



Zentrum für sichere Informationstechnologie – Austria Secure Information Technology Center – Austria

A-1040 Wien, Weyringergasse 35
Tel.: (+43 1) 503 19 63-0
Fax: (+43 1) 503 19 63-66

A-8010 Graz, Inffeldgasse 16a
Tel.: (+43 316) 873-5514
Fax: (+43 316) 873-5520

<http://www.a-sit.at>
E-Mail: office@a-sit.at

SIGNATURPRÜFSERVICE PROJEKTDOKUMENTATION VERSION 1.7, 15.07.2013

Thomas Knall – thomas.knall@a-sit.at

Alexander Marsalek – alexander.marsalek@a-sit.at

Zusammenfassung: Mit der steigenden Zahl von E-Business- und E-Government-Anwendungen stieg gleichermaßen die Notwendigkeit einen einheitlichen Prüfdienst für signierte Dokumente zu schaffen. Dies wurde mit dem Webservice "Signaturprüfservice" verwirklicht. Der Anwender hat die Möglichkeit signierte Dokumente (basierend auf Zertifikaten) upzuloaden und prüfen zu lassen. Das Prüfergebnis erhält er als signiertes Prüfprotokoll.

Um den Anforderungen eines zentralen (bzw. einheitlichen) Prüfdienstes gerecht zu werden, beschränkt sich das Service nicht auf ein bestimmtes Dokumentformat. Durch einen speziellen Erkennungsprozess wird zunächst der Dokumenttyp ermittelt und ggf. ein entsprechendes Prüfverfahren eingeleitet. Sowohl der Erkennungsprozess als auch der Prüfvorgang sind beliebig erweiterbar, sodass zukünftige Dokumentformate problemlos berücksichtigt werden können.

Inhaltsverzeichnis

Inhaltsverzeichnis	1
Abbildungsverzeichnis	2
1 Einleitung	3
2 Kurzbeschreibung	4
2.1 Erkennung des Dokument-Formats	4
2.2 Signaturverifikation	5
2.3 Zeitliche Abfolge	5
2.4 Komponenten	6
3 Anwendungsbeschreibung	7
3.1 Interpretation des Signaturprüfergebnisses	8
3.2 Anzeige der Signaturdaten	11
3.3 Inkrementelle Updates	13
4 Deployment	15
4.1 Voraussetzungen	15
4.2 Installation der Software	16
4.3 Konfiguration	16
Referenzen	26
Historie	27

Abbildungsverzeichnis

Abb. 2.1: Prinzip der Dokumentformat-Erkennung	4
Abb. 2.2: zeitliche Abfolge eines typischen Prüfvorgangs	5
Abb. 3.1: Start des Signaturprüfservice	7
Abb. 3.2: Startseite des Signaturprüfservice	7
Abb. 3.3: Zusammenfassung des Prüfergebnisses	8
Abb. 3.4: Beispiel-Ergebnis (ungültige Signatur)	8
Abb. 3.5: Beispiel eines signierten Prüfprotokolls.....	9
Abb. 3.6: Detailansicht einer Signaturprüfung	10
Abb. 3.7: Hinweis zur Ansicht der Signaturdaten	11
Abb. 3.8: Möglichkeit der Anzeige der Signaturdaten über die Detailansicht	12
Abb. 3.9: Ansicht der Signaturdaten.....	12
Abb. 3.10: Verhalten bei Ergänzung des Dokuments nach der letzten Signatur	13
Abb. 3.11: Verhalten bei Ergänzung des Dokuments vor der letzten Signatur.....	14
Abb. 4.1: Schritt 1, Start des Installationspakets	16
Abb. 4.2: Schritt 2, Akzeptieren der Lizenzbedingungen	16
Abb. 4.3: Schritt 3, Wahl des Installationsverzeichnisses	16
Abb. 4.4: Schritt 4, Installation starten.....	16
Abb. 4.5: Schritt 5, Installation ist abgeschlossen.....	16
Abb. 4.6: Starten/Stoppen des Windows-Dienstes	16
Abb. 4.7: Eigenschaften des Windows-Dienstes	16

1 Einleitung

Elektronische Signaturen stellen die Basis von E-Business- bzw. E-Government-Anwendungen dar. Mittels Signaturen werden Daten authentifiziert, vor unbemerkter Manipulation geschützt sowie die Möglichkeit geschaffen, behördliche Schriftstücke, die zuvor ausschließlich auf Papier existierten, nun elektronisch anzubieten.

Die Verwendung von elektronischen Signaturen wirft gleichfalls die Frage nach der Validierung der Signatur solcher Dokumente auf. Besonders verschiedene Formate elektronischer Signaturen oder deren Bindung an die signierten Dokumente sind für Anwender schwer zu verstehen. Bislang sind Anwender darauf angewiesen, dass die jeweilige Anwendung Möglichkeiten zur Verifikation der erstellten, signierten Dokumente zur Verfügung stellt. Trotz Verwendung von Frameworks wie MOA ([MOA], vor allem MOA-SP) stellt dies durch wiederholte Implementierung einen unnötigen Aufwand für die Entwickler der einzelnen Anwendungen dar. Zusätzlich verkomplizieren unterschiedliche Benutzer-Frontends bzw. die nicht-einheitliche Darstellung der Verifikationsergebnisse verschiedener Anwendungen den Verifikationsprozess, erschweren die Anwendung und tragen zur Verwirrung und Verunsicherung potentieller Benutzer bei.

Mit der Existenz einer einfachen zentralen Möglichkeit zur Prüfung beliebiger elektronisch signierter Dokumente wird sich das Vertrauen der Anwender in Signatur-Anwendungen steigern, was wiederum zu einer größeren Verbreitung bzw. gesteigerten Nutzungsbereitschaft führt.

Die Vorteile einer einheitlichen Prüfmöglichkeit lassen sich in einigen Punkten zusammenfassen:

- *einheitliches Layout*: Das Web-Frontend des Prüfdienstes bietet stets ein identisches Layout, unabhängig davon in welchem Kontext die Prüfung von Dokumenten stattfindet. Dies erleichtert – durch einen hohen Wiedererkennungsfaktor – die Anwendung der Prüffunktionen für den Benutzer.
- *redundante Nutzung verschiedener Basisdienste*: Durch die Zusammenfassung unterschiedlichster Prüfmethoden an zentraler Stelle besteht die Möglichkeit auf bereits vorhandene Basisdienste – wie z.B. MOA-SS/SP zur Prüfung von XML/CMS-Signaturen – zurückzugreifen, was in Summe den Implementierungsaufwand weiter reduziert.
- *Erweiterbarkeit*: Um zukünftige Erweiterbarkeit zu sichern, kann ein zentraler Prüfdienst offene Schnittstellen definieren, auf die zur Erweiterung der unterstützten Dokument- bzw. Signaturformate zurückgegriffen werden kann.
- *Wartung*: Für den Betreiber eines zentralen Prüfdienstes gestaltet sich die Wartung in Summe einfacher als die Wartung einzelner Prüfdienste. Ein Update des zu Grunde liegenden Frameworks MOA beispielsweise ist nur mehr an einer einzelnen Stelle erforderlich.
- *Bekanntheitsgrad*: Die Errichtung eines zentralen Prüfdienstes führt automatisch zu einer fortwährenden Steigerung des Bekanntheitsgrades. Dies umfasst sowohl die URL des Benutzer-Frontends wie auch das stets einheitliche und damit bereits "bekannte" Layout.
- *Vertrauen*: Ein steigender Bekanntheitsgrad und die damit einhergehende Zunahme der Nutzung eines solchen Dienstes führt zu einer Steigerung des Vertrauens durch die Anwender.
- *Nutzungsbereitschaft*: Ein steigender Bekanntheitsgrad und ein starkes Vertrauen in den Prüfdienst führt zu einer erhöhten Nutzungsbereitschaft.

2 Kurzbeschreibung

Das Signaturprüfservice gliedert sich in zwei Teilbereiche:

- Erkennung des vorliegenden Dokumenttyps
- Signaturverifikation

2.1 Erkennung des Dokument-Formats

Der Erkennungsprozess erfolgt auf hierarchische Weise (siehe Abb. 2.1). Jedes Element des dargestellten Baumes entspricht einem bestimmten Dokumenttyp für den das Element spezielle Erkennungsroutinen besitzt. Der Erkennungsvorgang geschieht iterativ ausgehend von der Wurzel ("Unbekanntes Format"). Jedes angesprochene Element liefert einen Status ("akzeptiert/nicht akzeptiert") über das jeweilig untersuchte Dokument zurück. "Akzeptiert" ein Element das Dokument werden dessen Kind-Elemente zur weiteren Untersuchung herangezogen usw. Durch den iterativen Prozess ergibt sich ein Pfad durch den Baum (über akzeptierende Elemente) von der Wurzel bis zu jenem Element, das als Ergebnis des Erkennungsvorgangs betrachtet werden kann.

