

Dokumentation Signaturprüfung

Signaturen im E-Government

Version 1.0, 06. Juni 2006

Klaus Stranacher – kstranacher@iaik.tugraz.at

Zusammenfassung: Aus den bestehenden Verfahren mit elektronisch signierter Erledigung oder signierter Empfangsbestätigungen (Bescheide, Zustellung, eRechnung, EPS2, etc.) ergibt sich immer wieder die Notwendigkeit, Verfahren zur Signaturprüfung vorzusehen. Dieses Dokument beschreibt zwei Varianten der Signaturprüfung: Signaturprüfung mittels Prüfservice und Signaturprüfung mittels Integration eines Prüfbuttons in ein HTML-Formular.

Es wird hier die Anwendung dokumentiert. Es werden dabei sowohl die Benutzung (Anwendungsbeschreibung), die getesteten Umgebungen (Testbeschreibung), sowie die Installation beschrieben (Deployment und Auslieferung).

Inhaltsverzeichnis:

Abbildungsverzeichnis.....	2
Revision History	3
1 Kurzbeschreibung	4
1.1 Grundlagen	4
1.2 Funktionsbeschreibung	4
1.3 Voraussetzungen zur Nutzung der Anwendung	8
2 Anwendungsbeschreibung	9
2.1 Signaturprüfung mittels Prüfservice	9
2.2 Signaturprüfung mittels Prüfbutton	12
3 Testbeschreibung.....	13
3.1 Signaturprüfung mittels Prüfservice	13
3.2 Signaturprüfung mittels Prüfbutton	14
4 Deployment	15
4.1 Systemanforderungen	15
4.2 Installation	15
4.3 Konfiguration	16
5 Auslieferung	18
5.1 Struktur	18

Anmerkung: Zur besseren Lesbarkeit wurde in diesem Dokument teilweise auf geschlechtsspezifische Formulierungen verzichtet. Die verwendeten Formulierungen richten sich jedoch ausdrücklich an beide Geschlechter.

Abbildungsverzeichnis

Abbildung 1.1: Grundlegende Funktion des Online-Prüfservice.	5
Abbildung 1.2: Grundlegende Funktion der Signaturprüfung mittels Prüfbutton.	7
Abbildung 1.3: Ausschnitt aus dem XML-Schema des elektronischen Bescheids bzw. XMLDSIG.	8
Abbildung 2.1: Screenshot des Web-Browser beim Upload der Dokumente.	10
Abbildung 2.2: Die Anzeige des Prüfergebnisses im Web-Browser.	11
Abbildung 2.3: Anzeige des signierten Bescheids und Prüfbuttons im Web-Browser.	12

Revision History

Version	Datum	Autor(en)	
0.1	31.05.2006	Klaus Stranacher	Dokumenterstellung
0.2	01.06.2006	Klaus Stranacher	Erweiterungen
1.0	06.06.2006	Klaus Stranacher Thomas Rössler	Überarbeitungen

1 Kurzbeschreibung

1.1 Grundlagen

Elektronische Signaturen spielen im E-Government eine wichtige Rolle. Sie werden in bestehenden Verfahren zur Wahrung der Authentizität, der Datenintegrität und zur eindeutigen Identifikation von Personen sowie Behörden eingesetzt. So werden beispielsweise ausgehende Bescheide von der Behörde zuvor signiert. Aus diesem Grund ergibt sich immer wieder die Notwendigkeit, Verfahren zur Signaturprüfung vorzusehen. Dabei kann es sich sowohl um automatische Prüfverfahren, die mit MOA-SP abgedeckt sind, aber auch um manuelle Prüfung durch den Bürger oder Sachbearbeiter handeln. Dieses Dokument zeigt zwei grundsätzliche Möglichkeiten der Signaturprüfung auf. Die erste Möglichkeit befasst sich mit dem Upload von signierten Dokumenten auf Prüfservices und die zweite Variante beschäftigt sich mit der Integration eines Prüfbuttons in HTML-Formularen.

1.2 Funktionsbeschreibung

Die Funktionsbeschreibung gliedert sich in zwei Teile. Der erste Teil beschäftigt sich mit dem Upload von signierten Dokumenten auf ein Online-Prüfservice. Im zweiten Teil wird die Integration eines Prüfbuttons in ein HTML-Formular beschrieben.

1.2.1 Signaturprüfung mittels Prüfservice

Das Ziel dieser Anwendung ist es Benutzern ein Web-Service zur Verfügung zu stellen, mit dem es möglich ist Signaturen zu überprüfen. Hierzu wird dem Benutzer eine Website angezeigt, die es ihm ermöglicht das signierte Dokument zu uploaden und anschließend verifizieren zu lassen. Zum Einsatz kommen an dieser Stelle Java-Servlets und Java-Server-Pages. Das prinzipielle Funktionsprinzip lautet dabei (siehe auch Abbildung 1.1):

Benutzerseite

1. Der Benutzer öffnet die URL-Adresse des Prüfservice in seinem Web-Browser.
2. Anschließend kann das zu verifizierende XML-Dokument angegeben werden.
3. Optional können benötigte Ergänzungsobjekte angegeben werden (siehe Abbildung 1.1).

