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 2.2.0, 27.02.2025

Thomas Lenz – thomas.lenz@a-sit.at

Christof Rabensteiner – christof.rabensteiner@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) hochzuladen 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

Abbildungsverzeichnis

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.

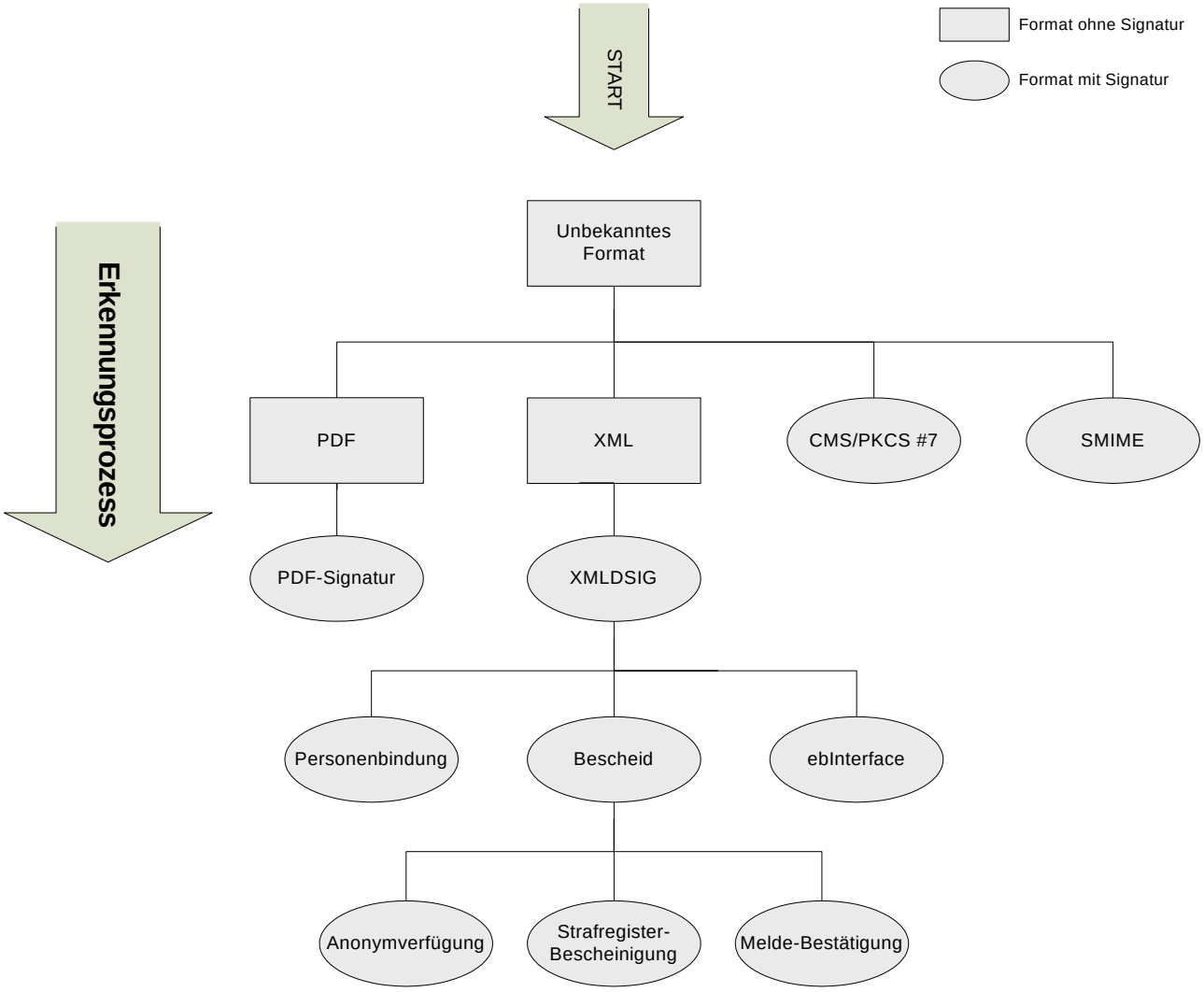

Abb. 2.1: Prinzip der Dokumentformat-Erkennung

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.

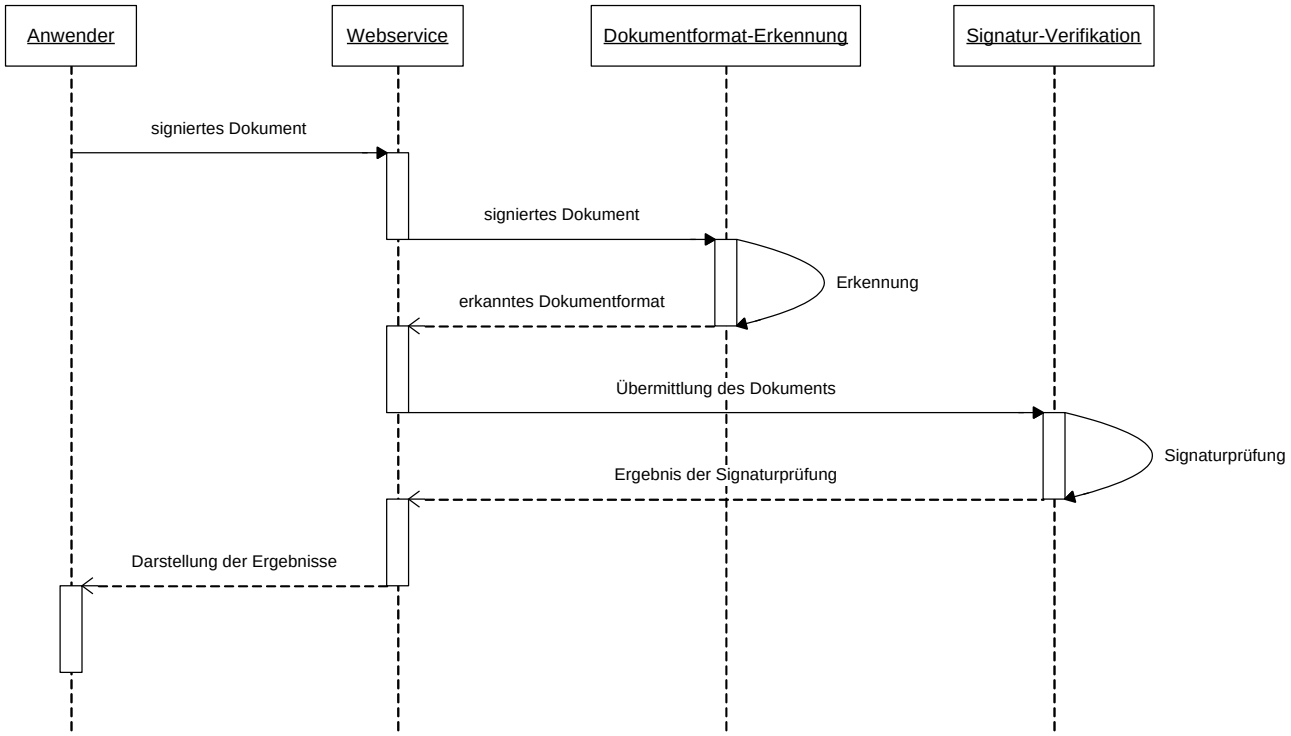

Abb. 2.2: zeitliche Abfolge eines typischen Prüfvorgangs

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, Version 8 oder höher
- Apache Tomcat, Version 8 oder höher

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-SPSS[1] (Modul für Online Applikationen – Serversignatur/Signaturprüfung) auf Basis der Version 3.1.0 genutzt. Die entsprechende MOA-Distribution ist im Installations-Paket (siehe Abschnitt 4.2) dieser Anwendung inkludiert.
Grundsätzlich können ältere Versionen bis 3.0.0 von MOA-SPSS verwendet werden. Dies kann aber zu Problemen mit der TrustList führen. Es wird empfohlen immer die aktuellste Version von MOA-SP/SS zu verwenden.

