<xs:element name="tr" type="tr.type" />
= <xs:attributeGroup name="td.attlist">
  <xs:attribute name="nowrap" fixed="nowrap" />
  <xs:attribute name="bgcolor" type="Color" />
  <xs:attribute name="text" type="Color" />
  <xs:attribute name="rowspan" type="Length" default="1" />
  <xs:attribute name="colspan" type="Length" default="1" />
  <xs:attributeGroup ref="CellHAlign.attrib" />
  <xs:attributeGroup ref="CellVAlign.attrib" />
</xs:attributeGroup>
= <xs:complexType name="td.type" mixed="true">
  <xs:group ref="Block.mix" />
  <xs:attribute name="height" type="restrictedLength" />
  <xs:attribute name="width" type="restrictedLength" />
  <xs:attributeGroup ref="td.attlist" />
</xs:complexType>
<xs:element name="td" type="td.type" />
```

```
= <xs:attributeGroup name="th.attlist">
  <xs:attribute name="nowrap" fixed="nowrap" />
  <xs:attribute name="bgcolor" type="Color" />
  <xs:attribute name="width" type="restrictedLength" />
  <xs:attribute name="height" type="restrictedLength" />
  <xs:attribute name="rowspan" type="Length" default="1" />
  <xs:attribute name="colspan" type="Length" default="1" />
  <xs:attributeGroup ref="CellHAlign.attrib" />
  <xs:attributeGroup ref="CellVAlign.attrib" />
</xs:attributeGroup>
= <xs:complexType name="th.type" mixed="true">
  <xs:group ref="Block.mix" />
  <xs:attributeGroup ref="th.attlist" />
</xs:complexType>
<xs:element name="th" type="th.type" />
<xs:element name="br" type="empty.type" />
<xs:element name="b" type="xs:string" />
<xs:element name="big" type="xs:string" />
<xs:element name="i" type="xs:string" />
<xs:element name="sub" type="xs:string" />
<xs:element name="sup" type="xs:string" />
<xs:element name="tt" type="xs:string" />
<xs:element name="hr" type="empty.type" />
= <xs:attributeGroup name="font.attlist">
  <xs:attribute name="size" type="FontSize" />
  <xs:attribute name="color" type="Color" />
  <xs:attribute name="face" type="FontFace" />
</xs:attributeGroup>
= <xs:complexType name="font.type" mixed="true">
  <xs:group ref="Text.class" minOccurs="0" maxOccurs="unbounded" />
  <xs:attributeGroup ref="font.attlist" />
</xs:complexType>
<xs:element name="font" type="font.type" />
</xs:schema>
```

Folgende Farben können wahlweise als Text- bzw. Hintergrundfarbe verwendet werden:

Black
Silver
Gray
White
Maroon
Red
Purple
Fuchsia
Green
Lime
Olive
Yellow
Navy
Blue
Teal
Aqua

Die Verwendung einer Farbe sowohl für den Text als auch für den Hintergrund wird ausgeschlossen. Zusätzlich werden noch die folgenden Farbkombinationen ausgeschlossen:

Navy	Black
Olive	Gray
Yellow	White
Purple	Maroon
Red	Fuchsia
Teal	Green
Aqua	Lime

- **hot:PDFSign**

Die Spezifikation des Dokumentenformats ist dem PDF-Standard in der Version \leq PDF 1.6 (siehe Adobe, PDF Reference 1.6) zu entnehmen.

Dynamische Elemente werden aus dem PDF-Dokument entfernt.

- **SecSigner ® 2.0.0 (SecCommerce Informationssysteme GmbH)**

Die folgenden Daten-/Dokumentformate werden von SecSigner unterstützt:

Reiner Text (plain text)

Für die Darstellung der Daten wird der Standardzeichensatz verwendet, den das Betriebssystem bereitstellt. Das ist in Europa das Character set nach ISO 8859-1.

HTML (Subset)

Es werden folgende Tags des HTML 2.0 Standards unterstützt:
A, ADDRESS, B, BASE, BIG, BLOCKQUOTE, BR, CENTER, CITE, CODE,
COMMENT, DD, DIR, DL, DT, EM, FONT, H1, H2, H3, H4, H5, H6, HEAD,
HTML, HR, I, KBD, LI, LISTING, MENU, OL, P, PRE, SAMP, SMALL,
STRONG, TABLE, TITLE, TT, TR, TD, UL, VAR, XMP.

Die HTML-Seite wird mit den Anzeigoptionen

- "Text",
- "HTML einfarbig" oder
- "HTML mehrfarbig"

dargestellt, die der Signator zur Laufzeit des Signierprozesses vor der eigentlichen Signatur selbst beliebig auswählen und wechseln kann. Die HTML-Darstellung nutzt die im HTML-Dokument enthaltenen Formatierungs- und Farbinformationen zur Darstellung, wobei der Hintergrund immer und unabhängig von der Farbinformation der HTML-Seite weiß dargestellt wird.

In der Darstellungsoption "HTML einfarbig" wird der in der HTML-Seite enthaltene Text unabhängig von der Farbinformation der HTML-Seite schwarz dargestellt.

In der Darstellungsoption "HTML mehrfarbig" wird der Text mit der Farbinformation der HTML-Seite dargestellt, mit folgender Ausnahme: unterscheidet sich die im HTML-Dokument definierte Textfarbe nicht deutlich genug von der Hintergrundfarbe "weiß", wird schwarz als Textfarbe verwendet.

Voraussetzungen für eine vertrauenswürdige Einsatzumgebung

Der Signator muss an seinen Signatarbeitsplatz besondere Bedingungen stellen, um die sichere Signaturerstellung in einer vertrauenswürdigen Einsatzumgebung zu gewährleisten:

- Wenn der Rechner mehr als einer Person zugänglich ist oder eine Internetverbindung besteht, muss regelmäßig für einen aktuellen Virenschutz gesorgt werden.
- Einrichtung einer Personal Firewall.
- Die vom Hersteller des Betriebssystems empfohlenen Sicherheitsupdates müssen installiert werden.
- Die von a.trust empfohlenen Komponenten und Verfahren müssen stets in der aktuellen Version verwendet werden (Secure Viewer, Treibersoftware für Kartenleser).