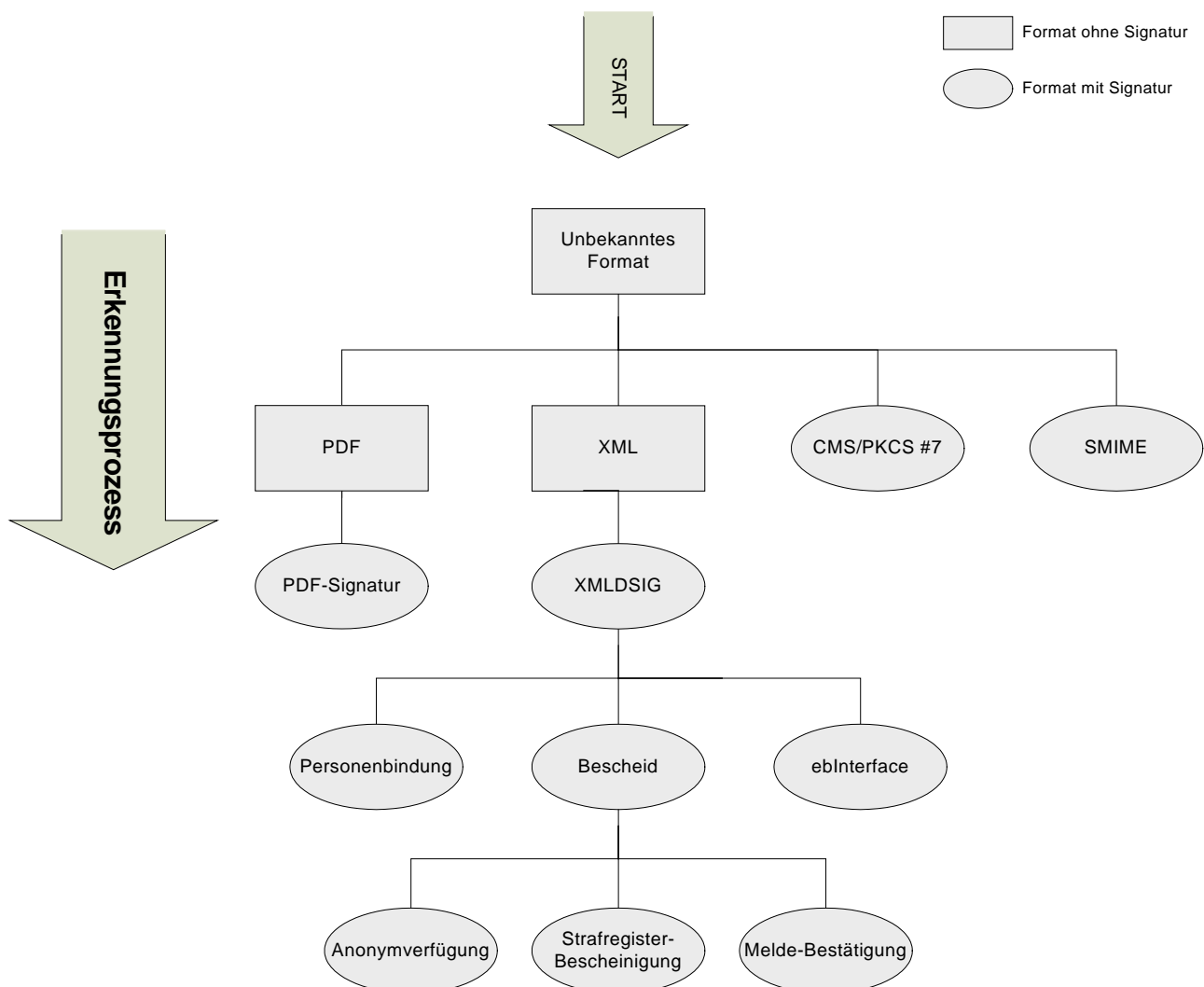


Abb. 2.1: Prinzip der Dokumentformat-Erkennung

Durch die hier dargestellte Architektur sind Erweiterungen zur Unterstützung neuer (zukünftiger) Dateiformate leicht realisierbar. Die Registrierung der einzelnen Elemente findet über eine XML-Konfigurationsdatei statt wobei sich die Hierarchie durch entsprechende Verschachtelung der konfigurierten Elemente ergibt.

2.2 Signaturverifikation

Wurde ein überprüfbarer Dokumenttyp gefunden müssen zunächst signaturrelevante Informationen (z.B. Zeitpunkt der Signatur) extrahiert und dem Modul für Online Applikationen ([MOA]) zur Signaturprüfung übergeben werden. Dabei spielt die Anzahl der enthaltenen Signaturen keine Rolle. Im Falle von Mehrfachsignaturen wird MOA entsprechend oft aufgerufen.

Um nun die passende Signaturextraktion durchführen zu können existiert eine – der Dokumenterkennung ähnliche – Hierarchie an Verifikatoren, die über eine weitere Konfigurationsdatei mit registrierten Dokumentformaten verknüpft werden.

Ergebnis der Signaturprüfung ist ein signiertes Prüfprotokoll im PDF-Format, das sowohl Informationen zum Dokument (Dateiname, Hash, Dokumenttyp) als auch Informationen zum Zertifikat und das Resultat der Signaturprüfung enthält.

2.3 Zeitliche Abfolge

Die zeitliche Abfolge bzw. das Zusammenspiel von Webservice, Dokumentformat-Erkennung und Signaturverifikation ist aus der folgenden Abbildung (Abb. 2.2) ersichtlich.

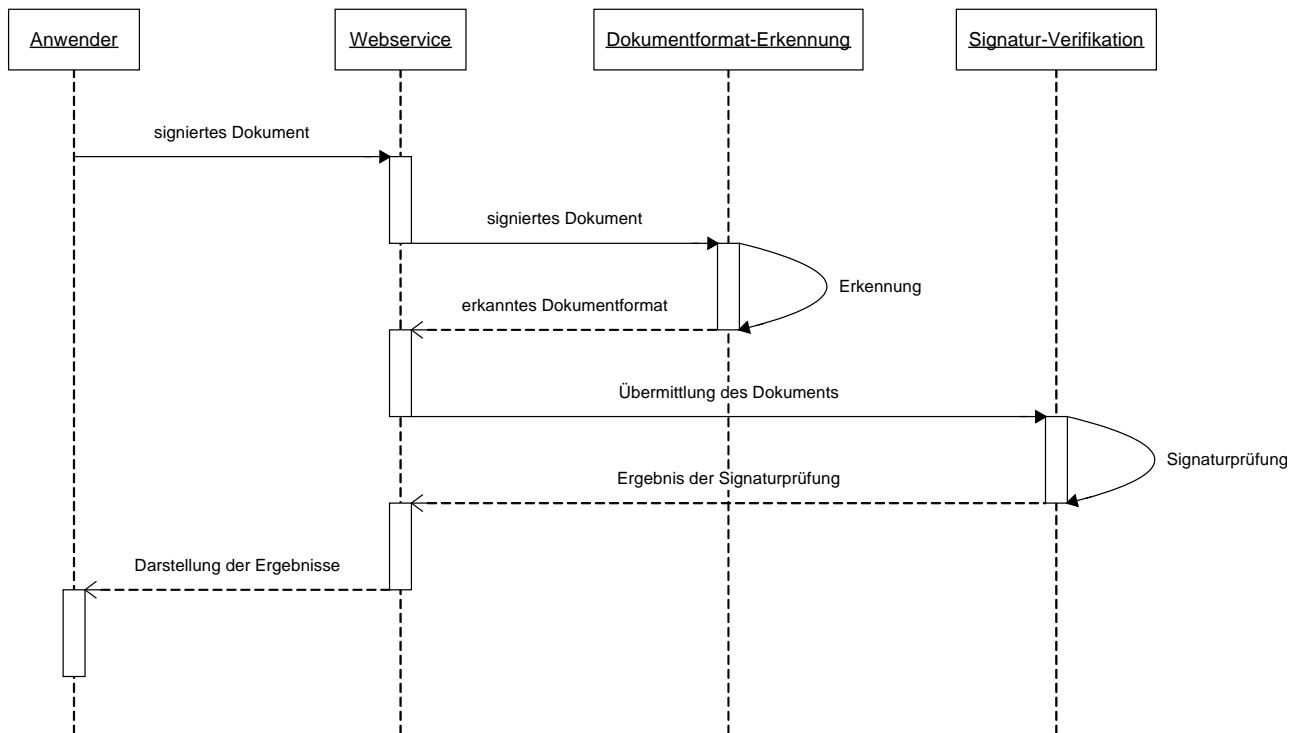


Abb. 2.2: zeitliche Abfolge eines typischen Prüfvorgangs

2.4 Komponenten

Sämtliche in diesem Abschnitt angeführten Komponenten sind als Voraussetzung zur Nutzung dieser Applikation zu betrachten. Eine hundertprozentige Funktionalität der Anwendung kann bei einer abweichenden Konfiguration nicht gewährleistet werden.

Das Signaturprüfservice ist eine Web-Anwendung bei der folgende Komponenten zum Einsatz kommen.

- Java 2 Plattform v1.6.0¹
- Apache Tomcat v6.0.37²
- Apache Struts v1.3.10³

Die nachfolgenden Projekte bzw. Infrastrukturelemente finden im Signaturprüfservice Verwendung:

2.4.1 Modul für Online Applikationen – Serversignatur/Signaturprüfung

Zur Signaturverifikation wird eine speziell angepasste Version des Moduls MOA-SPSS⁴ (Modul für Online Applikationen – Serversignatur/Signaturprüfung) auf Basis der Version 1.5.1 genutzt. Die entsprechende MOA-Distribution ist im Installations-Paket (siehe Abschnitt 4.2) dieser Anwendung inkludiert.

Grundsätzlich können ältere Versionen bis 1.3.x von MOA-SPSS verwendet werden. Dies schließt jedoch einige Dokumenttypen (z.B. EHVD-Token) bzw. einige Signatur-Varianten von der Prüfung aus.

2.4.2 PDF-Amtssignatur

Die Signatur des Prüfprotokolls findet mittels PDF-Textsignatur⁵ bzw. MOA-SS statt. Die PDF-Textsignatur wurde entwickelt, um PDF-Dokumente mit einer elektronischen Signatur versehen zu können, die bei Bedarf vom Papierausdruck rekonstruiert und validiert werden kann.

2.4.3 IAIK-Crypto Toolkits

Beim Signaturprüfservice kommen einige IAIK-Toolkits⁶ wie IAIK-Java Cryptography Extension, IAIK ECC, IAIK CMS-S/MIME oder XSECT zum Einsatz. Diese werden unter anderem zur Untersuchung von Zertifikaten, zur Hashberechnung, für einen Zertifikatdownload über LDAP oder für die Verifikation von CMS-Dokumenten verwendet.

Diese Komponenten sind im kommerziellen Umfeld kostenpflichtig, für Forschung und Ausbildung sind kostenlose Lizenzen⁷ verfügbar.

¹ <http://www.oracle.com/technetwork/java/javase/downloads/index.html>

² <http://tomcat.apache.org/download-60.cgi>

³ <http://struts.apache.org/release/1.3.x/index.html>

⁴ <https://joinup.ec.europa.eu/software/moa-idspss/release/all>

⁵ https://demo.egiz.gv.at/plain/projekte/signatur_im_e_government/pdf_signatur

⁶ <http://jce.iaik.tugraz.at/sic/products>

⁷ <http://jce.iaik.tugraz.at/sic/sales/licences>

3 Anwendungsbeschreibung

Nach erfolgter Installation (siehe Abschnitt 4.2) kann die Anwendung durch ausführen des Tomcat Servers gestartet werden. Nun kann das Frontend der Anwendung durch Aufruf der URL <http://localhost:30080/signature-verification>

aufgerufen werden. Wurde der Tomcat Container für einen anderen Port als 30080 konfiguriert oder befindet sich das Signaturprüfservice nicht lokal am PC des Anwenders dann ist der Link entsprechend anzupassen.

Beim Start der Anwendung präsentiert sich dem Benutzer folgende Oberfläche.