2.4.2 PDF-Amtssignatur

Die Signatur des Prüfprotokolls findet mittels PDF-Textsignatur 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-Toolkits[2] 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 Lizenzen[3] verfügbar.

[1] https://joinup.ec.europa.eu/software/moa-idspss/release/all
[2] https://jce.iaik.tugraz.at/products/
[3] https://jce.iaik.tugraz.at/sales/

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:8080/pruefdienst-frontend-app

aufgerufen werden. Wurde der Tomcat Container für einen anderen Port als 8080 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.

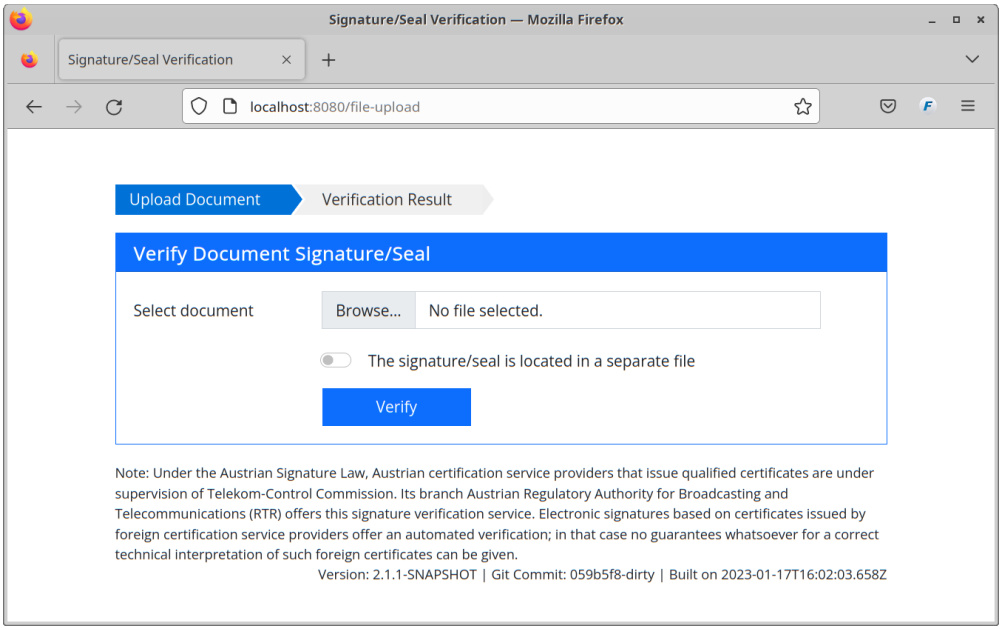

Abb. 3.3: Startseite des Signaturprüfservice

Nach Auswahl eines signierten Dokuments über den "Durchsuchen"-Button kann die Prüfung mit Klick auf "Prüfen" ausgelöst werden. Zur Prüfung von „Detached“-Signaturen muss „Signatur in separater Datei“ ausgewählt werden. Anschließend können die Signatur und die Referenzen ausgewählt und geprüft werden.

Nach erfolgter Prüfung wird dem Anwender eine Seite mit einer Zusammenfassung der Prüfergebnisse präsentiert (Abb. 3.4). 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. 3.4 gezeigten Ergebnisses handelt es sich beispielsweise um eine einzelne Signatur.

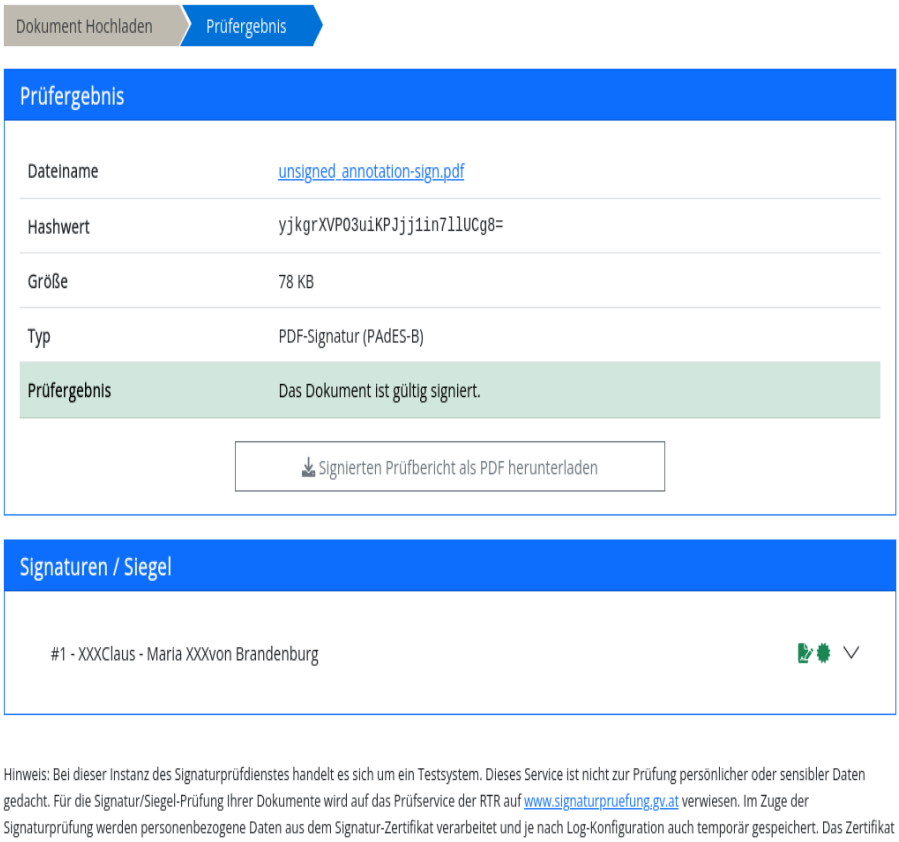

Abb. 3.4: Zusammenfassung des Prüfergebnisses

3.1 Interpretation des Signaturprüfergebnisses

Zunächst präsentiert sich dem Anwender eine Übersicht über alle geprüften Signaturen (Abb. 3.4, Abschnitt „Signaturen / Siegel“). Für jede Signatur im Dokument befindet sich in diesem Abschnitt eine Zeile mit der Ergebnisübersicht. Links steht der Name des Signatars, rechts die Prüfergebnisse in Form von mit eingefärbten Symbolen (siehe: Abb. 3.5): Signatur, Zertifikat, und Manifest. Je nach Signaturtyp können ein oder zwei Symbole fehlen. Eine Signatur gilt genau dann als **gültig** wenn alle Symbole grün gefärbt sind. Ist das Symbol gelb gefärbt, enthält das Dokument zwar eine gültige Signatur, es wurden jedoch Änderungen am Dokument erkannt welche eine manuelle Kontrolle erfordern.