Prüfserviceseite

4. Die Dokumente werden mittels des *FileUpload-Servlet* an das Prüfservice übergeben.

Benutzerseite

5. Der Benutzer kann in einem nächsten Schritt zwischen der Signaturprüfung mittels Bürgerkartenumgebung (BKU) und der Signaturprüfung mittels des Moduls MOA-SP auswählen.

Prüfserviceseite

6. Je nach Auswahl wird eines der beiden Servlets *VerifyMOA-Servlet* oder *VerifyBKU-Servlet* aktiviert.
7. Das jeweilige Servlet führt anschließend die Signaturprüfung durch und übermittelt das Ergebnis der Prüfung an den Web-Browser des Benutzers.

Benutzerseite

8. Der Benutzer kann das Ergebnis der Signaturprüfung in seinem Web-Browser betrachten.

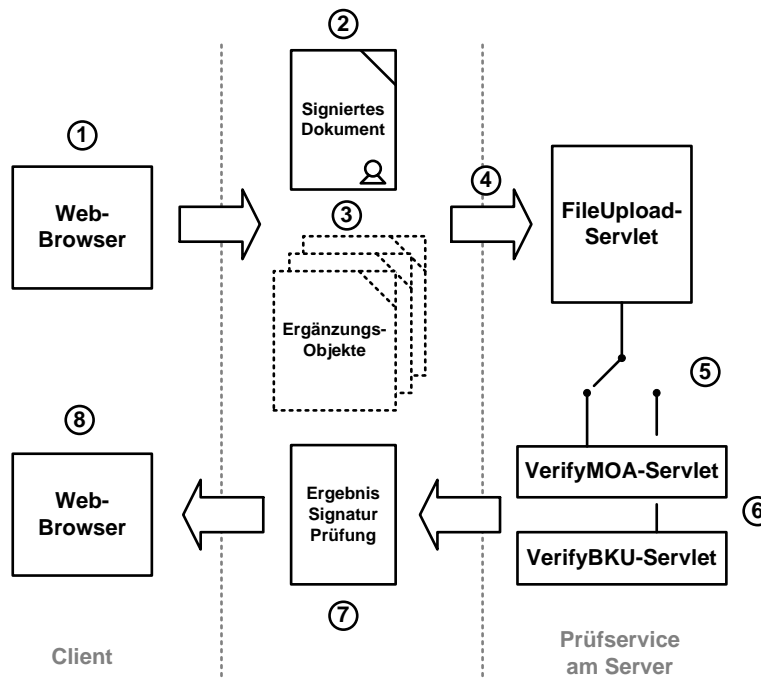


Abbildung 1.1: Grundlegende Funktion des Online-Prüfservice.

1.2.2 Signaturprüfung mittels Prüfbutton

1.2.2.1 Das Ziel

Das Ziel besteht darin einen Prüfbutton in ein signiertes Dokument zu integrieren, der eine einfache Überprüfung der enthaltenen Signatur ermöglicht. Nach Aktivierung des Prüfbuttons soll hierzu eine Signaturprüfung mittels der lokalen Bürgerkartenumgebung gestartet werden. Hierzu muss ein *VerifyXMLSignatureRequest* für das signierte Dokument generiert werden, das das signierte Dokument enthält. Das Problem besteht nun darin, aus dem signierten Dokument mittels eines Stylesheet diesen Request zu erzeugen. Hierzu bieten sich grundsätzlich folgenden zwei Varianten an.

Variante 1

In dieser Variante wird mittels des Stylesheet der Bescheid direkt XML-codiert in den *VerifyXMLSignatureRequest* transformiert. Hierzu stellt XSL mit den Befehlen *copy* bzw. *copy-of* die entsprechenden Mittel zur Verfügung. Es ergeben sich bei dieser Variante jedoch eine Reihe von Schwierigkeiten, die einen praktischen Einsatz verhindern. So muss für eine erfolgreiche Verifikation das Dokument bitgenau transformiert werden. Dieser Umstand bereitet Probleme, da vor allem ein richtiges Transformieren der Whitespaces nicht zu hundert Prozent garantiert werden kann. Des Weiteren werden beim Kopieren des Dokumentes die Namespace-Deklarationen umgeschichtet. Dieser Umstand würde zwar eine erfolgreiche Verifikation nicht verhindern, jedoch wird dadurch ein Einlesen des Dokumentes durch die Bürgerkartenumgebung verhindert¹. Aufgrund dieser Tatsachen ist diese Variante der Request-Erzeugung nicht zu empfehlen.

¹ Es tritt ein „Allgemeiner Fehler beim Einlesen des Dokumentes“ auf, da Namespace-Deklarationen nach der Umschichtung von Namespaces offenbar nicht mehr gefunden werden können.

Variante 2

Bei dieser Variante wird das signierte Dokument in einen Base64-Wert codiert. Dieser Wert wird anschließend dem Dokument hinzugefügt. Das Hinzufügen muss dabei so erfolgen, dass die Signatur nicht gebrochen wird. Durch den entsprechenden Stylesheet kann danach der *VerifyXMLSignatureRequest* zusammengestellt werden. Der Vorteil dieser Variante besteht darin, dass durch die Base64-Transformation sowohl die Namespace-Deklarationen als auch die Whitespaces erhalten bleiben. Der Nachteil ist, dass das signierte Dokument nochmals Base64-codiert ins Dokument eingefügt werden muss. Trotz dieses Nachteils ist diese Variante zu empfehlen.