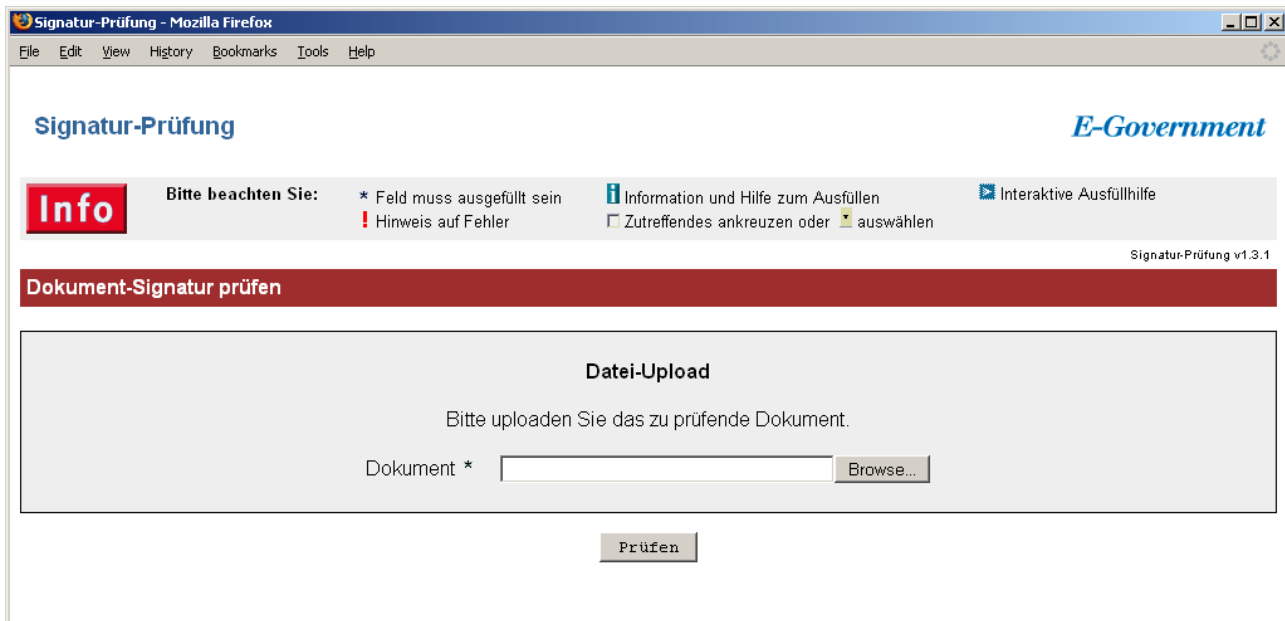


Abb. 3.1: Startseite des Signaturprüfservice

Nach Auswahl eines signierten Dokuments über den "Browse"-Button kann die Prüfung mit Klick auf "Prüfen" ausgelöst werden.

Nach erfolgter Prüfung wird dem Anwender eine Seite mit einer Zusammenfassung der Prüfergebnisse präsentiert (Abb. 3.3). Diese gliedert sich in zwei Teile:

Informationen zum Dokument: Diese umfassen den Namen des untersuchten Dokuments, dessen Größe sowie den erkannten Dokumenttyp. Zur leichteren späteren Identifizierung wird zusätzlich ein Hash-Wert (Base64-encodierter SHA1-Hash) über das Dokument berechnet.

Informationen zum Prüfergebnis: Der Abschnitt "Signaturen" umfasst alle im Dokument enthaltenen und geprüften Signaturen. Im Fall des in Abb. 3.3 gezeigten Ergebnisses handelt es sich beispielsweise um eine einzelne Signatur.

Signatur-Prüfung - Mozilla Firefox

File Edit View History Bookmarks Tools Help

Signatur-Prüfung *E-Government*

Info **Bitte beachten Sie:** * Feld muss ausgefüllt sein i Information und Hilfe zum Ausfüllen 🔗 Interaktive Ausfüllhilfe
 ! Hinweis auf Fehler ☐ Zutreffendes ankreuzen oder 📄 auswählen

Signatur-Prüfung v1.3.1

Dokument-Signatur prüfen

Prüfprotokoll

Nachfolgend finden Sie Informationen über das eingereichte Dokument.

Dokument	
Dateiname	meldebestaetigung_01.xml
Hash-Wert	m2x5Ps+ZJotqhLRWbMCWV75fkF4=
Größe	7,27 kB
Typ	Meldebestätigung

Nachfolgend finden Sie einen Überblick über die geprüften Signaturen des eingereichten Dokuments.
 Details zu einer Signatur können durch Klick auf den Namen des Unterzeichners eingesehen werden.

Signaturen			
Unterzeichner	S	Z	M
Oswald Kessler	OK	OK	OK

Eine Signatur ist dann als "gültig" zu betrachten wenn jede der Prüfungen Signaturwert (S), Zertifikat (Z) sowie Manifest (M) mit "OK" abgeschlossen wurde.

[Prüfprotokoll downloaden](#)

Weitere Dokumente prüfen

Abb. 3.2: Zusammenfassung des Prüfergebnisses

3.1 Interpretation des Signaturprüfergebnisses

Zunächst präsentiert sich dem Anwender eine Übersicht über alle geprüften Signaturen (Abb. 3.3). Die linke Spalte zeigt den Namen des Signators, die rechten drei Spalten zeigen das Ergebnis der **S**ignatur-, **Z**ertifikats- bzw. **M**anifest-Verifikation der jeweiligen Signatur. Eine Signatur gilt genau dann als "gültig" wenn jede einzelne der Prüfungen ein "OK" mit grünem Hintergrund aufweist.

Analog dazu gilt ein Dokument dann als "gültig" wenn jede einzelne Signatur als "gültig" betrachtet werden kann. Abb. 3.4 zeigt ein Signaturprüfergebnis das zwar ein gültiges Zertifikat sowie ein gültiges Manifest bescheinigt, die Signatur jedoch als "nicht gültig" anzeigt. Grund dafür ist eine nachträgliche (eventuell böswillige) Veränderung des Dokuments. Das Dokument kann demnach nicht als "gültig" bestätigt werden.

S	Z	M
X	OK	OK

Abb. 3.3: Beispiel-Ergebnis (ungültige Signatur)

Ein Klick auf "Prüfprotokoll downloaden" liefert ein Prüfprotokoll (Abb. 3.5) das selbst mit einer PDF-Textsignatur versehen ist. Dies Protokoll enthält alle aus der Prüfung hervorgehenden Informationen.

Prüfprotokoll

Dokument

Dateiname meldebestaetigung_01.xml
 Hash-Wert m2x5Ps+ZJotqhLRWbMCWV75fkF4= (SHA-1, Base64-kodiert)
 Größe 7,27 kB
 Typ Meldebestätigung

Signatur

Prüfungen

Signatur- bzw. Prüfzeitpunkt (UTC) 2005-09-04T11:30:38Z

Signatur Die Überprüfung der Hash-Werte und des Werts der Signatur konnte erfolgreich durchgeführt werden.

Manifest Für diese Signatur ist kein Signaturmanifest notwendig.

Zertifikat Eine formal korrekte Zertifikatskette vom Signatorzertifikat zu einem vertrauenswürdigen Wurzelzertifikat konnte konstruiert werden. Jedes Zertifikat dieser Kette ist zum in der Anfrage angegebenen Prüfzeitpunkt gültig.

Unterzeichner

Name Oswald Kessler
 Organisationseinheit Sektion IV, SU-ZMR
 Organisation Bundesministerium fuer Inneres
 Staat AT
 Seriennummer dez.: 202828344996, hex.: 2F3982EEA4

Aussteller

Name a-sign-corporate-light-02
 Organisationseinheit a-sign-corporate-light-02
 Organisation A-Trust Ges. f. Sicherheitssysteme im elektr. Datenverkehr GmbH
 Staat AT

Zertifikat

Seriennummer dez.: 80524, hex.: 013A8C
 Qualität Gewöhnliches Zertifikat
 zeitliche Gültigkeit gültig von 2005-05-25T08:10:54Z bis 2010-05-25T08:10:54Z
 Der Prüfzeitpunkt liegt innerhalb des Gültigkeitszeitraumes.

Eigenschaften Verwaltungseigenschaft
 Verwendungszweck digitalSignature, keyEncipherment, dataEncipherment
 Zertifizierungsstatement <http://www.a-trust.at/docs/cp/a-sign-Amtssignatur>

Signaturwert	EobKG/L1AMcVGN9ji6MhYFy73VLqrAI3+gFRS2mHF/oTAcjb7vR4dgEvBqdBTYh3bwÜsP1/AKBYK6ZphhISpJ4Gvs2L05eLCRIFBRGtR6fdVtgTY2FIk65Sbzf1jFy+aQabTQtByC+ThTSBRSmISP9q0QzqN9UURfj7W5cACZC=	
	Unterzeichner	Signatur-Prüfung v1.3.1
	Datum/Zeit-UTC	2008-03-06T08:30:22+01:00
	Aussteller-Zertifikat	CN=SignaturPruefung-TestSignatur,OU=Institute for Applied Information Processing and Communications,O=Graz University of Technology,STREET=Inffeldgasse 16a,L=Graz,ST=Styria,C=AT
	Serien-Nr.	1168252097279
	Methode	urn:pdfsigfilter:bka.gv.at:text:v1.1.0
Prüfhinweis	Prüfservice: https://demo.a-sit.at/el_signatur/verification	

Abb. 3.4: Beispiel eines signierten Prüfprotokolls

Details zu jeder einzelnen Signatur (siehe Abb. 3.6) können durch Klick auf den jeweiligen Signator-Namen in der Zusammenfassung des Prüfergebnisses aufgerufen werden.

Prüfungen	
Signatur- bzw. Prüfzeitpunkt (UTC)	2005-09-04T11:30:38Z
Signatur	Die Überprüfung der Hash-Werte und des Werts der Signatur konnte erfolgreich durchgeführt werden.
Zertifikat	Eine formal korrekte Zertifikatskette vom Signatorzertifikat zu einem vertrauenswürdigen Wurzelzertifikat konnte konstruiert werden. Jedes Zertifikat dieser Kette ist zum in der Anfrage angegebenen Prüfzeitpunkt gültig.
Manifest	Für diese Signatur ist kein Signaturmanifest notwendig.

Unterzeichner	
Name	Oswald Kessler
Organisationseinheit	Sektion IV, SU-ZMR
Organisation	Bundesministerium fuer Inneres
Staat	AT
Seriennummer	dez.: 202828344996, hex.: 2F3982EEA4

Aussteller	
Name	a-sign-corporate-light-02
Organisationseinheit	a-sign-corporate-light-02
Organisation	A-Trust Ges. f. Sicherheitssysteme im elektr. Datenverkehr GmbH
Staat	AT

Informationen zum Zertifikat	
Seriennummer	dez.: 80524, hex.: 013A8C
Qualität	Gewöhnliches Zertifikat
zeitliche Gültigkeit	gültig von 2005-05-25T08:10:54Z bis 2010-05-25T08:10:54Z Der Prüfzeitpunkt liegt innerhalb des Gültigkeitszeitraumes.
Eigenschaften	Verwaltungseigenschaft
Verwendungszweck	digitalSignature, keyEncipherment, dataEncipherment
Zertifizierungsstatement	http://www.a-trust.at/docs/cp/a-sign-Amtssignatur
Zertifikat downloaden	

Abb. 3.5: Detailansicht einer Signaturprüfung

Die Kategorie "Prüfungen" der Detailansicht enthält neben dem Zeitpunkt der Signatur zusätzlich eine nähere Erläuterung der bereits auf der Übersichtsseite präsentierten Prüfergebnisse für Signatur, Zertifikat und Manifest. Hier gilt grundsätzlich wiederum, dass ein grün hinterlegtes Ergebnis als positiv (d.h. gültig) betrachtet werden kann.

Die folgenden beiden Kategorien "Unterzeichner" bzw. "Aussteller" zeigen Informationen zum Unterzeichner des Dokuments bzw. Aussteller des Zertifikats.