Ein Dokument ist **gültig** wenn alle Signaturen **gültig** sind. Abb. 3.5 zeigt ein Signaturprüfergebnis mit einer gültigen Signatur. Zertifikat und Manifest sind jedoch nicht gültig. Das Dokument kann demnach nicht als gültig betrachtet werden.

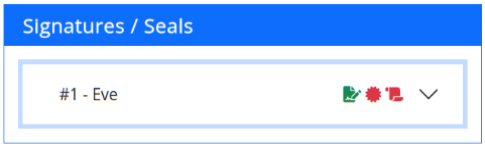

Abb. 3.5: Beispiel-Ergebnis (ungültiges Zertifikat & Manifest)

Anhand des eingefärbten Symbol für die Zertifikatsprüfung lässt sich auch bereits direkt unterscheiden ob des sich um ein qualifiziertes Zertifikat handelt oder nicht.

- Gewöhnliches Zertifikat
- Qualifiziertes Zertifikat

Ein Klick auf "Prüfprotokoll downloaden" liefert ein Prüfprotokoll (Abb. 3.6) das selbst mit einer PDF-Textsignatur versehen ist. Dies Protokoll enthält alle aus der Prüfung hervorgehenden Informationen.

Prüfbericht

Dokument

Dateiname	unsigned_annotation-sign.pdf
Hash-Wert	yjkgrXVPO3uiKPJjj1in7llUCg8= (SHA-1, Base64-kodiert)
Größe	78,58 kB
Typ	PDF-Signatur (PAdES-B)

Signatur/Siegel

Prüfungen

Signatur/Siegel- bzw. Prüfzeitpunkt (UTC)	2024-11-26T09:51:57Z
Signatur/Siegel	Die Überprüfung des Werts der Signatur bzw. des Siegels konnte erfolgreich durchgeführt werden.
Zertifikat	Eine formal korrekte Zertifikatskette vom Signatur/Siegel-Zertifikat zu einem vertrauenswürdigen Wurzelzertifikat konnte konstruiert werden. Jedes Zertifikat dieser Kette ist zum in der Anfrage angegebenen Prüfzeitpunkt gültig.

Zusatzinformationen

Signaturtyp/Siegeltyp	PAdES-B
Signaturalgorithmus	SHA256withECDSA
Die Signatur deckt den/die folgende/n Bereich/e an Bytes ab	0,33151,41345,39126

Unterzeichner/Siegelersteller

Name	XXXClaus - Maria XXXvon Brandenburg
Staat	AT
Seriennummer	dez.: 807147545276, hex.: bb:ed:be:22:bc

Aussteller

Name	a-sign-Test-Premium-Mobile-05
Organisationseinheit	a-sign-Test-Premium-Mobile-05
Organisation	A-Trust Ges. f. Sicherheitssysteme im elektr. Datenverkehr GmbH
Staat	AT

Zertifikat

Seriennummer	dez.: 1839421720, hex.: 6d:a3:59:18
Qualität	Gewöhnliches Zertifikat
zeitliche Gültigkeit	gültig von 2020-10-01T13:08:50Z bis 2025-10-01T13:08:50Z Der Prüfzeitpunkt liegt innerhalb des Gültigkeitszeitraumes.
Key Usage	Digitale Signatur, Nichtabstreitbarkeit
Zertifizierungsstatement	http://www.a-trust.at/docs/cp/a-sign-TEST Dieses Zertifikat dient nur zu Testzwecken

Hinweis: Bei dieser Instanz des Signaturprüfdienstes handelt es sich um ein Testsystem. Dieses Service ist nicht zur Prüfung persönlicher oder sensibler Daten gedacht. Für die Signatur/Siegel-Prüfung Ihrer Dokumente wird auf das Prüfservice der RTR auf verwiesen. Im Zuge der Signaturprüfung werden personenbezogene Daten aus dem Signatur-Zertifikat verarbeitet und je nach Log-Konfiguration auch temporär gespeichert. Das Zertifikat enthält üblicherweise den Namen des Unterzeichners, kann aber auch weitere personenbezogene Daten wie das Geburtsdatum, Email-Adressen oder Organisationszugehörigkeiten sowie alle anderen bei der Ausstellung angegebenen Daten enthalten. Dieses Testsystem dient zur Fehleranalyse und - behebung, weshalb zu Analysezwecken, je nach Log-Konfiguration auch das gesamte Dokument gespeichert werden kann. Im Fehlerfall kann es zudem Vorkommen, dass temporäre Daten nicht sofort gelöscht werden. Temporäre Daten sowie die Log-Dateien werden nach 5 Tagen automatisch gelöscht.

Abb. 3.6: Beispiel eines signierten Prüfprotokolls

Details zu jeder einzelnen Signatur (siehe Abb. 3.7) können durch Klick auf den jeweiligen Signatar-Namen in der Zusammenfassung des Prüfergebnisses aufgerufen werden.