1.2.2.2 Grundlegende Funktion

Abbildung 1.2 zeigt einen Überblick über die prinzipielle Funktionsweise.

Von Seiten der Behörde müssen folgenden Funktionen erfüllt werden:

1. Die Behörde erstellt Bescheid und signiert diesen.
2. Anschließend wird der signierte Bescheid in einen Base64-Wert codiert.
3. Dieser Base64-Wert wird daraufhin dem Bescheid hinzugefügt. Details hierzu siehe Abschnitt 1.2.2.3.
4. Des Weiteren wird in dem Bescheid zu Beginn ein Stylesheet als XSL-Instruction angegeben, der den Bescheid bei der Anzeige im Web-Browser entsprechend transformiert und anzeigt.
5. Abschließend übermittelt die Behörde den Bescheid an den Bürger.

Auf Bürgerseite geschieht folgendes:

- A. Der Bürger öffnet den übermittelten Bescheid in seinem Web-Browser.
- B. Daraufhin wird mit Hilfe des im Header angegebenen Stylesheet der Bescheid in ein HTML-Dokument transformiert, welches anschließend angezeigt wird. Angezeigt werden dabei der Akteninhalt, die Signatur und der Prüfbutton mit dem die Signaturprüfung gestartet werden kann. Der Prüfbutton ist dabei in ein HTML-Formular eingebettet, welches eine hidden-field enthält. In diesem Feld wird mittels des Stylesheet und dem Base64-Wert aus dem Bescheid der *VerifyXMLSignatureRequest* an die Bürgerkartenumgebung zusammengesetzt.
- C. Klickt der Bürger auf den Prüfbutton, so wird der *VerifyXMLSignatureRequest* an die Bürgerkartenumgebung übermittelt.
- D. Diese überprüft die angegebene Signatur und zeigt abschließend das Prüfergebnis dem Bürger an.

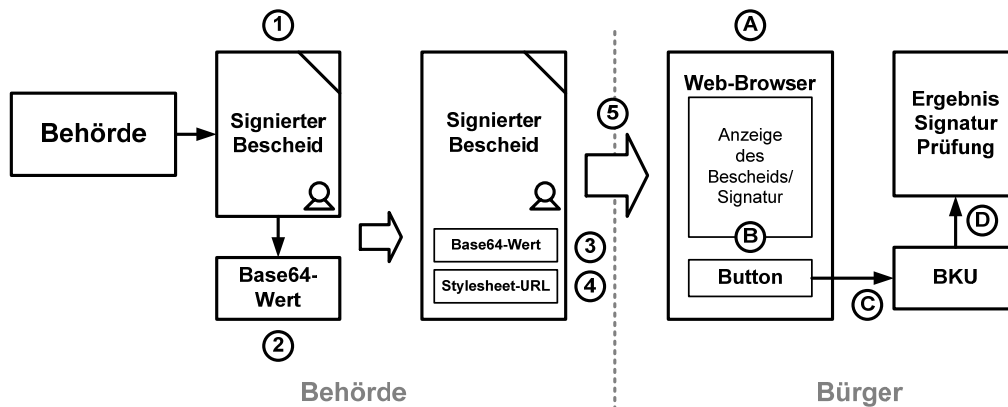


Abbildung 1.2: Grundlegende Funktion der Signaturprüfung mittels Prüfbutton.

1.2.2.3 Details zum Hinzufügen des Base64-Wertes

Beim Hinzufügen des Base64-Wertes in das signierte Dokument besteht die Anforderung, dass dadurch die Signatur nicht gebrochen werden darf. Dabei gibt es wiederum zwei Ansätze, die im Folgenden kurz aufgezeigt werden.

Ansatz 1

In diesem Ansatz wird der Base64-Wert direkt in die jeweilige Struktur des zu signierenden Dokuments eingefügt. Am Beispiel eines elektronischen Bescheids wird eine Element *Base64Content* als letztes Kindelement des Wurzelements *Bescheid* eingefügt (siehe Abbildung 1.3). Die entsprechenden Schemen im E-Government sind dabei so spezifiziert, dass Erweiterungen einfach möglich sind. Dies wird durch das *any*-Element erreicht. Um die Signatur nicht zu brechen muss bereits bei der Signaturerzeugung eine entsprechende XPath-Transformation angegeben werden, die das Element *Base64Content* von der Signatur ausnimmt. Der Nachteil dieser Variante besteht darin, dass man für den produktiven Einsatz das jeweilige Schema um das Element *Base64Content* erweitern muss.

Ansatz 2

Im zweiten Ansatz wird innerhalb der vorhandenen Signatur ein weiteres *dsig:Object* erzeugt, welchem ein Element *Base64Content* hinzugefügt wird (siehe Abbildung 1.3). Da nur jene *dsig:Object* Elemente mitsigniert werden, die auch entsprechend referenziert sind, bricht die Signatur nicht. In diesem Fall muss das Schema des jeweiligen Dokuments nicht erweitert werden. Aus diesem Grund ist dieser Variante der Vorzug zu geben.