Der letzte Abschnitt liefert Informationen zum Zertifikat. Neben der Seriennummer wird zusätzlich die Qualität des Zertifikats angegeben. Unterschieden wird zwischen "gewöhnlichem" Zertifikat und "qualifiziertem"⁸ Zertifikat. Weiters ist der Gültigkeitszeitraum des Zertifikats ersichtlich. Dies kann beispielsweise nützlich sein um ein negatives Zertifikatsprüfergebnis zu interpretieren.


⁸ <http://www.signatur.rtr.at/de/security/faq210.html>

Die für die öffentliche Verwaltung relevanten Eigenschaften⁹ eines Zertifikats werden anhand ihrer Object Identifier ([OID]) aufgelöst und dargestellt (siehe auch Abschnitt 4.3.3.8).

Schließlich werden noch der Verwendungszweck des Zertifikats und eventuell enthaltene Zertifizierungsstatements präsentiert.

Das Zertifikat selbst kann durch Klick auf "Zertifikat downloaden" heruntergeladen werden.

3.2 Anzeige der Signaturdaten

Je nach Dokumentformat besteht die Möglichkeit, die der Signatur zugrunde liegenden Daten zu betrachten. Die Möglichkeit wird auf der Übersichtsseite durch das Symbol  neben dem Namen des Signators angezeigt (siehe Abb. 3.7).

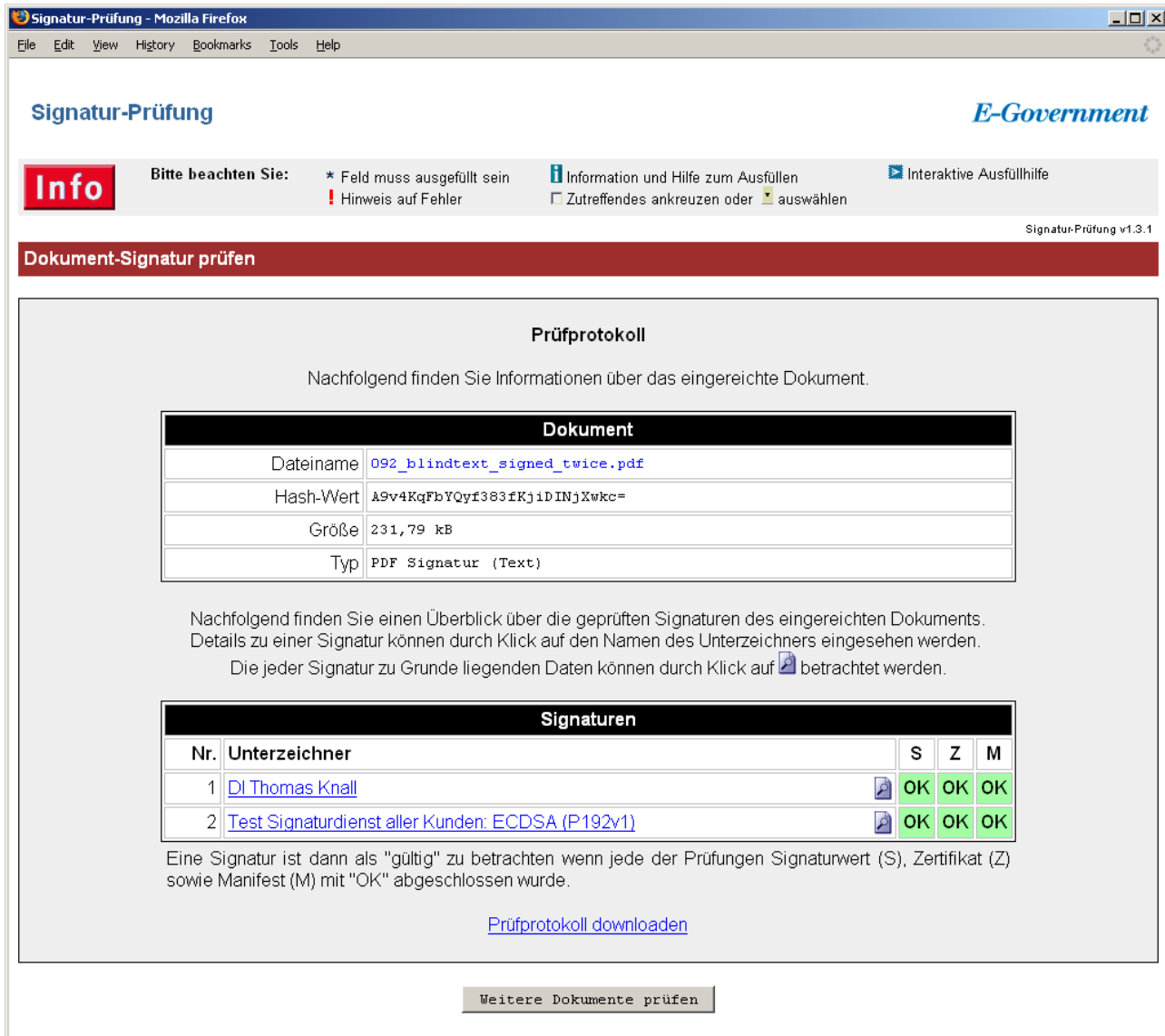


Abb. 3.6: Hinweis zur Ansicht der Signaturdaten


⁹ z.B. Verwaltungseigenschaft, Dienstleistungseigenschaft, Notareigenschaft, Rechtsanwaltseigenschaft, Ziviltechnikereigenschaft, Eigenschaft zur Ausstellung von Personenbindungen, Eigenschaft zur Eintragung von elektronischen Vollmachten oder Organwaltereigenschaft

Alternativ dazu kann in der Detail-Ansicht jeder Signatur der Link "signierte Daten anzeigen" (siehe Abb. 3.8) angeklickt werden.

Prüfungen	
Signatur- bzw. Prüfzeitpunkt (UTC)	2008-03-06T07:44:57Z
Signatur	Die Überprüfung der Hash-Werte und des Werts der Signatur konnte erfolgreich durchgeführt werden.
Zertifikat	Eine formal korrekte Zertifikatskette vom Signatorzertifikat zu einem vertrauenswürdigen Wurzelzertifikat konnte konstruiert werden. Jedes Zertifikat dieser Kette ist zum in der Anfrage angegebenen Prüfzeitpunkt gültig.
Manifest	Für diese Signatur ist kein Signaturmanifest notwendig.

Zusatzinformationen	
Signaturtyp	Text
signierte Daten anzeigen	

Abb. 3.7: Möglichkeit der Anzeige der Signaturdaten über die Detailansicht

Sowohl der Klick auf das Symbol  als auch auf den Link "signierte Daten anzeigen" führt zur Anzeige der Daten, die der jeweiligen Signatur zugrunde liegen. Dabei kann es sich um ein reines Text-Format (siehe Abb. 3.9) oder aber auch um ein anderes Format (z.B. PDF, falls eine Binärsignatur vorliegt) handeln.

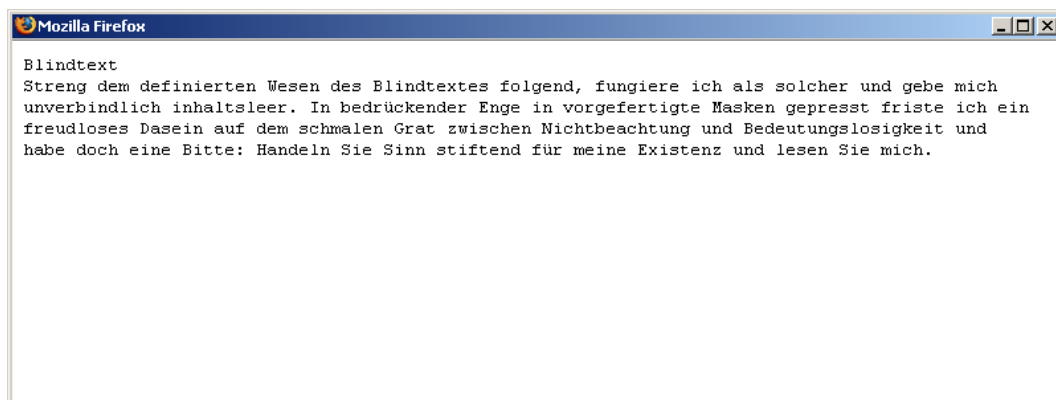


Abb. 3.8: Ansicht der Signaturdaten

3.3 Inkrementelle Updates

Dieser Abschnitt betrifft ausschließlich die Prüfung des Dokumenttyps "Portable Document Format" (PDF).

Das PDF-Format bietet die Möglichkeit, Änderungen am Dateiinhalt als sogenannte "Inkrementelle Updates" am Ende der Datei anzuhängen. Dabei bleibt der bisherige (signierte) Dokumentinhalt vollständig erhalten. Mögliche Anwendungsfälle sind Verfahren, bei denen nach einer Signatur Ergänzungen wie z.B. Genehmigungsvermerke angebracht werden.

Neu hinzugefügte Inhalte befinden sich allerdings in unsignierten Bereichen. Um eine missbräuchliche Verwendung zu verhindern, kann PDF-AS so konfiguriert werden (siehe Abschnitt 4.3.5), dass es Dokumente mit inkrementellen Ergänzungen gesondert behandelt.

Hierbei müssen zwei Fälle unterschieden werden:

Fall #1: Ein Dokument wurde zunächst signiert und dann dessen Inhalt mittels eines inkrementellen Updates ergänzt.

Fall #2: Ein Dokument wurde signiert, dann dessen Inhalt mittels eines inkrementellen Updates ergänzt und schließlich nochmals signiert.

Das Dokument aus dem ersten Fall wird als ungültig betrachtet, weil ein inkrementelles Update nach der letzten Signatur angebracht wurde. Dies führt zu der in Abb. 3.10 gezeigten Meldung bei der Signaturprüfung.