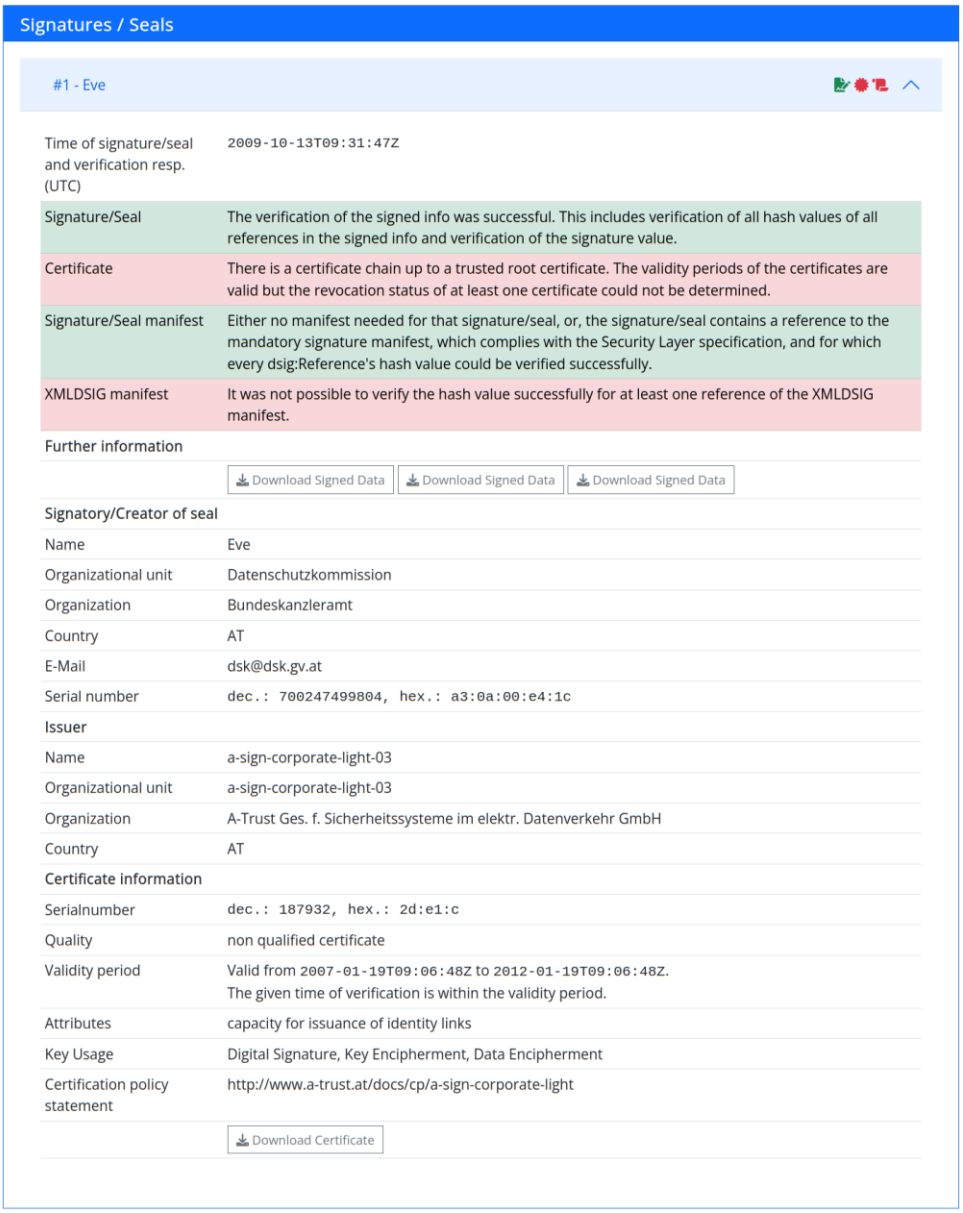

Abb. 3.7: Detailansicht einer Signaturprüfung

Der erste Abschnitt 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“[4] Zertifikat. Des Weiteren ist der Gültigkeitszeitraum des Zertifikats ersichtlich. Dies kann beispielsweise nützlich sein um ein negatives Zertifikatsprüfergebnis zu interpretieren.

[4] http://www.signatur.rtr.at/de/security/faq210.html

Die für die öffentliche Verwaltung relevanten Eigenschaften[5] eines Zertifikats werden anhand ihrer Object Identifier ([OID]) aufgelöst und dargestellt.

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 in der Detailansicht durch die Schaltfläche „Signierte Daten Herunterladen“ unterhalb der Prüfergebnisse angeboten (siehe Abb. 3.8).

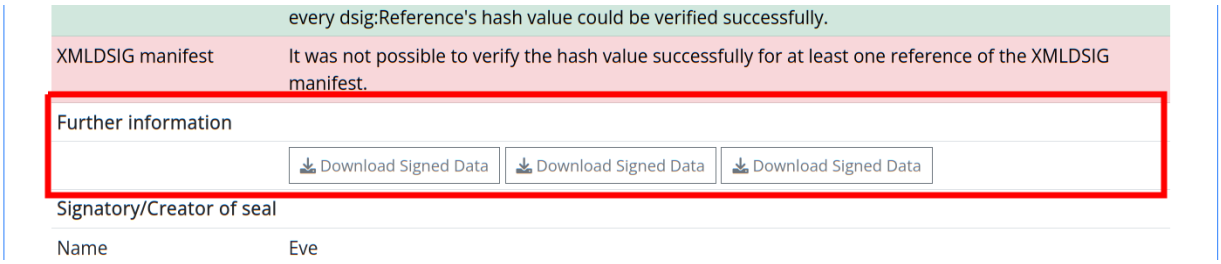

Abb. 3.8: Schaltfläche zum Herunterladen der Signaturdaten

Ein Klick auf die Schaltfläche "signierte Daten herunterladen" führt dazu dass der Browser die signierten Daten in Form einer Datei herunterlädt. 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.

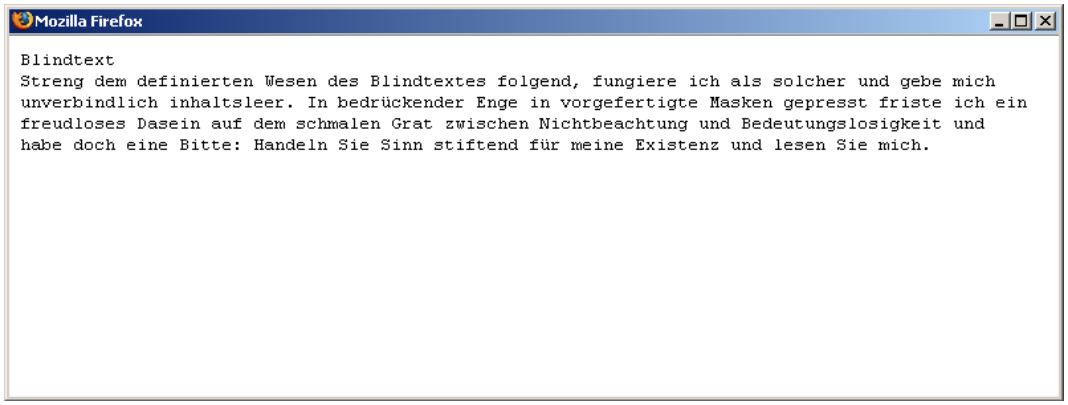

Abb. 3.9: Ansicht der Signaturdaten

[5] z.B. Verwaltungseigenschaft, Dienstleistereigenschaft, Notarseigenschaft, Rechtsanwaltseigenschaft, Ziviltechnikereigenschaft, Eigenschaft zur Ausstellung von Personenbindungen, Eigenschaft zur Eintragung von elektronischen Vollmachten oder Organwaltereigenschaft

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, zusätzliche Prüfinformationen, … angebracht werden.

Neu hinzugefügte Inhalte befinden sich allerdings in unsignierten Bereichen und werden einer erweiterten Prüfung unterzogen. Unabhängig vom Ergebnis dieser erweiterten Prüfung wird das Vorhandensein eines hinzugefügten Inhaltes auf jeden Fall mittels Kommentar angezeigt.