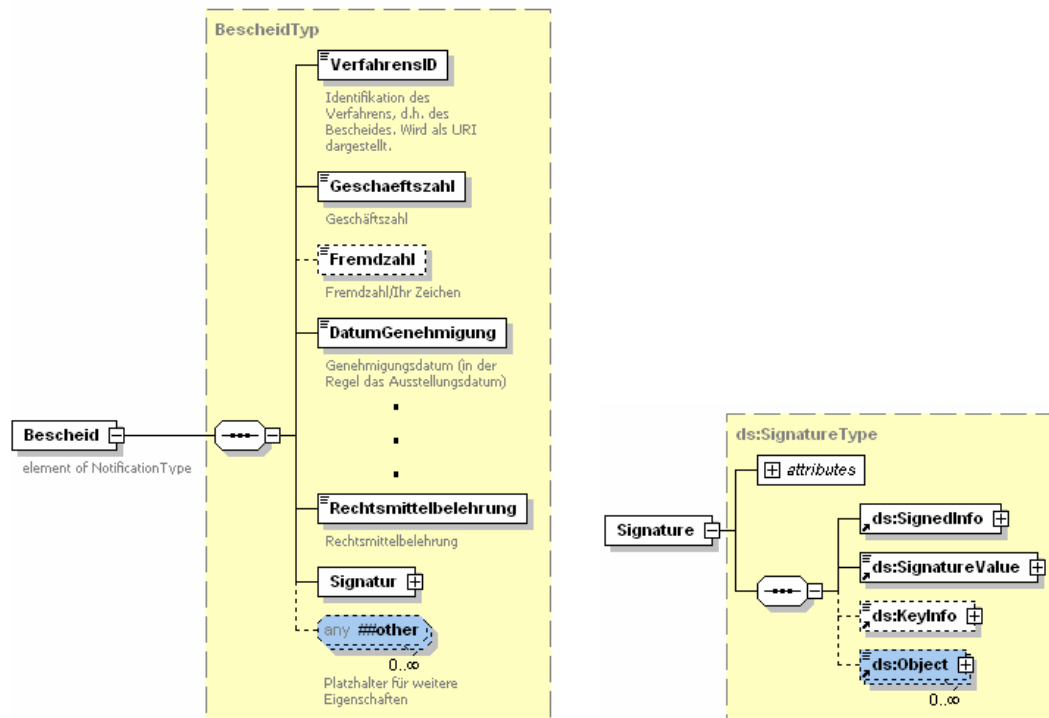


Abbildung 1.3: Ausschnitt aus dem XML-Schema des elektronischen Bescheides bzw. XMLDSIG.

1.3 Voraussetzungen zur Nutzung der Anwendung

- Ein Web-Browser
- Eine lokal installierte Bürgerkartenumgebung, die die Security-Layer Spezifikation v1.2 unterstützt MOA-SP.
- J2SE ab 5.0 SDK/JRE
- Installation des mitgelieferten vorkonfigurierten Web-Servers.

2 Anwendungsbeschreibung

2.1 Signaturprüfung mittels Prüfservice

Der nachfolgenden Anwendungsbeschreibung wird folgende Terminologie bzw. werden folgende Parameter zu Grunde gelegt:

Parameter

Name	Wertebereich	Beschreibung
%CONTAINER%	n/a	Bezeichnet den verwendeten Servlet-Container.
%CONTEXT%	n/a	Bezeichnet den Servlet-Kontext, der dem Service zugeordnet wurde.

Der Anwender startet das Prüfservice *ValidateSignature* über den Aufruf der URL:

```
%CONTAINER%/%CONTEXT%/
z.B. http://localhost:8080/ValidateSignature/
```

Der Anwender erhält nach Aufruf dieser URL, die Möglichkeit ein signiertes XML-Dokument und allfällige Ergänzungsobjekte (wie Bilder oder Stylesheets) an das Prüfservice zu übermitteln.

2.1.1 Upload der Dokumente

Abbildung 2.1 zeigt hierbei einen Screenshot des Web-Browsers. In dieser Abbildung ist ersichtlich, dass ein signiertes XML-Dokument und zwei Ergänzungsobjekte auf das Prüfservice hochgeladen wurden. Hierbei wird bei dem Upload eines Ergänzungsobjektes die Möglichkeit geboten einen separaten Referenznamen anzugeben. Wird kein Name angegeben, so wird der Dateinamen als Referenzname herangezogen. Damit ist es möglich den Referenznamen an den in der Signatur angegebenen Namen anzupassen. Beispielsweise wird in der Abbildung das Ergänzungsobjekt *xmldocument.xsd* in der Signatur mit *urn:XMLDocument.xsd* angegeben.

2.1.2 Auswahl der Prüfmethode

Bei der Auswahl der Prüfmethode werden zwei Möglichkeiten angeboten.

- Signaturprüfung mittels der Bürgerkartenumgebung (BKU)
- Signaturprüfung mittels des Moduls MOA-SP

Durch Auswahl einer der beiden Varianten, wird die Signaturprüfung gestartet und anschließend das Prüfergebnis angezeigt.