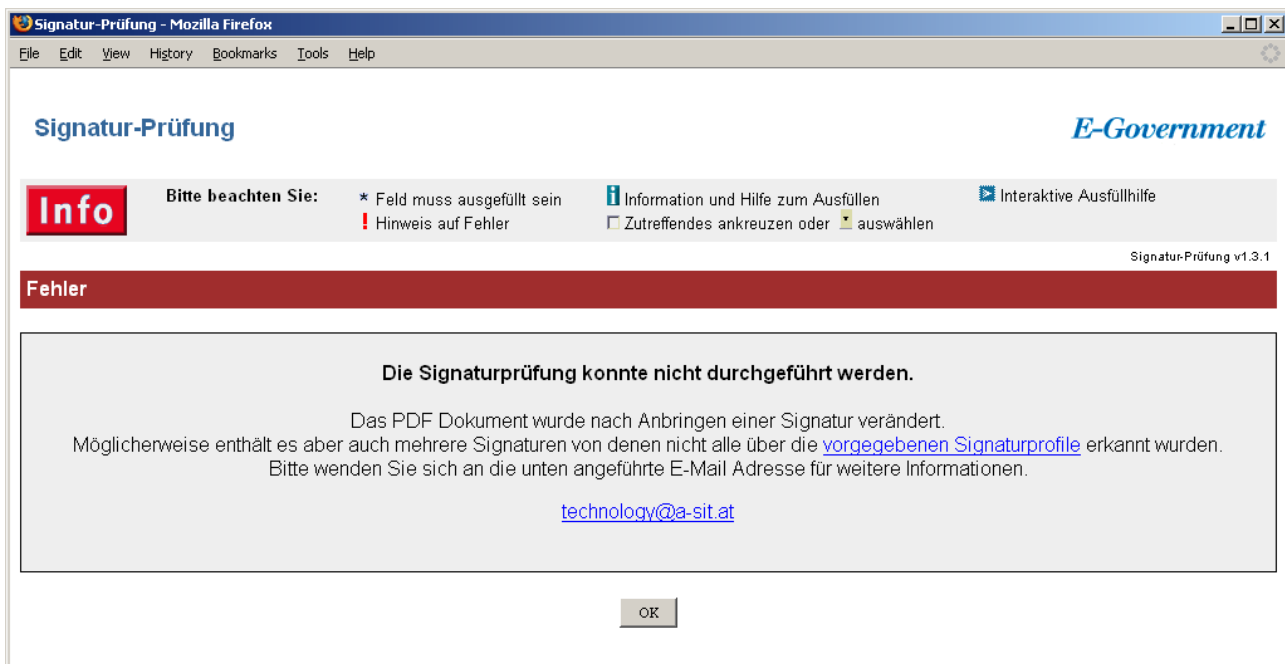


Abb. 3.9: Verhalten bei Ergänzung des Dokuments nach der letzten Signatur

Das Dokument aus dem zweiten Fall enthält zwar ein inkrementelles Update, dieses wurde jedoch durch eine nachträgliche Signatur mitsigniert. Dadurch sind die Änderungen zumindest durch die letzte Signatur abgedeckt. Das Ergebnis einer Signaturprüfung dieses Dokuments ist aus Abb. 3.11 ersichtlich. Sämtliche durch die inkrementellen Änderungen betroffene Signaturen (d.h. sämtliche Signaturen, die die inkrementellen Updates nicht abdecken) sind gesondert durch einen Stern " * " markiert. Es wird empfohlen, die Signaturdaten für diese markierten Signaturen zusätzlich zur Prüfung einzusehen (siehe auch Abschnitt 3.2).

Signatur-Prüfung *E-Government*

Info **Bitte beachten Sie:** * Feld muss ausgefüllt sein ! Hinweis auf Fehler i Information und Hilfe zum Ausfüllen 🔗 Interaktive Ausfüllhilfe
 Zutreffendes ankreuzen oder auswählen

Signatur-Prüfung v1.3.1

Dokument-Signatur prüfen

Prüfprotokoll

Nachfolgend finden Sie Informationen über das eingereichte Dokument.

Dokument	
Dateiname	blindtext_BSIG_MOD_TSIG.pdf
Hash-Wert	w8fxqYSEmvtN/XyerkYQjA1lr0A=
Größe	247,81 kB
Typ	PDF Signatur (Text + Binär)

Nachfolgend finden Sie einen Überblick über die geprüften Signaturen des eingereichten Dokuments. Details zu einer Signatur können durch Klick auf den Namen des Unterzeichners eingesehen werden. Die jeder Signatur zu Grunde liegenden Daten können durch Klick auf betrachtet werden.

Signaturen				
Nr.	Unterzeichner	S	Z	M
1	DI Thomas Knall *	OK	OK	OK
2	DI Thomas Knall	OK	OK	OK

Eine Signatur ist dann als "gültig" zu betrachten wenn jede der Prüfungen Signaturwert (S), Zertifikat (Z) sowie Manifest (M) mit "OK" abgeschlossen wurde.

* Das Dokument wurde nach Aufbringung dieser Signatur ergänzt. Die Daten, die dieser Signatur zu Grunde liegen, sollten gesondert betrachtet werden.

[Prüfprotokoll downloaden](#)

Abb. 3.10: Verhalten bei Ergänzung des Dokuments vor der letzten Signatur

4 Deployment

Diese Kapitel beschreibt die Server-seitigen Voraussetzungen sowie die Installation und Einrichtung des Signaturprüfportals. Clientseitig gibt es abgesehen von einem Web-Browser keine speziellen Voraussetzungen zur Nutzung der Anwendung.

4.1 Voraussetzungen

Zur effektiven Bereitstellung der Anwendung (speziell für den Bereich PDF-Dokumente) empfiehlt sich ein Rechner

- mit mindestens 1 GB Hauptspeicher,
- mindestens 150 MB freien Festplattenspeicher
- sowie einem Prozessor mit mindestens 2 GHz Taktfrequenz.

Außerdem werden für den Betrieb des Signaturprüfdienstes eine Java-Runtime-Environment sowie ein Apache-Tomcat-Servlet Container inkl. MOA-SPSS und dem Signaturprüfservice benötigt.

Die Java-Installation muss folgende Punkte erfüllen:

- Es muss mindestens ein JDK 1.5.x¹¹ oder neuer verwendet werden.
- Das JDK muss mit den "Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files" gepatcht worden sein (Download siehe Fußnote #12).
- Das Verzeichnis `jre/lib/ext` des JDKs muss zusätzlich folgende Dateien beinhalten:
 - `activation-1.1.jar`
 - `iaik_cms-4.01_MOA.jar`
 - `iaik_ecc.jar`
 - `iaik_jce_full-4.0_MOA.jar`
 - `iaik_Pkcs11Provider.jar`
 - `iaik_Pkcs11Wrapper.jar`
 - `mail-1.4.jar`
 - `xalan.jar`
 - `xercesImpl.jar`

Es werden Apache Tomcat Versionen ab v5.5.25 unterstützt¹³. Dieses Dokument beschreibt die Installation des Apache Tomcat Servers anhand der Version 6.0.37¹⁴ für Windows.

Nach der Installation der Software hat der Anwender grundsätzlich die Möglichkeit die Anwendung über ein DOS-Fenster laufen zu lassen oder diese als Systemdienst einzurichten¹⁵.

¹¹ http://java.sun.com/javase/downloads/index_jdk5.jsp

¹² <http://www.oracle.com/technetwork/java/javase/downloads/jce-6-download-429243.html>

¹³ <http://tomcat.apache.org/>

¹⁴ <http://www.eu.apache.org/dist/tomcat/tomcat-6/v6.0.37/bin/apache-tomcat-6.0.37-windows-x86.zip>

¹⁵ <http://tomcat.apache.org/tomcat-6.0-doc/windows-service-howto.html>

4.2 Installation der Software

Das Installations-Paket besteht aus 3 Dateien, der Beispielkonfiguration, MOA-SPSS und dem Signaturprüfdienst. Die beiden Web Archive „moa-spss.war“ und „signature-verification.war“ müssen in den Ordner „webapps“ im Tomcat Installationsverzeichnis kopiert werden. Die Beispielkonfiguration muss in den Ordner „config“ im Tomcat Installationsverzeichnis entpackt werden. Abschnitt 4.3 beschreibt die Konfigurationsmöglichkeiten.

4.3 Konfiguration

Grundsätzlich kann jede zum Einsatz kommende Komponente gesondert konfiguriert werden. Diese Einzelkonfigurationen werden in den folgenden Abschnitten behandelt.

Hinweis: Informationen zur Anpassung des Signaturblocks im Prüfprotokoll (z.B. Wahl einer eigenen Bildmarke) sind in Abschnitt 4.3.5 zu finden. Wie das Signaturzertifikat für das Prüfprotokoll angepasst werden kann beschreibt Abschnitt 4.3.6.

4.3.1 Mehrsprachigkeit

Die beiden XML-Konfigurationsdateien für die Dokumenterkennung (Abschnitt 4.3.2) sowie für die Signaturprüfung (Abschnitt 4.3.3) können Platzhalter-Elemente der Form `#{bezeichnung}` enthalten. Diese Platzhalter werden zur Laufzeit durch Texte aus dem Sprach-ResourceBundle ersetzt. So wird der Platzhalter `#{error.mail.subject}` zur Laufzeit abhängig von der Spracheinstellung des jeweiligen Browsers entweder durch "error notification" oder durch "Fehlerbenachrichtigung" ersetzt.

Gleichermaßen wird das zur Signatur von Prüfprotokollen verwendete PDF-AS-Profil bestimmt.

Das Sprach-ResourceBundle besteht aus einer Sprachdatei für Deutsch (`applicationResources_de.properties`) und einer Sprachdatei für Englisch (`applicationResources.properties`). Diese Dateien befinden sich im Ordner `%TOMCAT_DIR%\webapps\signature-verification\WEB-INF\classes\languages`.

4.3.2 Dokumenttyp-Erkennung

Die Konfigurationsdatei für die Dokumenttyp-Erkennung ist unter

```
%TOMCAT_DIR%\conf\signature-verification\formatdetection-config.xml
```

zu finden. Sollte ein davon abweichender Konfigurations-Pfad erwünscht sein, kann dieser über das System-Property "formatdetection.configuration" definiert werden.

Grundsätzlich sollte es nicht notwendig sein, diese Konfiguration zu ändern. Sollte jedoch die Umbenennung eines Dokumentformats oder das Hinzufügen neuer Formate gewünscht werden, ist die Modifikation dieser Datei erforderlich.

Wie bereits in Abschnitt 2.1 erläutert erfolgt die Formaterkennung hierarchisch. Dementsprechend hierarchisch ist auch die entsprechende Konfigurationsdatei¹⁶ aufgebaut:

```
<cfg:FormatDetectionConfiguration
  xmlns:cfg="http://reference.e-government.gv.at/namespace/formatdetectionconfig/20060914#">
  <cfg:FormatDetector id="unknown">
    <cfg:Class>at.asit.formatdetection.impl.UnknownFormatDetector</cfg:Class>
    <cfg:FullName>unbekanntes Dokumentformat</cfg:FullName>
    <cfg:ShortName>unbekannt</cfg:ShortName>

  <cfg:FormatDetector id="xml">
    <cfg:Class>at.asit.formatdetection.impl.XMLFormatDetector</cfg:Class>
    <cfg:FullName>XML Digital Signature</cfg:FullName>
    <cfg:ShortName>XMLDsig</cfg:ShortName>

  <cfg:FormatDetector id="urn:oasis:names:tc:SAML:1.0:assertion:identitylink">
    <cfg:Class>at.asit.formatdetection.impl.personenbindung.PersonenbindungFormatDetector</cfg:Class>
    <cfg:FullName>Personenbindung</cfg:FullName>
    <cfg:ShortName>Personenbindung</cfg:ShortName>
  </cfg:FormatDetector>

  <cfg:FormatDetector id="birthcertificate/20060814#">
    <cfg:Class>at.asit.formatdetection.impl.geburtsurkunde.GeburtsurkundeFormatDetector</cfg:Class>
    <cfg:FullName>Geburtsurkunde (Demonstrator A-SIT)</cfg:FullName>
  </cfg:FormatDetector>

  <cfg:FormatDetector id="http://reference.e-government.gv.at/namespace/mandates/20040701#">
    <cfg:Class>at.asit.formatdetection.impl.mandate.MandateFormatDetector</cfg:Class>
    <cfg:FullName>Elektronische Vollmacht</cfg:FullName>
    <cfg:ShortName>Vollmacht</cfg:ShortName>
  </cfg:FormatDetector>

</cfg:FormatDetector>

  <cfg:FormatDetector id="pdf">
    <cfg:Class>at.asit.formatdetection.impl.PDFFormatDetector</cfg:Class>
    <cfg:FullName>Portable Document Format</cfg:FullName>
    <cfg:ShortName>PDF</cfg:ShortName>

  <cfg:FormatDetector id="pdf-as">
    <cfg:Class>at.asit.formatdetection.impl.PDFASFormatDetector</cfg:Class>
    <cfg:FullName>PDF Signatur</cfg:FullName>
    <cfg:ShortName>PDF-AS</cfg:ShortName>
  </cfg:FormatDetector>

</cfg:FormatDetector>

  <cfg:FormatDetector id="multipurpose_internet_mail_extension">
    <cfg:Class>at.asit.formatdetection.impl.mime.MimeFormatDetector</cfg:Class>
    <cfg:FullName>Multipurpose Internet Mail Extension</cfg:FullName>
    <cfg:ShortName>smime-signed-data</cfg:ShortName>
  </cfg:FormatDetector>

</cfg:FormatDetector>
</cfg:FormatDetectionConfiguration>
```