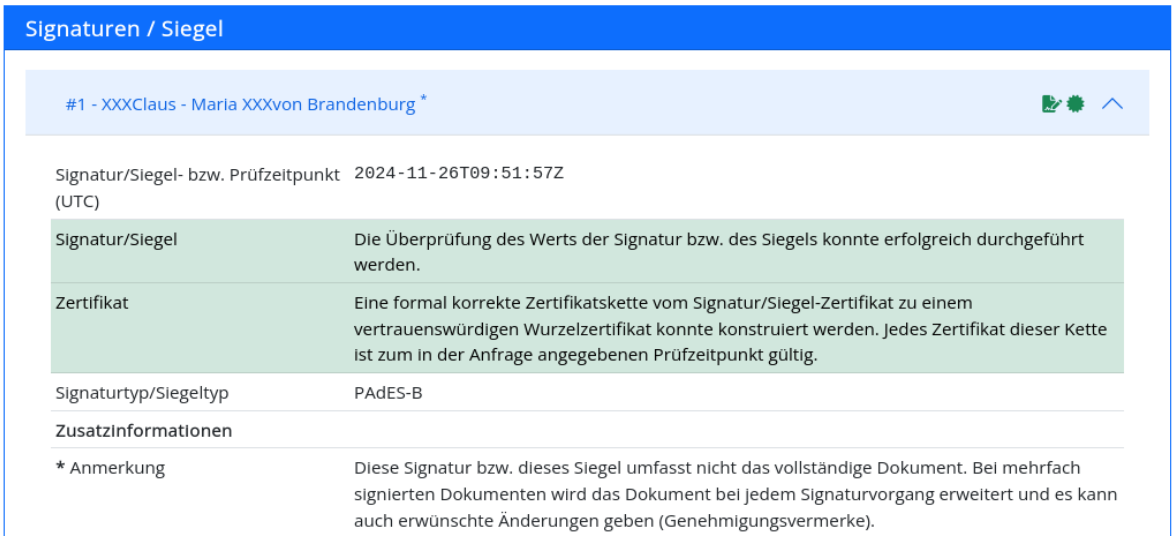

Abb. 3.10: PDF Signatur mit inkrementellen Update

Das Ergebnis dieser erweiterten Prüfung kann in zwei Kategorien unterteilt werden:

Kategorie#1: Die neu hinzugefügten Inhalte wurden als zulässig entsprechend der PDF Spezifikation und dem zu prüfenden Dokument bewertet. In diesem Fall wird die Signatur / das Siegel als „grün“ dargestellt, siehe Abbildung 3.10.

Kategorie#2: Die neu hinzugefügten Inhalte wurden als unzulässig entsprechend der PDF Spezifikation und dem zu prüfenden Dokument bewertet. In diesem Fall wird die Signatur / das Siegel als „gelb“ dargestellt, siehe Abbildung 3.11. In diesem Fall beinhaltet die Anmerkung zusätzliche Details zur Ursache wobei diese im nachfolgenden Abschnitt näher beschrieben werden. Es wird empfohlen, die Signaturdaten für diese markierten Signaturen zusätzlich zur Prüfung einzusehen (siehe auch Abschnitt 3.2).

Kategorie#2.1 „visuelle Änderung“

Das von dieser Signatur bzw. diesem Siegel umfasste Dokument enthält visuelle Änderungen. Um unerwünschte Veränderungen auszuschließen, sollten die Daten, die dieser Signatur bzw. diesem Siegel tatsächlich zu Grunde liegen, gesondert betrachtet werden.

In diesem Fall wurden im durch diese Signatur umfassten Bereich des Dokuments ein visueller Unterschied zum aktuellen Gesamtdokument gefunden welcher nicht einer zulässigen Änderungskategorie zugeordnet werden konnte. Zulässige Änderungen sind aus Sicht der PDF Spezifikation z.B. das Einfügungen von Annotations oder das Hinzufügen von

zusätzlichen Prüfinformationen. Aus praktischer Sicht handelt es sich hierbei zum Beispiel um folgende Änderungen (unvollständige Liste):

- Aufbringen einer weiteren Signatur inklusive Signaturblock als Signatureform entsprechend PDF Spezifikation
- Aufbringen von Kommentaren, Markierungen, Unterstreichen, … sofern es sich um Anmerkungen (Annotation) entsprechend PDF Spezifikation handelt
- Hinzufügen von Prüfinformationen für Long-Time (LT) Signaturformate

Sollte die Anmerkungen (Annotation) jedoch bereits zum Signaturzeitpunkt vorhanden gewesen sein und diese wurde nachträglich modifiziert, z.B. eine Markierung entfernt, ein Kommentar verschoben, …, dann handelt es sich hierbei dennoch um eine visuelle Änderung.

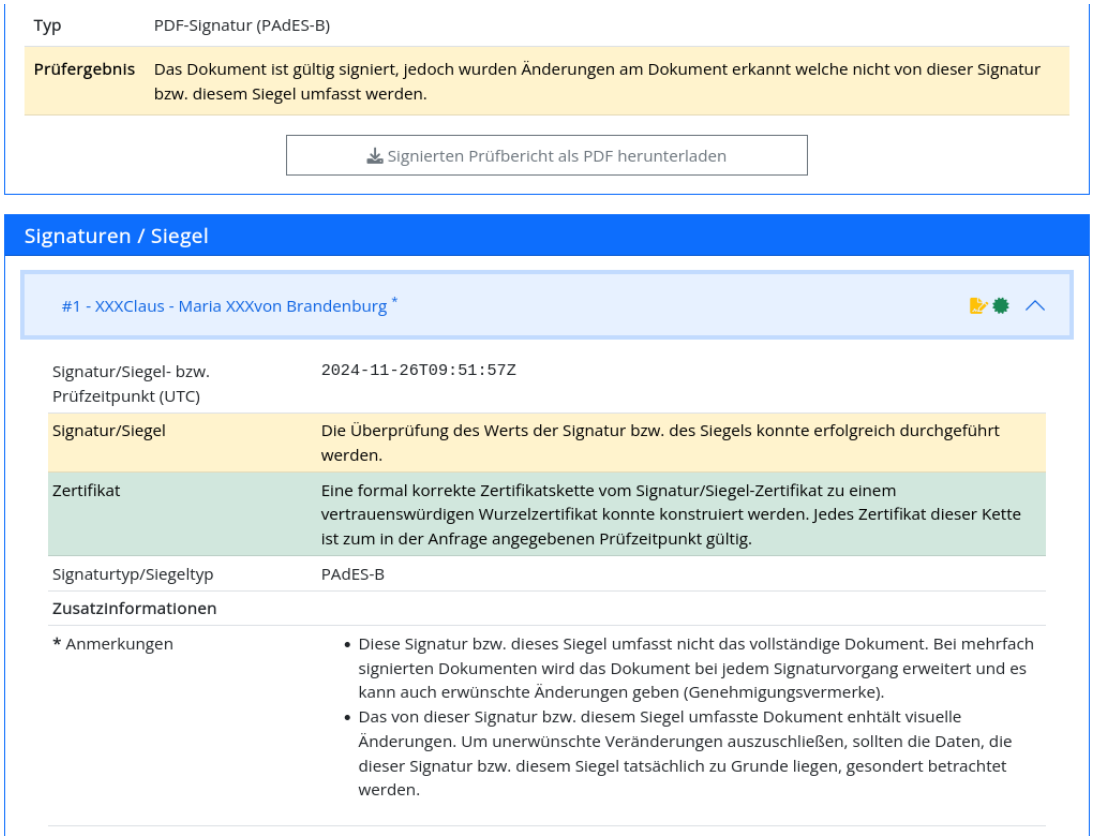

Abb. 3.11: PDF Signatur mit visueller Änderung

Kategorie#2.2 „neue Seite(n) hinzugefügt“

Das von dieser Signatur bzw. diesem Siegel umfasste Dokument enthält zusätzliche Seiten welche zum Signaturzeitpunkt nicht vorhanden waren.