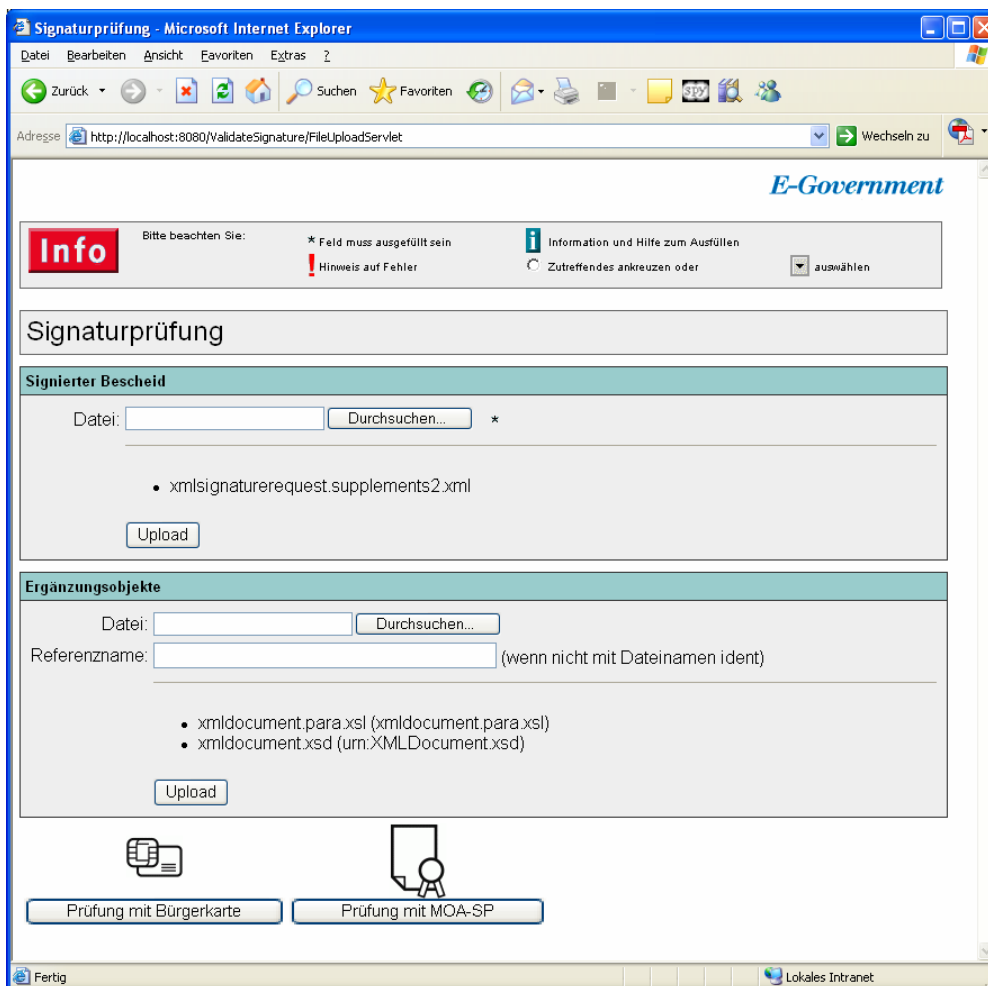


Abbildung 2.1: Screenshot des Web-Browsers beim Upload der Dokumente.

2.1.3 Anzeige des Prüfergebnisses

Das Prüfergebnis enthält dabei folgende Daten.

- *Angaben zum Unterzeichner bzw. Aussteller*
 - *Name*: Entspricht CN (CommonName)
 - *Organisationseinheit*: Entspricht OU (OrganisationUnit)
 - *Organisation*: Entspricht O (Organisation)
 - *Staat*: Entspricht C (Country)
- *Informationen zum Zertifikat*
 - *Seriennummer des Zertifikats*
- *Prüfungen*
 - *Signatur*: Dieses Feld gibt das Ergebnis der Signaturprüfung an.
 - *Signaturmanifest*: Zeigt das Prüfergebnis eines potentiell vorhandenen Signaturmanifest gemäß der Security-Layer Spezifikation an.
 - *XMLDSIG-Manifest*: Beinhaltet die Signatur ein Manifest gemäß der XMLDSIG-Spezifikation, so wird an dieser Stelle das Prüfergebnis des Manifests angezeigt.
 - *Zertifikat*: Gibt das Ergebnis der Zertifikatsprüfung an.