Jeder Block entspricht einem bestimmten Dokumentformat. Ein Block besteht aus folgenden Komponenten (**fett** gekennzeichnet sind die variablen Teile).

```
<cfg:FormatDetector id="eindeutiger Bezeichner">
  <cfg:Class>implementierende Klasse</cfg:Class>
  <cfg:Version>Versionsnummer des Dokumentformats</cfg:Version>
  <cfg:FullName>vollständiger Name des Dokumentformats</cfg:FullName>
  <cfg:ShortName>Kurzbezeichnung des Dokumentformats</cfg:ShortName>
</cfg:FormatDetector>
```

eindeutiger Bezeichner: Dieser Bezeichner stellt eine eindeutige Identifikation des Dokumentformats dar. Über diesen Bezeichner wird im Verifikationsteil der Anwendung ein entsprechender Verifikator ausgewählt. Als Bezeichner können beliebige Texte (die jedoch keinen Beistrich beinhalten dürfen) gewählt werden – solange diese eindeutig innerhalb dieser Konfiguration sind.

¹⁶ Die abgebildete Datei soll die Konfiguration verdeutlichen und ist deshalb in verkürzter Form dargestellt.

implementierende Klasse: bezeichnet die Klasse, die ein bestimmtes Dokumentformat erkennen kann. (z.B. `at.asit.formatdetection.impl.XMLFormatDetector`). Jede Klasse muss das Interface `at.asit.formatdetection.FormatDetector` implementieren.

Versionsnummer des Dokumentformats (optional): Hiermit kann eine Versionsbezeichnung angegeben werden, die auch im Prüfprotokoll aufscheint. Als Versionsnummer kann ein beliebiger Text angegeben werden (z.B. "1.0.1rev2").

vollständiger Name des Dokumentformats (optional): Hiermit kann ein Begriff definiert werden, der das Dokumentformat bezeichnet. (z.B. "XML Digital Signature").

Kurzbezeichnung des Dokumentformats (optional): Hiermit kann eine Kurzbezeichnung für das Dokumentformat definiert werden (z.B. "XMLDsig").

4.3.3 Signaturprüfservice

Die Konfigurationsdatei für das Signaturprüfservice ist unter

```
%TOMCAT_DIR%\conf\signature-verification\application_config.xml
```

zu finden. Sollte ein davon abweichender Konfigurationspfad erwünscht sein, kann dieser über das System-Property "signatureverification.configuration" definiert werden.

Das folgende Listing zeigt zum besseren Verständnis eine leicht verkürzte Darstellung.

```
<properties>

<category name="general">
  <fileupload.maxfilesize>20971520</fileupload.maxfilesize>
</category>

<category name="error">
  <mailto>technology@a-sit.at</mailto>
  <subject>Fehlerbenachrichtigung</subject>
</category>

<category name="pdf-as">
  <resource.path>${catalina.base}/conf/signaturpruefservice/pdf-as</resource.path>
  <signature.type>SIGNATUR</signature.type>
  <sign.report>>true</sign.report>
  <signature.mode>textual</signature.mode>
</category>

<category name="moa">
  <category name="sp">
    <connection.url>http://localhost:30080/moa-spss/services/SignatureVerification</connection.url>
    <trustprofile.id>signature-verification-trustprofile-id</trustprofile.id>
  </category>
</category>

<category name="format-to-verifyclient">
  <!-- category names like "xmldsig" may be chosen arbitrarily -->
  <category name="xmldsig">
    <format.id>
      xml,
      birthcertificate/20060814#
    </format.id>
    <verifier.impl>at.asit.verifysignaturetool.verification.client.impl.XMLVerifyClientImpl</verifier.impl>
  </category>
  <category name="identitylink">
    <format.id>urn:oasis:names:tc:SAML:1.0:assertion:identitylink</format.id>
    <verifier.impl>
      at.asit.verifysignaturetool.verification.client.impl.PersonenbindungVerifyClientImpl
    </verifier.impl>
  </category>
  <category name="mandate">
    <format.id>http://reference.e-government.gv.at/namespace/mandates/20040701#</format.id>
    <verifier.impl>at.asit.verifysignaturetool.verification.client.impl.MandateVerifyClientImpl</verifier.impl>
  </category>
  <category name="pdf-as">
    <format.id>pdf-as</format.id>
    <verifier.impl>at.asit.verifysignaturetool.verification.client.impl.PDFASVerifyClientImpl</verifier.impl>
  </category>
  <category name="smime">
    <format.id>multipurpose_internet_mail_extension</format.id>
    <verifier.impl>at.asit.verifysignaturetool.verification.client.impl.SMimeVerifyClientImpl</verifier.impl>
  </category>
</category>

<category name="known_egov_oids">
  <root.oid>1.2.40.0.10</root.oid>
</category>
```

```

<category name="1.2.40.0.10.1.1.1">
  <name>Verwaltungseigenschaft</name>
  <description>Der Zertifikatsinhaber gehört einer Verwaltungsorganisation an.</description>
</category>
<category name="1.2.40.0.10.1.1.2">
  <name>Dienstleistereigenschaft</name>
  <description>
    Der Zertifikatsinhaber ist im Auftrag (als Dienstleister) einer öffentlichen Verwaltungsorganisation
    tätig.
  </description>
</category>
<category name="1.2.40.0.10.3.1">
  <name>Notarseigenschaft</name>
</category>
<category name="1.2.40.0.10.3.2">
  <name>Rechtsanwaltseigenschaft</name>
</category>
<category name="1.2.40.0.10.3.3">
  <name>Ziviltechnikereigenschaft</name>
</category>
<category name="1.2.40.0.10.1.7.1">
  <name>Eigenschaft zur Ausstellung von Personenbindungen</name>
</category>
<category name="1.2.40.0.10.1.7.2">
  <name>Eigenschaft zur Eintragung von elektronischen Vollmachten</name>
</category>
<category name="1.2.40.0.10.3.10">
  <name>Organwälttereigenschaft</name>
</category>
</category>

<category name="certificate_annotation">
  <!-- category names like "ECard VSig CA 2" may be chosen arbitrarily -->

  <category name="ECard VSig CA 2">
    <issuer.name>C=AT,O=Hauptverband österr. Sozialvers.,CN=VSig CA 2</issuer.name>
    <annotation>E-Card Verwaltungssignatur</annotation>
  </category>

  <category name="Al SIGNATUR">
    <issuer.name>CN=Al SIGNATUR,OU=Al.net,O=mobilkom austria AG & amp; Co KG,C=AT</issuer.name>
    <annotation>Al Verwaltungssignatur</annotation>
  </category>
</category>

<category name="ldap-mapping">
  <!-- sub-category names like "a-sign-Premium-Sig-01" may be chosen arbitrarily -->

  <category name="a-sign-Premium-Sig-01">
    <issuer.name>CN=a-sign-Premium-Sig-01,OU=a-sign-Premium-Sig-01,O=A-Trust Ges. f. Sicherheitssysteme im
    elektr. Datenverkehr GmbH,C=AT</issuer.name>
    <ldap.url>ldap://ldap.a-trust.at/ou=a-sign-Premium-Sig-01,o=A-Trust,c=at</ldap.url>
  </category>

  <category name="a-sign-Premium-Sig-02">
    <issuer.name>CN=a-sign-Premium-Sig-02,OU=a-sign-Premium-Sig-02,O=A-Trust Ges. f. Sicherheitssysteme im
    elektr. Datenverkehr GmbH,C=AT</issuer.name>
    <ldap.url>ldap://ldap.a-trust.at/ou=a-sign-Premium-Sig-02,o=A-Trust,c=at</ldap.url>
  </category>

  <category name="a-sign-Premium-Enc-01">
    <issuer.name>CN=a-sign-Premium-Enc-01,OU=a-sign-Premium-Enc-01,O=A-Trust Ges. f. Sicherheitssysteme im
    elektr. Datenverkehr GmbH,C=AT</issuer.name>
    <ldap.url>ldap://ldap.a-trust.at/ou=a-sign-Premium-Enc-01,o=A-Trust,c=at</ldap.url>
  </category>

  <category name="a-sign-Premium-Enc-02">
    <issuer.name>CN=a-sign-Premium-Enc-02,OU=a-sign-Premium-Enc-02,O=A-Trust Ges. f. Sicherheitssysteme im
    elektr. Datenverkehr GmbH,C=AT</issuer.name>
    <ldap.url>ldap://ldap.a-trust.at/ou=a-sign-Premium-Enc-02,o=A-Trust,c=at</ldap.url>
  </category>

  <category name="a-sign-Corporate-Light-01">
    <issuer.name>CN=a-sign-corporate-light-01,OU=a-sign-corporate-light-01,O=A-Trust Ges. f. Sicherheitssysteme
    im elektr. Datenverkehr GmbH,C=AT</issuer.name>
    <ldap.url>ldap://ldap.a-trust.at/ou=a-sign-corporate-light-01,o=A-Trust,c=at</ldap.url>
  </category>

  <category name="a-sign-Corporate-Light-02">
    <issuer.name>CN=a-sign-corporate-light-02,OU=a-sign-corporate-light-02,O=A-Trust Ges. f. Sicherheitssysteme
    im elektr. Datenverkehr GmbH,C=AT</issuer.name>
    <ldap.url>ldap://ldap.a-trust.at/ou=a-sign-corporate-light-02,o=A-Trust,c=at</ldap.url>
  </category>

  <category name="a-sign-Corporate-Light-03">
    <issuer.name>CN=a-sign-corporate-light-03,OU=a-sign-corporate-light-03,O=A-Trust Ges. f. Sicherheitssysteme
    im elektr. Datenverkehr GmbH,C=AT</issuer.name>
    <ldap.url>ldap://ldap.a-trust.at/ou=a-sign-corporate-light-03,o=A-Trust,c=at</ldap.url>
  </category>

```

```

<category name="ECard VSig CA 2">
  <issuer.name>C=AT,O=Hauptverband österr. Sozialvers.,CN=VSig CA 2</issuer.name>
  <ldap.url>ldap://ldap.ecard.sozialversicherung.at/ou=VSig CA 2,o=Hauptverband österr.
  Sozialvers.,c=at</ldap.url>
  <serialnumber.attrname>serialNumber</serialnumber.attrname>
</category>

</category>
</properties>

```

Die Konfiguration ist ähnlich wie jene für die Dokumenterkennung aufgebaut. Sie gliedert sich in mehrere Teile (Kategorien) die nun im Folgenden erläutert werden.