In diesem Fall gibt es einen Unterschied in der Seitenanzahl des durch diese Signatur umfassten Bereichs des Dokuments im Vergleich zum aktuellen Gesamtdokument. Das Hinzufügen weiter Seiten in ein bereits signiertes PDF Dokument ist jedoch aus Sicht der PDF Spezifikation nicht zulässig.

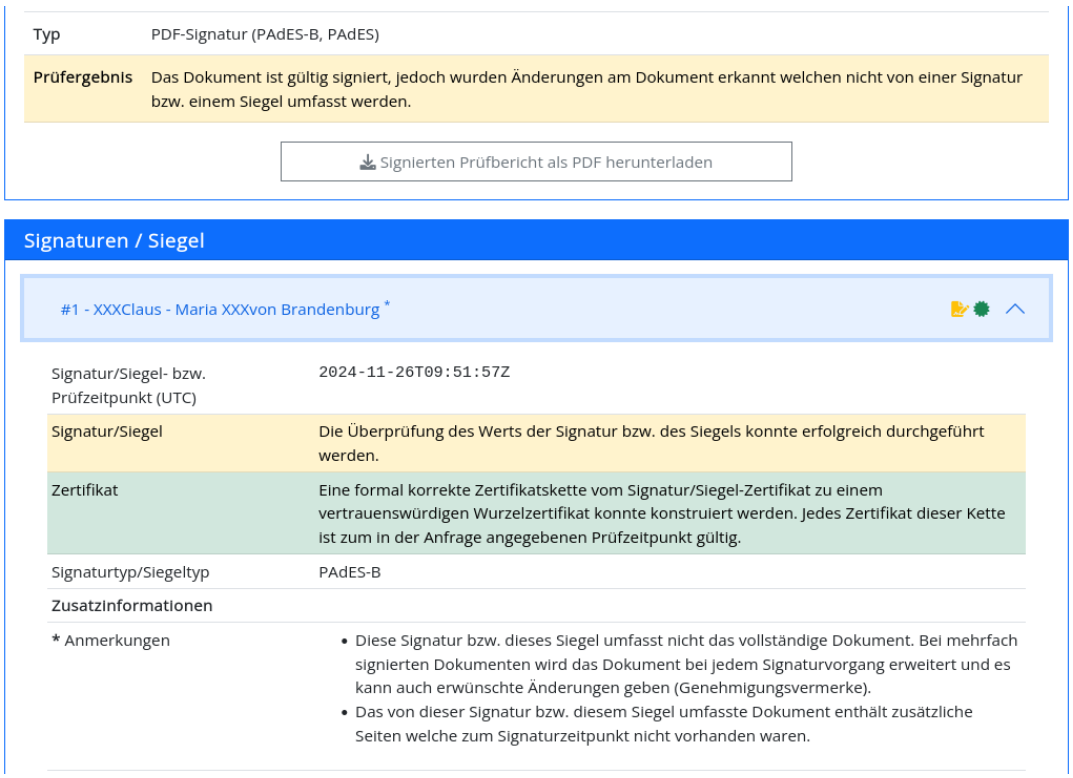

Abb. 3.12: PDF Signatur mit zusätzlicher Seite

Kategorie#2.3 „Anmerkungen / Kommentare überlappen sich“

Das von dieser Signatur bzw. diesem Siegel umfasste Dokument enthält Kommentare, Vermerke oder Signaturblöcke welche andere Teile des Dokuments überdecken.

In diesem Fall beinhaltet das PDF Dokument Anmerkungen (Annotation), wie zum Beispiel Kommentare, Vermerke, Signaturblöcke, …, welche sich gegenseitig überdecken. Da das Aufbringen von Anmerkungen (Annotation) entsprechend der Beschreibung in Kategorie#2.1 zulässig sein kann, wird die Überlagerung jedoch als problematisch bewertet da in diesem Fall nicht ausgeschlossen werden kann, dass Anmerkungen bewusst überdeckt wurden.

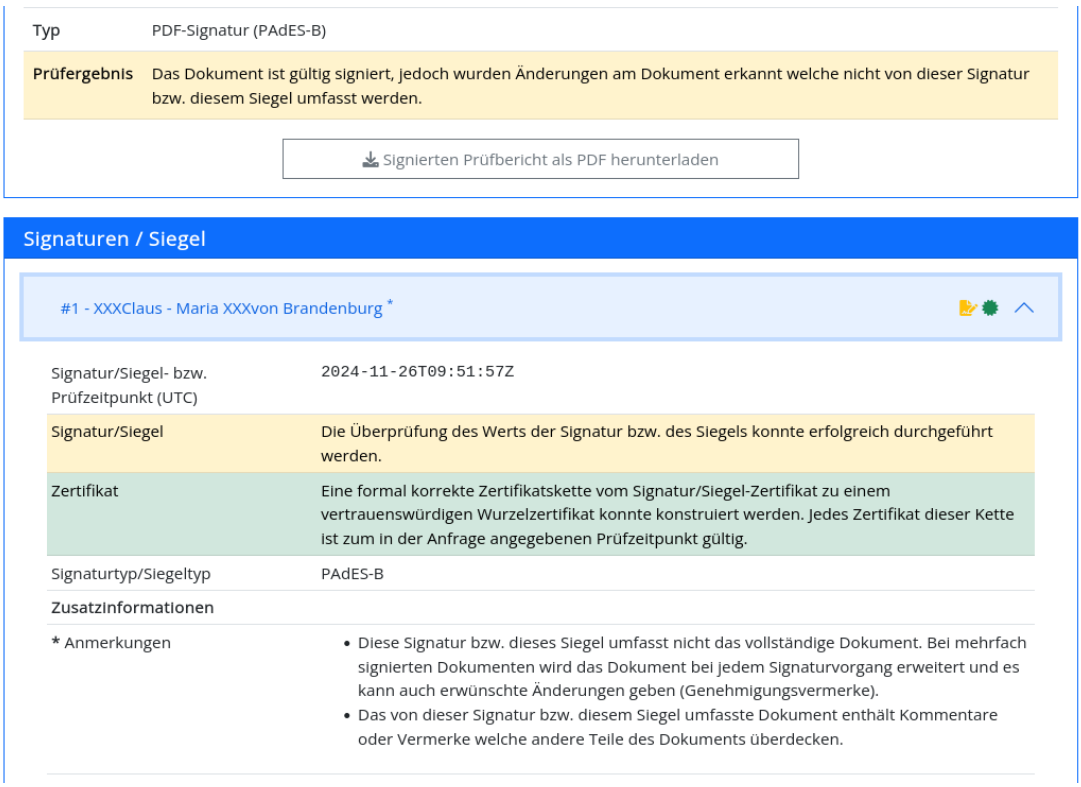

Abb. 3.13: PDF Signatur mit überlappenden Kommentaren

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 4 GB Hauptspeicher,
- mindestens 1GB freien Festplattenspeicher
- sowie einem aktuellem Multicore-Prozessor.