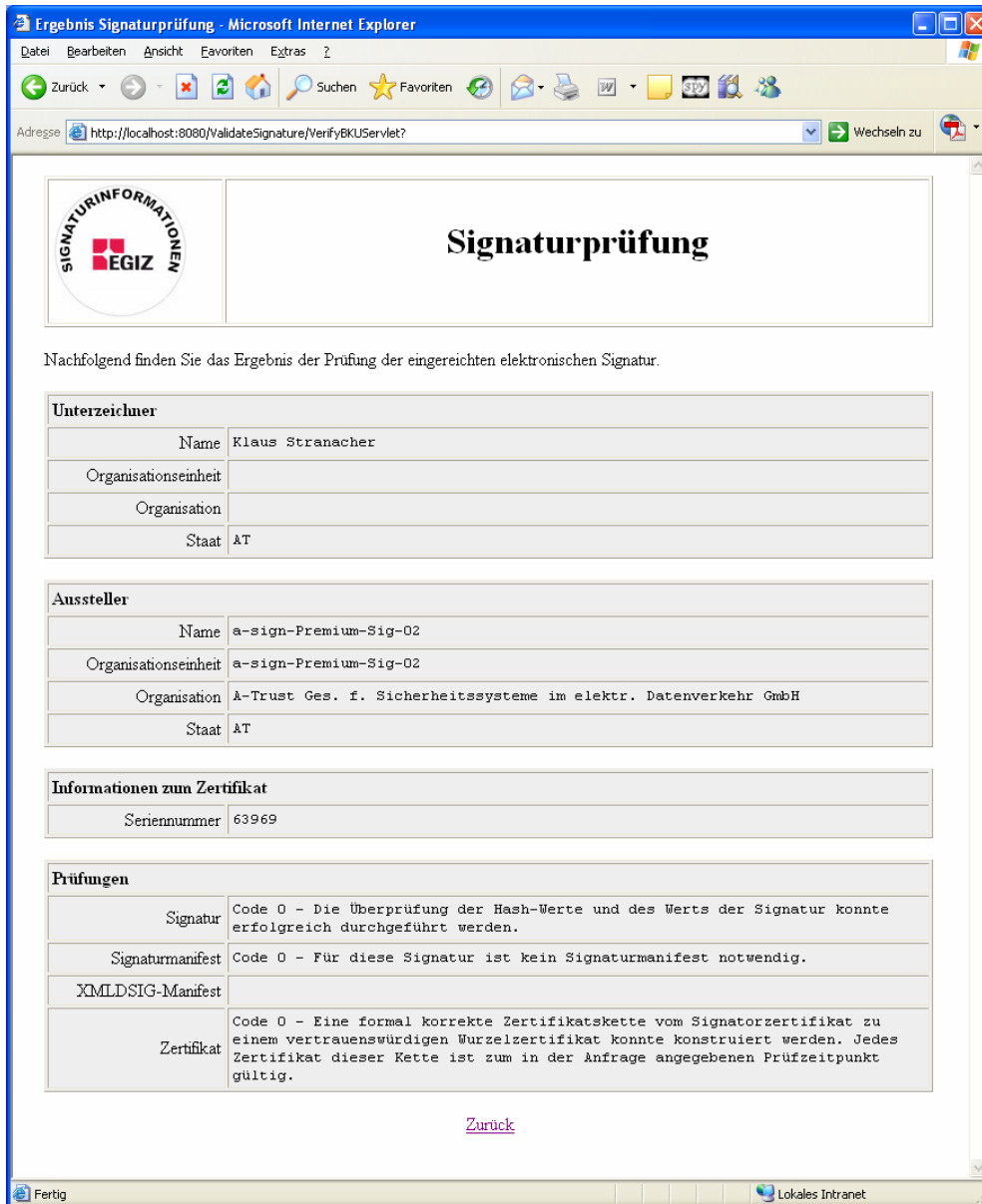


Abbildung 2.2: Die Anzeige des Prüfergebnisses im Web-Browser.

2.2 Signaturprüfung mittels Prüfbutton

Die Signaturprüfung mittels Prüfbutton stellt keine Anwendung im eigentlichen Sinn dar. Der in der Funktionsbeschreibung dargestellte Mechanismus wird mit dem angegebenen Beispiel lediglich verdeutlicht. Abbildung 2.3 zeigt hierbei einen Bescheid, der in einem Web-Browser geöffnet wurde. Neben der Darstellung des eigentlichen Bescheides werden auch die Signatur und der Prüfbutton angezeigt. Wird der Prüfbutton aktiviert, so erfolgt die Überprüfung der Signatur mittels der lokalen Bürgerkartenumgebung, welche das Ergebnis der Prüfung anschließend entsprechend anzeigt.

The screenshot shows a web browser window titled 'Musterbescheid'. The address bar contains the file path: C:\daten\signaturprüfung\pruefbutton\ok\05_BKU_Signed_DemoBescheid-1_Final.xml. The main content area displays the following sections:

Rechtsgrundlage
Kommissionsgebühren nach §§ 76 und 77 AVG in Verbindung mit § 1 lit. a der Landes-Kommissionsgebührenverordnung LGBl.Nr. 71/1990 i.d.g.F.
, für die Augenscheinsverhandlung am 02.05.2002 an der Organe des Amtes der Burgenländischen Landesregierung angefangene halbe Stunden teilgenommen haben:
Betrag: 3.14159 EUR

Begründung
Bei einer am 02.05.2002 durchgeführten Augenscheinsverhandlung wurde der in der Betriebsbeschreibung beschriebene Sachverhalt festgestellt. Gegen die Erteilung der angestrebten Bewilligung bestehen vom Standpunkt der von der Behörde wahrzunehmenden öffentlichen Interessen bei plan- und befundgemäßer Ausführung sowie bei Einhaltung der im Spruch angeführten Auflagen keine Bedenken, da bei Einhaltung der Auflagen zu erwarten ist, dass eine Gefährdung im Sinne des § 74 Abs. 2 Z 1 ausgeschlossen ist und Belästigungen, Beeinträchtigungen oder nachteilige Einwirkungen im Sinne des § 74 Abs. 2 Z 2 bis 5 GewO 1994 auf ein zumutbares Maß beschränkt werden.