4.3.3.1 Kategorie "general"

In dieser Rubrik kann die maximale erlaubte Größe von zu prüfenden Dateien festgelegt werden. Der Schlüssel `<fileupload.maxfilesize>` muss die maximale Größe in Bytes enthalten.

Mit dem Schlüssel `<info.url>` kann eine URL zu einer Informationsseite angegeben werden. Die URL kann dann über den großen "Info"-Button in der Kopfzeile jeder Seite aufgerufen werden. Der Parameter ist optional.

4.3.3.2 Kategorie "error"

Im Falle eines unerwarteten Fehlers wird dem Benutzer eine Fehlerseite mit einem Hinweis einen Verantwortlichen unter einer bestimmten E-Mail-Adresse zu kontaktieren präsentiert. Bei Klick auf die E-Mail-Adresse öffnet sich eine leere E-Mail-Nachricht des Standard-E-Mail-Client des Anwenders bei dem die Empfänger-Adresse (`<mailto>`) sowie der Betreff (`<subject>`) bereits vorausgefüllt sind.

4.3.3.3 Kategorie "pdf-as"

Diese Kategorie umfasst die Bezeichnung des Konfigurationsverzeichnis (`<resource.path>`) von PDF-AS (siehe auch Abschnitt 4.3.5) sowie das zur Signatur des Prüfprotokolls verwendete PDF-AS-Profil (`<signature.type>`). Um die Mehrsprachigkeit auch in der Signatur des Prüfprotokolls einfließen lassen zu können wurde an dieser Stelle kein absoluter Profilname sondern wiederum ein Platzhalter (siehe Abschnitt 4.3.1) eingetragen. Der eigentliche Profilname ist im Sprach-Resource-Bundle konfiguriert.

Die Festlegung des Konfigurationsverzeichnis kann absolut, relativ oder in Kombination mit System-Properties wie `catalina.base`, `catalina.home` etc. erfolgen:

z.B. `#{catalina.base}/conf/signaturpruefservice/pdf-as`

Der Schlüssel `<sign.report>` legt fest, ob das Prüfprotokoll signiert werden soll (→ "true") oder nicht (→ "false").

Der Schlüssel `<signature.mode>` legt fest, ob das Prüfprotokoll textuell (`textual`) oder binär (`binary`) signiert werden soll. Default-Wert ist `textual`.

Mit dem Schlüssel `<signeddata.source>` kann festgelegt werden, ob die Anzeige signierter Daten angeboten werden soll bzw. wenn ja, dann ob die Daten von MOA-SP oder von PDF-AS entnommen werden. Die MOA-Variante ist empfohlen.

Erlaubte Werte für diesen Schlüssel sind:

- OFF: Die signierten Daten werden nicht angezeigt.
- MOA: Die der Signatur zu Grunde liegenden Daten werden von MOA-SP entnommen.
- PDFAS: Die der Signatur zu Grunde liegenden Daten werden von PDF-AS entnommen.

4.3.3.4 Kategorie "moa"

Die Festlegung der URL (`<connection.url>`) der MOA-Signaturprüfung sowie das zu verwendende TrustProfile (`<trustprofile.id>`) wird in der Unterkategorie "sp" definiert.

4.3.3.5 Kategorie "format-to-verifyclient"

Dieser Abschnitt umfasst die Zuordnung von Dokumentformaten über deren Identifier (siehe Abschnitt 4.3.2) zu bestimmten Verifikatoren. Diese Verifikatoren extrahieren – auf geeignete Weise – signaturrelevante Informationen und geben diese an die MOA-Signaturprüfung weiter.

Jedem Verifier ist eine eigene Kategorie gewidmet, deren Name frei gewählt werden kann. Jede dieser Unter-Kategorien besteht aus folgenden Elementen:

`<format.id>` hier werden die Identifier der zu verknüpfenden Dokumentformate angegeben. Im Regelfall wird für jeden Verifikator ein bestimmtes Dokumentformat festgelegt. Es ist jedoch auch möglich mehrere Dokumentformate einem Verifikator zuzuordnen. In diesem Fall sind die Identifier durch Komma von einander zu trennen.
z.B. `xml, birthcertificate/20060814#`

`<verifier.impl>` Hiermit wird die konkrete Implementierung eines Verifiers angegeben. Jeder Verifier muss das Interface `at.asit.verifysignaturetool.verification.client.VerifyClient` implementieren.

4.3.3.6 Kategorie "format-filter"

Hier können spezielle Filter definiert werden, die nach einer Signaturprüfung aufgerufen werden. Hiermit ist es möglich, dokumentformatspezifische Anmerkungen im Prüfprotokoll unterzubringen.

4.3.3.7 Kategorie "erechnung"

Diese Kategorie ist eigens für den Bereich "Elektronische Rechnung" geschaffen worden. Hier können E-Rechnungs-spezifische Einstellungen (z.B. erlaubte E-Rechnungs-Aussteller oder nicht erlaubte E-Rechnungs-Aussteller) getroffen werden.

4.3.3.8 Kategorie "known_egov_oids"

X.509-Zertifikate erlauben die Aufnahme von zusätzlichen Attributen in Form von Zertifikatserweiterungen. Spezielle – für die öffentliche Verwaltung relevante – Attribute sind über deren Object Identifier (OID) erkennbar (siehe auch [OID]). Diese OIDs besitzen einen gemeinsamen Wurzel-Identifier (1.2.40.0.10).

Über die Kategorie "known_egov_oids" werden die einzelnen Eigenschaften über ihre OIDs registriert und mit einem entsprechenden Text (wie z.B. "Verwaltungseigenschaft") versehen. Dieser Text scheint anstelle der OIDs im Prüfprotokoll auf.

Der oben angesprochene Wurzel-Identifier wird mit dem Schlüssel `<root.oid>` festgelegt.

Jede registrierte OID wird über eine eigene Kategorie abgebildet. Der Name der Kategorie enthält die jeweilige OID (z.B. `<category name="1.2.40.0.10.1.1.1">`).

Jede Kategorie kann folgende zwei Schlüssel enthalten:

`<name>` Dies kennzeichnet den zur jeweiligen OID gehörenden Namen, der auch im Prüfprotokoll aufscheint (z.B. "Verwaltungseigenschaft").

`<description>` Dieser Parameter ist optional und erlaubt die Angabe einer genaueren Erläuterung (z.B. "Der Zertifikatsinhaber gehört einer Verwaltungsorganisation an.").

4.3.3.9 Kategorie "certificate_annotation"

Vielfach erscheint es nützlich geprüfte Dokumente mit Signaturen von Zertifikaten eines bestimmten Ausstellers mit speziellen Anmerkungen zu versehen. Dies erlaubt es beispielsweise eine E-Card- bzw. A1-Verwaltungssignatur zu kennzeichnen.

Jede Anmerkung wird über eine Unter-Kategorie realisiert, deren Name frei wählbar ist. Jede Kategorie besitzt folgende zwei Parameter:

- <issuer.name> enthält den Distinguished Name des Zertifikat-Ausstellers in einem RFC2253 ([RFC2253]) konformen Format (z.B. "C=AT,O=Hauptverband österr. Sozialvers.,CN=VSig CA 2")
- <annotation> enthält eine Anmerkung in Form eines beliebigen Texts. Die Anmerkung wird Prüfprotokollen für Dokumente basierend auf Zertifikaten des jeweiligen Ausstellers hinzugefügt.

4.3.3.10 Kategorie "ldap-mapping"

Zur Verifikation von signierten Dokumenten ist stets das der Signatur zu Grunde liegende Zertifikat erforderlich. Befindet sich das Zertifikat nicht eingebettet im Dokument ist es erforderlich, dieses über einen LDAP-Verzeichnisdienst herunterzuladen.

Die Kategorie "ldap-mapping" verknüpft nun Ausstellernamen (Distinguished Names im RFC2253 ([RFC2253]) Format) mit LDAP-URLs. Diese Verknüpfung findet beispielsweise bei CMS-Dokumenten ohne Signaturzertifikat Verwendung. Jede Unterkategorie (der jeweilige Name ist frei wählbar) umfasst eine bestimmte Verknüpfung von Ausstellernamen zu LDAP-URL.

Eine Kategorie enthält folgende Schlüssel:

- <issuer.name> enthält den Distinguished Name des Zertifikat-Ausstellers in einem RFC2253 ([RFC2253]) konformen Format
- <ldap.url> enthält die URL über die ein bestimmter LDAP-Server angesprochen werden kann
- <serialnumber.attrname> enthält den Namen des Attributs das die Seriennummer auf dem jeweiligen LDAP-Server kennzeichnet. Der Parameter ist optional. Ist kein Parameter angegeben, wird "eidCertificateSerialNumber" angenommen.

4.3.3.11 Kategorie "logging"

In dieser Kategorie kann ein Logger registriert werden, der Informationen über durchgeführte Prüfungen aufzeichnet. Es wird empfohlen, diesen Logger über einen eigenen Log4j-Appender zu konfigurieren.

Der Logger muss das Interface `at.asit.verifysignaturetool.logger.SVLogger` implementieren.

Die Kategorie enthält folgenden Schlüssel:

- <svlogger.impl> bezeichnet die zu verwendende Logger-Implementierung. Wird der Konfigurationsschlüssel nicht verwendet, wird automatisch die interne Implementierung `at.asit.verifysignaturetool.logger.impl.SimpleSVLogger` verwendet. Diese zeichnet statistische Daten (Zeit, Ergebnis, Anzahl der Signaturen sowie Typ und Hashwert des geprüften Dokuments) auf, lässt jedoch keinen Rückschluss auf das geprüfte Dokument zu. Hinweis: Ist kein Logging erwünscht, kann folgende interne Implementierung angegeben werden: `at.asit.verifysignaturetool.logger.NoSVLogger`.

4.3.4 Apache Tomcat Servlet Container

Die Haupt-Konfigurationsdatei des Apache Tomcat Servlet Containers befindet sich unter

```
%TOMCAT_DIR%\conf\server.xml
```

Hier kann der Port für MOA-SPSS/Signaturprüfservice bzw. der Shutdown-Port geändert werden, falls es zu Konflikten mit Anwendungen kommt, die den vorgegebenen Ports 30080 bereits verwenden.

Warnung: Nach einer Port-Änderung darf nicht vergessen werden, diese auch in den Konfigurationen für das Signaturprüfservice (Abschnitt 4.3.3.3 und 4.3.3.4) sowie für PDF-AS (Abschnitt 4.3.5) nachzuziehen.

Weitere Informationen zur Konfiguration des Servlet Containers entnehmen Sie bitte der Apache Tomcat Dokumentation ([TOMCAT]).