Außerdem werden für den Betrieb des Signaturprüfdienstes eine Java-Runtime-Environment sowie ein Apache-Tomcat-Servlet Container inkl. dem Signaturprüfservice benötigt. Zusätzlich wird ein zweiter Tomcat mit einer aktuellen MOA-SP/SS Version benötigt.

Die Java-Installation für das Signaturprüftool muss folgende Punkte erfüllen:

- Es muss mindestens Java 17 oder neuer verwendet werden.
- Es werden Apache Tomcat Versionen ab v10 unterstützt[6]. Dieses Dokument beschreibt die Installation des Apache Tomcat Servers anhand der Version 10.1.36[7] für Windows.

Die Java-Installation für MOA-SPSS muss folgende Punkte erfüllen:

- Es muss mindestens Java 8 oder neuer verwendet werden.
- Für die Verwendung von MOA wird ein JDK benötigt. Das JDK muss mit den "Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files" gepatcht worden sein. Für Details wird auf die Installationsanleitung von MOA-SP/SS verwiesen [MOA-SP/SS].
- Es werden Apache Tomcat Versionen ab v8 unterstützt[8]

Nach der Installation der Software hat der Anwender grundsätzlich die Möglichkeit die Anwendung über ein DOS-Fenster laufen zu lassen oder diese als Systemdienst einzurichten[9].

4.2 Installation der Software

Das Installations-Paket besteht aus der Beispielkonfiguration und dem Signaturprüfdienst Web Archive. Das Web Archive „signature-verification.war“ muss in den Ordner „webapps“ im Tomcat Installationsverzeichnis kopiert werden. Die Beispielkonfiguration muss in den Ordner „config“ im Tomcat Installationsverzeichnis entpackt werden. Der Ordner „scripts“ enthält ein Script zum Starten des Tomcats und eines zum Stoppen des Tomcats. Bevor diese Scripte verwenden werden können muss im Script „setVariables.bat“ der Pfad zum Tomcat (TOMCAT_DIR) und der Pfad der Java Installation (TOMCAT_DIR) angepasst werden. Abschnitt 4.3 beschreibt die Konfigurationsmöglichkeiten. Die Beispielkonfiguration nimmt an, dass die MOA-SP/SS Instanz unter „localhost:12380/moa-spss“ erreichbar ist und der KeyIdentifier „signature-verification-keygroup-id“ sowie die TrustProfileID „signature-verification-trustprofile-id“ konfiguriert sind. Für Details wird auf Abschnitt 4.3.6 bzw. [MOA] und [MOA-SPSS] verwiesen. Aus Sicherheitsgründen wird empfohlen ein separates temporäres Verzeichnis für den Prüfdienst zu konfigurieren.

---

6 http://tomcat.apache.org/
7 https://tomcat.apache.org/download-10.cgi
8 http://tomcat.apache.org/
9 https://tomcat.apache.org/tomcat-10.1-doc/windows-service-howto.html

4.3 Konfiguration

Im Prüfdienst Release Archiv befindet sich der `conf/` Ordner, welcher eine Vorlage für die Konfiguration des Prüfdienstes beinhaltet. Der Inhalt dieses Ordners kann nach `%TOMCAT_DIR%\conf` kopiert werden um den Prüfdienst mit der Default Konfiguration zu betreiben. In diesem `conf/` Ordner können einzelne Komponenten des Prüfdienstes 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 (`labels_de.properties`) und einer Sprachdatei für Englisch (`labels.properties`). Diese Dateien befinden sich im Konfigurationsordner `conf/signature-verification/frontend-app/i18n`.

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 Konfigurationsdatei[10] 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="asic">
        <cfg:Class>at.asit.formatdetection.impl.asic.ASICFormatDetector</cfg:Class>
        <cfg:FullName>#{documentformat.asic.fullname}</cfg:FullName>
        <cfg:ShortName>asic</cfg:ShortName>
      </cfg:FormatDetector>

    <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>
```

[10] Die abgebildete Datei soll die Konfiguration verdeutlichen und ist deshalb in verkürzter Form dargestellt.

```
      <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.

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.

4.3.3.1 Kategorie "general"

Der Schlüssel `messagedigest.algorithm` legt fest, mit welchem Algorithmus die Prüfsumme des Dokuments, welches der Benutzer hochlädt, berechnet wird. Diese Prüfsumme wird in der Graphischen Benutzeroberfläche angezeigt, beim Webservice in die Antwort eingebunden und landet auch im Anwendungslog (Defaultwert: `SHA-1`).

Der Schlüssel <`detached.signature.enabled`> aktiviert bzw. deaktiviert die Unterstützung für „Detached"-Signaturen. Der Schlüssel <form.validation.enabled> aktiviert bzw. deaktiviert die Formenvalidierung. Derzeit werden XAdES, CAdES, PAdES und ASiC Signaturen in den Profilen B, T, LT, LTA unterstützt.

Der Schlüssel `form.validation.enabled` bestimmt, ob die Funktion „extended Validation" bei der Prüfung über MOA SPSS aktiviert ist (Defaultwert: `true`).

4.3.3.2 Kategorie "pdf-as"

Diese Kategorie umfasst die Bezeichnung des Konfigurationsverzeichnisses (`<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 Konfigurationsverzeichnisses 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`").

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.

4.3.3.3 Kategorie "adobe"

In diese Kategorie können die unterstützten Filter (`<supported.filters>`), bzw. die zu ignorierenden Filter (`<ignored.filters>`), Subfilter (`<ignored.subfilters>`) und Subfilterpräfixe (`<ignored.subfilter.prefixes>`) für PDF Signaturen angegeben werden. Es kann auch die Unterstützung für alle Filter aktiviert werden (`<support.all.filters>`), dann müssen die unterstützten Filter nicht mehr einzeln angeben werden.

Die Konfiguration im Bereich `<supported.subfilters>` sollte übernommen werden um maximale Kompatibilität sicherzustellen.

4.3.3.4 Kategorie "moa.sp"

Diese Kategorie beinhaltet alle Parameter die der Prüfdienst benötigt um mit MOA SPSS kommunizieren zu können um signierte Daten prüfen zu lassen. Der Schlüssel `<connection.url>` legt fest unter welcher URL die MOA SP Instanz erreichbar ist. Der Schlüssel `<trustprofile.id>` bestimmt das Vertrauensprofil dass verwendet wird um eine Prüfung durchzuführen. Der Schlüssel `<service.uri>` sollte mit dem Pfad in `<connection.url>` übereinstimmen.

4.3.3.5 Kategorie "moa.ss“

Diese Kategorie beinhaltet alle Parameter die der Prüfdienst benötigt um Webservice Antworten über MOA SPSS signieren zu können. Der Schlüssel `<connection.url>` legt fest unter welcher URL die MOA SPSS Instanz erreichbar ist. Der Schlüssel `<key.idendifier>` bestimmt den privaten Schlüssel mit dem Antworten signiert werden.