Rechtsmittelbelehrung
Gegen diesen Bescheid kann binnen zwei Wochen ab der Zustellung beim Amt der Burgenländischen Landesregierung schriftlich, telegraphisch oder mittels Telekopie sowie im Wege automationsunterstützter Datenübertragung Berufung eingebracht werden. Die Berufung hat den Bescheid zu bezeichnen, gegen den sie sich richtet (bitte erlassende Behörde und Bescheidzahl angeben), und einen begründeten Berufungsantrag zu enthalten. Für die Berufung des Antragstellers (Anlageninhabers) ist nach Zustellung der Entscheidung über die Berufung eine Gebühr von 13 Euro (178,88 Schilling), für Beilagen je 3,60 Euro (49,54 Schilling) pro Bogen, maximal aber 21,80 Euro (299,97 Schilling) pro Beilage zu entrichten. Berufungen sonstiger Parteien sind gebührenfrei.

Hinweis
Die Genehmigung der Betriebsanlage erlischt, wenn der Betrieb der Anlage nicht binnen fünf Jahren nach erteilter Genehmigung in zumindest einem für die Erfüllung des Anlagenzweckes wesentlichen Teil der Anlage aufgenommen, oder durch mehr als fünf Jahre in allen für die Erfüllung des Anlagenzweckes wesentlichen Teile der Anlage unterbrochen wird. Diese Frist kann äußerstenfalls bis auf sieben Jahre verlängert werden; ein diesbezüglicher Antrag muss vor Ablauf der fünfjährigen Frist eingebracht werden.

	Verfahren	1234567890
	Datum	2006-05-31T13:36:23Z
	Aussteller	CN=a-sign-Premium-Sig-02,OU=a-sign-Premium-Sig-02,O=A-Trust Ges. f. Sicherheitssysteme im elektr. Datenverkehr GmbH,C=AT
	Seriennummer	63969
	Signaturwert	Lq0Jb4St9EuKMXG8U6Wa81p/iK+E041hZZLjI+eIE2iEoVRHOT3DqHERbg1T fnD+

Below the table is a button labeled 'Signatur prüfen'.

Abbildung 2.3: Anzeige des signierten Bescheids und Prüfbuttons im Web-Browser.

3.2 Signaturprüfung mittels Prüfbutton

Das Beispiel zur Signaturprüfung mittels Prüfbutton befindet sich im Ordner EXAMPLES/BUTTON (siehe 5.1). Folgende Dateien sind darin enthalten:

- *01_BKU_CreateXMLSignatureRequest_DemoBescheid-1.xml*
CreateXMLSignatureRequest an die BKU mit dem der Bescheid signiert wird.
- *02_BKU_CreateXMLSignatureResponse_DemoBescheid-1.xml*
CreateXMLSignatureResponse der BKU von obigem Request.
- *03_BKU_Signed_DemoBescheid-1.xml*
Der signierte Bescheid ohne dem noch hinzuzufügenden Base64-Wert.
- *04_BKU_Base64_Content_Signed_DemoBescheid-1.xml*
Der signierte Bescheid Base64-codiert.
- *05_BKU_Signed_DemoBescheid-1_Final.xml*
Hier wurden dem Bescheid der Base64-Wert als weiteres dsig:Object hinzugefügt und es wurde ein Stylesheet angegeben.
- *b2html.xsl*
Der Stylesheet zur Anzeige des Bescheids, der Signatur und des Prüfbuttons im Web-Browser
- *06_BKU_Signed_DemoBescheid-1_WebBrowser.html*
Die HTML-Datei wie sie vom Stylesheet erzeugt und im Web-Browser angezeigt wird. Darin ist ein hidden-field namens XMLRequest ersichtlich, dass den gesamten VerifyXMLSignatureRequest (entsprechend gequotet) an die BKU enthält.

4 Deployment

4.1 Systemanforderungen

Es gelten folgende generellen Anforderungen an die Installationsplattform bzw. an die Client-Komponenten (die angegebenen Versionen entsprechen der getesteten Umgebung). Da jedoch die gesamte Server-Komponente als vorkonfigurierter Web-Server ausgeliefert wird, sind ein Teil dieser Anforderungen bereits implizit erfüllt. Über die Installation dieser vorkonfigurierten Serverkomponente gibt Abschnitt 4.2 Auskunft.

4.1.1 Server-Komponenten

Auf der Serverseite kommen Java-Servlets, Java-Server-Pages und das Modul MOA-SP zum Einsatz. Aus diesem Grund gelten folgenden Anforderungen:

- Modul für Online Applikationen (MOA) zur Signaturprüfung – MOA-SP v1.3
- Apache Tomcat 4.1.31
- J2SE ab 5.0 SDK

Für die Signaturprüfung mittels Prüfbutton wird keine Server-Komponente benötigt und daher sind die Anforderungen in diesem Fall nicht relevant.