4.3.5 PDF-AS

Die Konfigurationsdatei für die Komponente PDF-AS befindet sich unter

```
%TOMCAT_DIR%\conf\signature-verification\pdf-as\cfg\config.properties
```

Warnung: Der Speicherort der PDF-AS-Konfiguration kann über die Konfiguration des Signaturprüfservice angepasst werden. Der oben stehende Pfad bezieht sich auf den nach der Installation voreingestellten Pfad. Sollte dieser nachträglich durch den Anwender verändert werden muss dies gegebenenfalls berücksichtigt werden.

Für eine detaillierte Konfiguration sei auf die PDF-AS Dokumentation ([PDF-AS]) verwiesen. Jene Schlüssel, die für die Anpassung im Zusammenhang mit dem Signaturprüfservice relevant sind werden jedoch im Folgenden inkl. der voreingestellten Werte aufgezählt und erläutert.

Prüfung inkrementeller Updates

```
check_document = true
```

Diese Eigenschaft bestimmt die Behandlung inkrementeller Updates . Wird `true` gesetzt , werden inkrementelle Updates erkannt und wie in Abschnitt 3.3 erläutert, behandelt. Ein Wert `false` deaktiviert die Berücksichtigung inkrementeller Updates.

MOA-Serversignatur

```
moa.sign.url = http://localhost:30080/moa-spss/services/SignatureCreation
```

Kennzeichnet die URL mit der die Serversignaturfunktion von MOA genutzt werden kann.

```
moa.sign.KeyIdentifier = signature-verification-keygroup-id
```

Entspricht dem in der MOA-Konfiguration festgelegten KeyIdentifier und steht stellvertretend für den privaten Schlüssel mit dem das Prüfprotokoll unterzeichnet wird.

MOA-Signaturprüfung

```
moa.verify.url = http://localhost:30080/moa-spss/services/SignatureVerification
```

Kennzeichnet die URL mit der die Signaturprüffunktion von MOA genutzt werden kann.

```
moa.verify.TrustProfileID = signature-verification-trustprofile-id
```

Entspricht dem in der MOA-Konfiguration festgelegten Trustprofile welches Zertifikate als Vertrauensanker (siehe Fußnote 17) enthält.

PDF-AS Profilkonfiguration

Zur Erläuterung einiger Konfigurationsschlüssel wird folgender Parameter als Platzhalter verwendet.

Name	Beschreibung	vorgegebener Standardwert
%PROFIL%	Bezeichnet das PDF-AS-Signaturprofil	AMTSSIGNATURBLOCK_DE

Hinweis: Das voreingestellte Profil zur Signatur des Prüfprotokolls lautet "AMTSSIGNATURBLOCK_DE". Sollte der Profilname an dieser Stelle geändert werden muss gleichfalls die Konfiguration des Signaturprüfservice nachgezogen werden (siehe Abschnitt 4.3.3.3 bzw. 4.3.1).

`sig_obj.%PROFIL%.value.SIG_SUBJECT` = Signaturprüfservice v1.7.0

Dieser Text erscheint unter der Rubrik "Signator" im Signaturblock des Prüfprotokolls.

`sig_obj.%PROFIL%.value.SIG_META` = Prüfservice: <http://localhost:30080/signature-verification>

Dieser Text erscheint unter der Rubrik "Hinweis" im Signaturblock des Prüfprotokolls.

`sig_obj.%PROFIL%.value.SIG_LABEL` = ./images/signatur-logo.jpg

Dieser Schlüssel dient dazu eine Bildmarke für den Signaturblock zu definieren.

4.3.6 MOA-SPSS

Die Konfiguration für die Komponente MOA-SPSS ist im Verzeichnis

```
%TOMCAT_DIR%\conf\moa-spss
```

zu finden. Hier sind

- die Konfigurationsdatei `spss.config.xml`
- sowie der private Schlüssel `keys/signature-verification` für die Signatur des Prüfprotokolls in Form einer PKCS#12-Datei
- als auch das Trustprofil `trustProfiles/signature-verification` für die Verifikation in Form von X.509-Zertifikaten abgelegt.

Einrichtung des Signaturzertifikats für das Prüfprotokoll

Das neu einzurichtende Signaturzertifikat muss in Form einer PKCS#12-Datei vorliegen. Diese Datei repräsentiert einen passwortgeschützten Software-Schlüsselspeicher für private Schlüssel.

Zur Einrichtung sind folgende Schritte erforderlich:

1. Kopieren Sie Ihre neue Schlüsseldatei (z.B. `pkcs12_schluessel_datei.p12`) in den Ordner `keys/signature-verification`.
2. Öffnen der oben genannten Datei `spss.config.xml` mit einem Editor.
3. Geben Sie Ihre neue Schlüsseldatei durch Modifikation des folgenden Abschnitts bekannt:

```
<cfg:SoftwareKeyModule>  
  <cfg:Id>signature-verification-keymodule-id</cfg:Id>  
  <cfg:FileName>keys/signature-verification/pkcs12_schluessel_datei.p12</cfg:FileName>  
  <cfg>Password>passwort</cfg>Password>  
</cfg:SoftwareKeyModule>
```

Anstelle von `pkcs12_schluessel_datei.p12` ist der Name der neuen Schlüsseldatei, bzw. anstelle von `passwort` ist das entsprechende Passwort anzugeben.

4. Wählen Sie den Signaturschlüssel durch Anpassung des folgenden Abschnitts aus:

```
<cfg:KeyGroup>  
  <cfg:Id>signature-verification-keygroup-id</cfg:Id>  
  <cfg:Key>  
    <cfg:KeyModuleId>signature-verification-keymodule-id</cfg:KeyModuleId>  
    <cfg:KeyCertIssuerSerial>  
      <dsig:X509IssuerName>aussteller_distinguished_name</dsig:X509IssuerName>  
      <dsig:X509SerialNumber>seriennummer</dsig:X509SerialNumber>  
    </cfg:KeyCertIssuerSerial>  
  </cfg:Key>  
</cfg:KeyGroup>
```

Anstelle von `aussteller_distinguished_name` geben Sie bitte den RFC2253-konformen Distinguished Name des Ausstellers Ihres neuen Signaturzertifikats an (z.B. `CN=MOA Test CA,OU=EGIZ,O=TU Graz,C=AT`).

Für `seriennummer` ist die Seriennummer des neuen Signaturzertifikats im Dezimalformat (z.B. 21) anzugeben.

5. Schließlich darf nicht darauf vergessen werden, das neue Zertifikat (oder ein passendes Root- bzw. Intermediate-Zertifikat) als Vertrauensanker¹⁷ in das Verzeichnis `trustProfiles\signature-verification` zu kopieren. Andernfalls können Prüfprotokolle – signiert mit dem neuen Zertifikat – nicht mit dem Signaturprüfservice verifiziert werden.

Die Konfiguration von MOA-SPSS wird an dieser Stelle nicht weiter erläutert. Dazu sei auf die MOA-SPSS-Handbücher ([MOA-SPSS]) verwiesen.

¹⁷ Ein Vertrauensanker ist ein CA-Zertifikat, das explizit als vertrauenswürdig eingestuft wird. MOA-SP versucht bei der Konstruktion einer Zertifikatskette, einen Pfad vom Signaturzertifikat bis hin zu einem der konfigurierten Vertrauensanker zu finden. Gelingt dies, wird auch das Signaturzertifikat als vertrauenswürdig betrachtet, ansonsten nicht.

Referenzen

[MOA]	<i>MOA: Serversignatur (SS), Signaturprüfung (SP)</i> , Version 1.4.2 http://egovlabs.gv.at/projects/moa-idspss abgerufen aus dem WWW am 13.12.2007
[MOA-SPSS]	<i>MOA: Serversignatur (SS) und Signaturprüfung (SP) Handbücher</i> , Version 1.4 http://moa-idspss.egovlabs.gv.at/moa-spss-1.4.2/doc/handbook/index.html abgerufen aus dem WWW am 3.12.2007
[OID]	Bundeskanzleramt Österreich; Arno Hollosi, Herbert Leitold, Thomas Rössler, Robert Wollendorfer <i>Object Identifier der öffentlichen Verwaltung</i> , Version 1.0.7
[PDF-AS]	Wilfried Lackner, Wolfgang Prinz: <i>PDF-AS Amtssignatur für elektronische Aktenführung</i> , Version 2.2 https://demo.egiz.gv.at/plain/projekte/signatur_im_e_government/pdf_signatur
[RFC2253]	IETF Network Working Group: LDAP (v3): UTF-8 String Representation of Distinguished Names http://www.ietf.org/rfc/rfc2253.txt abgerufen aus dem WWW am 14.02.2007
[TOMCAT]	The Apache Software Foundation: <i>The Apache Tomcat 5.5 Servlet/JSP Container</i> http://tomcat.apache.org/tomcat-5.5-doc/index.html abgerufen aus dem WWW am 14.02.2007

Historie

Version 0.8	Datum 14.02.2007	Kommentar erster Entwurf
Ersteller Thomas Knall		
Version 0.9	Datum 14.02.2007	Kommentar Durchsicht
Ersteller Herbert Leitold		
Version 1.0	Datum 15.02.2007	Kommentar minimale Änderungen, Abschnitt bzgl. "Einrichtung als Windows-Service" hinzugefügt
Ersteller Thomas Knall		
Version 1.1	Datum 3.12.2007	Kommentar Anpassungen an die Version 1.2.5 der Anwendung.
Ersteller Thomas Knall		
Version 1.2	Datum 19.12.2007	Kommentar Anmerkungen zur Konfiguration über System-Properties sowie Ergänzung der Rubrik "pdf-as" bzgl. Flag zum Ein/Ausschalten der Signatur des Prüfprotokolls.
Ersteller		
Version 1.3	Datum 24.01.2008	Kommentar <ul style="list-style-type: none"> • Konfigurationsschlüssel "<providesignaturedata.url>" entfernt. • Neuen Schlüssel <info.url> eingeführt. • Neuen Schlüssel <signeddata.source> eingeführt.
Ersteller Thomas Knall		
Version 1.4	Datum 06.03.2008	Kommentar <ul style="list-style-type: none"> • Neuen Schlüssel <signature.mode> eingeführt (Wahl des Signatur-Modus <code>textual</code> oder <code>binary</code>). • Hinweis zu Inkrementellen Updates hinzugefügt. • Hinweis zur Anzeige der Signaturdaten hinzugefügt.
Ersteller Thomas Knall		
Version 1.5	Datum 20.06.2008	Kommentar Neue Kategorie <code>logging</code> mit Konfigurationsschlüssel <svlogger.impl> eingeführt (Wahl eines Loggers für das Prüfergebnis).
Ersteller Thomas Knall		
Version 1.6	Datum 25.06.2008	Kommentar Abschnitt mit Hinweisen zum Deployment eingefügt (Abschnitt 4.4)
Ersteller Thomas Knall		
Version 1.7	Datum 15.07.2013	Kommentar Anpassungen an die Version 1.7 der Anwendung
Ersteller Alexander Marsalek		