4.3.3.6 Kategorie "report"

Mit dem Schlüssel `pdfreport.enabled` kann man PDF Prüfberichte aktivieren oder deaktivieren (Default: `true`).

4.3.3.7 Kategorie "oid-mapping"

Diese Kategorie erlaubt OIDs auf Namen zu mappen.

4.3.3.8 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 voneinander zu trennen, z.B. `xml, birthcertificate/20060814#`. Wichtig ist dass der Wert mit den definierten Format IDs in der `formatdetection-config.xml` Datei übereinstimmen.

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

4.3.3.9 Kategorie "format-filter"

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

4.3.3.10 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.

4.3.3.11 Kategorie "officialsignature"

Diese Kategorie definiert eine Liste von Ausstellern von qualifizierten Zertifikaten.

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 8080 bereits verwenden.
Warnung: Nach einer Port-Änderung darf nicht vergessen werden, diese auch in den Konfigurationen für das Signaturprüfservice (Abschnitt 4.3.3.2 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:12380/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:12380/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 11) enthält.

moa.sign.Certificate = Signature_Verification-IAIK_Test_Intermediate_CA.cer

Seit Version 4.0 von PDF-AS muss der Speicherort zum verwendeten Zertifikat für die Signatur in der Konfiguration angebeben werden. Relative Pfadangaben beziehen sich auf den Speicherort:

%TOMCAT_DIR%\conf\signature-verification\pdf-as\

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 4.3.3.2 bzw. 4.3.1).

sig_obj.%PROFIL%.value.SIG_SUBJECT = Signaturprüfservice v1.8.0

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

sig_obj.%PROFIL%.value.SIG_META = Prüfservice: http://localhost:8080/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

%MOA-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 Austellers 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 Vertrauensanker[11] 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.

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

Referenzen

[MOA]	*MOA: Serversignatur (SS), Signaturprüfung (SP),* Version 2.0.3 https://joinup.ec.europa.eu/software/moa-idspss/home abgerufen aus dem WWW am 15.05.2015
[MOA-SPSS]	*MOA: Serversignatur (SS) und Signaturprüfung (SP) Handbücher*, Version 2.0.3 https://joinup.ec.europa.eu/asset/moa-idspss/asset_release/moa-spss abgerufen aus dem WWW am 15.05.2015
[OID]	Bundeskanzleramt Österreich; Arno Hollosi, Herbert Leitold, Thomas Rössler, Robert Wollendorfer *Object Identifier der öffentlichen Verwaltung*, Version 1.0.7
[PDF-AS]	Wilfried Lackner, Wolfgang Prinz: *PDF-AS Amtssignatur für elektronische Aktenführung*, 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: *The Apache Tomcat 5.5 Servlet/JSP Container* http://tomcat.apache.org/tomcat-5.5-doc/index.html abgerufen aus dem WWW am 14.02.2007

Historie

Version	Datum	Kommentar
0.8	14.02.2007	erster Entwurf
Ersteller		
Thomas Knall		

Version	Datum	Kommentar
0.9	14.02.2007	Durchsicht
Ersteller		
Herbert Leitold		

Version	Datum	Kommentar
1.0	15.02.2007	minimale Änderungen, Abschnitt bzgl. "Einrichtung als Windows-Service" hinzugefügt
Ersteller		
Thomas Knall		

Version	Datum	Kommentar
1.1	3.12.2007	Anpassungen an die Version 1.2.5 der Anwendung.
Ersteller		
Thomas Knall		

Version	Datum	Kommentar
1.2	19.12.2007	Anmerkungen zur Konfiguration über System-Properties sowie Ergänzung der Rubrik "pdf-as" bzgl. Flag zum Ein/Ausschalten der Signatur des Prüfprotokolls.
Ersteller		

Version	Datum	Kommentar
1.3	24.01.2008	• Konfigurationsschlüssel "`<providesignaturedata.url>`" entfernt. • Neuen Schlüssel `<info.url>` eingeführt. • Neuen Schlüssel `<signeddata.source>` eingeführt.
Ersteller		
Thomas Knall		

Version	Datum	Kommentar
1.4	06.03.2008	• Neuen Schlüssel `<signature.mode>` eingeführt (Wahl des Signatur-Modus `textual` oder `binary`). • Hinweis zu Inkrementellen Updates hinzugefügt. • Hinweis zur Anzeige der Signaturdaten hinzugefügt.
Ersteller		
Thomas Knall		

Version	Datum	Kommentar
1.5	20.06.2008	Neue Kategorie `logging` mit Konfigurationsschlüssel `<svlogger.impl>` eingeführt (Wahl eines Loggers für das Prüfergebnis).
Ersteller		
Thomas Knall		

Version	Datum	Kommentar
1.6	25.06.2008	Abschnitt mit Hinweisen zum Deployment eingefügt (Abschnitt 4)
Ersteller		
Thomas Knall		

Version	Datum	Kommentar
1.7	15.07.2013	Anpassungen an die Version 1.7 der Anwendung
Ersteller		
Alexander Marsalek		

Version	Datum	Kommentar
1.8	19.05.2015	Anpassungen an die Version 1.8 der Anwendung
Ersteller		
Alexander Marsalek		
Version	**Datum**	**Kommentar**
2.0.0	30.06.2016	Anpassungen an die Version 2.0.0 der Anwendung
Ersteller		
Alexander Marsalek Dominik Ziegler		
Version	**Datum**	**Kommentar**
2.0.1	11.04.2017	Anpassungen an die Version 2.0.1 der Anwendung
Ersteller		
Alexander Marsalek Dominik Ziegler		
Version	**Datum**	**Kommentar**
2.0.3	15.06.2018	Anpassungen an die Version 2.0.3 der Anwendung
Ersteller		
Alexander Marsalek Dominik Ziegler Peter Aufner		
Version	**Datum**	**Kommentar**
2.0.5	06.05.2021	Anpassungen an die Version 2.0.5 der Anwendung
Ersteller		
Gerald Palfinger Alexander Marsalek		
Version	**Datum**	**Kommentar**
2.0.5.1	13.01.2022	Anpassungen an die Version 2.0.5.1 der Anwendung
Ersteller		
Gerald Palfinger		
Version	**Datum**	**Kommentar**
2.1.1	20.01.2023	• Links aktualisiert • Minimalanforderungen erneuert • Abschnitt 3: Screenshots aus der überarbeiteten Oberfläche von Version 2.1.0 eingepflegt • Abschnitt 4: Konfiguration aktualisiert, veraltete Schlüssel entfernt, fehlende Schlüssel beschrieben.
Ersteller		
Christof Rabensteiner		
Version	**Datum**	Anpassungen an die Version 2.2.0 der Anwendung • Erweiterte PDF Prüfung hinzugefügt • Gesonderte visuelle Auszeichnung von qualifizierten Zertifikaten hinzugefügt
2.2.0	27.02.2025	
Ersteller		
Thomas Lenz		