4.1.2 Client-Komponenten

Für die Client-Seite gelten folgende Anforderungen:

- J2SE ab 5.0 SDK/JRE
- Eine lokal installierte Bürgerkartenumgebung, die die Security-Layer Spezifikation v1.2 unterstützt. Die frei zur Verfügung stehende BKU Software des Bundes kann von folgender Seite bezogen werden: <http://www.buergerkarte.at>

4.2 Installation

Nachfolgend wird die Installation der vorkonfigurierten Server-Komponente (beinhaltet MOA-SP und Online-Prüfservice) beschrieben. Für die Clientseite ist abseits der in den Anforderungen angegebenen Software keine weitere Installation nötig. Die Installationsanleitung des Prüfservice auf einen bereits vorhandenen Web-Server wird in Abschnitt 4.2.2 beschrieben.

4.2.1 Installation der vorkonfigurierten Server-Komponente

4.2.1.1 Installation des vorkonfigurierten Web-Servers

Starten Sie hierzu die Datei *ValidateSignature.msi* (unter Verzeichnis *DEPLOY*, siehe 5.1). Der Windows-Installer führt Sie dann weiter durch die Installation. Nach der Fertigstellung der Installation kann der Web-Server über Start-Programme-ValidateSignature gestartet und wieder gestoppt werden.

Weitere Einstellungen oder eine Konfiguration muss nicht vorgenommen werden.

4.2.1.2 Testen der Installation

- Nach erfolgreicher Installation ist der Web-Server unter folgender Adresse erreichbar:
<http://localhost:8080>
- Die einzelnen Services stehen dabei unter folgenden Adressen zur Verfügung:
 - Prüfservice ValidateSignature:
<http://localhost:8080/ValidateSignature>
 - MOA-SP:
<http://localhost:8080/moa-spss/services/SignatureVerification>
 - MOA-SS (nicht benötigt, der Vollständigkeit halber aber angeführt):
<http://localhost:8080/moa-spss/services/SignatureCreation>

4.2.1.3 Deinstallation

Der vorkonfigurierte Web-Server kann über die Windows Systemssteuerung wieder entfernt werden.

4.2.2 Installation des Prüfservice auf einem vorhandenem Web-Server

Möchte man das Online-Prüfservice in seinem eigenen Web-Server integrieren, so muss die Datei *ValidateSignature.war* in das entsprechende Verzeichnis des Web-Servers kopiert werden (z.B. Verzeichnis *webapps* bei Tomcat). Nach einem Neustart des Servers wird diese Datei entpackt und anschließend kann das Service konfiguriert werden.

4.3 Konfiguration

4.3.1 Minimalkonfiguration

Wird das Prüfservice auf einem vorhandenen Web-Server installiert, so müssen die hier angeführten Parameter vor Inbetriebnahme der Anwendung angepasst werden. Die Einstellungen werden dabei über die Datei *web.xml* im Verzeichnis `%WEBSERVER_HOME%/webapps/ValidateSignature/WEB-INF` vorgenommen.

Property	Default-Wert	Beschreibung
SL_HOST	127.0.0.1	Host zur BKU
SL_PORT	3495	Port zur BKU
MOA_SP_SERVICENAME	SignatureVerification	Servicename von MOA-SP
MOA_SP_TRUSTPROFILEID	Test-Signaturdienste	Zu verwendende Trustprofile-ID
MOA_SP_HTTP_URL	http://localhost:8080/moa-spss/services/SignatureVerification	HTTP-URL zu MOA-SP
MOA_USE_HTTPS	false	false = Verwende HTTP true = Verwende HTTPS

4.3.2 Erweiterte Konfiguration

Wird in der Minimalkonfiguration angegeben, dass eine Verbindung zu MOA-SP über HTTPS erfolgen soll, so müssen die folgenden Einträge entsprechend angepasst werden.

Property	Default-Wert	Beschreibung
MOA_SP_HTTPS_URL	https://moa.egiz.gv.at/moa-spss/services/SignatureVerification	HTTPS-URL zu MOA-SP
MOA_TRUSTSTORE	c:/validatesignature/conf/validatesignature/moatruststore.keystore	Pfad zum MOA Truststore
MOA_TRUSTSTORETYPE	JKS	Typ des MOA Truststore
MOA_TRUSTSTOREPASSWORD	moa	Passwort des MOA Truststore

5 Auslieferung

Die vorliegende Applikation wird zusammen mit den Quelltexten und dieser Dokumentation ausgeliefert.

5.1 Struktur

/readme.txt	Kurzbeschreibung des Projekts
/DOC	Dokumentation
/SOURCE	Quelltexte zu Signaturprüfung mit Servlets
/DEPLOY	
/validatesignature.msi	Windows-Installer mit vorkonfigurierten Web-Server
/validatesignature.war	Prüfservice als WAR-Datei
/EXAMPLES	
/SERVLETS	Beispiele zu Signaturprüfung mit Servlets
/BUTTON	Beispiele zu Signaturprüfung mit Prüfbutton