

---

## Sicheres HTTP-Session-Tracking und MOA-ID Single Sign-On (SSO)

Konzept

Dipl.-Ing. Martin Centner ([martin.centner@egiz.gv.at](mailto:martin.centner@egiz.gv.at))

Dipl.-Ing. Thomas Rössler ([thomas.roessler@egiz.gv.at](mailto:thomas.roessler@egiz.gv.at))

---

### Zusammenfassung

MOA-ID [1] spezifiziert ein Servermodul zur sicheren Identifikation und Authentifikation von Benutzern mittels Bürgerkarte. Das vorliegende Konzept beschreibt die Erweiterung von MOA-ID zur Unterstützung von Client-Zertifikaten zur Authentifizierung von SSL/TLS-Sessions. Dies ermöglicht sicheres HTTP-Session-Tracking und die Implementierung eines Single Sign-On (SSO) Mechanismus.

### Inhaltsverzeichnis

1	Einleitung . . . . .	2
2	HTTP-Session-Tracking mit Client-Zertifikaten . . . . .	6
3	MOA-ID . . . . .	6
4	MOA-ID Single Sign-On mit Client-Zertifikaten . . . . .	9
5	Zusammenfassung . . . . .	10
A	MOA-ID Assertion Query/Request Profile . . . . .	12
B	Examples . . . . .	13

## 1 Einleitung

Web-Applikationen die auf dem HTTP-Protokoll (RFC 2616 [2]) aufbauen, brauchen einen Mechanismus zum Session-Tracking. Motivation und Überblick über die verschiedenen Methoden zum Session-Tracking gibt der folgende Abschnitt. Im Zusammenhang mit Authentifizierung kommt dem Session-Tracking Mechanismus auch eine sicherheitskritische Rolle zu. Das vorliegende Konzept beschreibt daher einen Session-Tracking Mechanismus, welcher auf der Authentifizierung von SSL/TLS-Session mittels Client-Zertifikaten beruht. Dieser Mechanismus kann letztendlich auch als Kernprinzip für die Implementierung eines Single Sign-On Frameworks auf Basis von MOA-ID [1] herangezogen werden, welches höchste Sicherheitsanforderungen erfüllt.

### 1.1 HTTP-Session-Tracking

Bei Web-Applikationen erfolgt die Darstellung der Benutzerschnittstelle durch den Web-Browser. Die Kommunikation zwischen dem Web-Browser und der Applikationslogik am Web-Server erfolgt über HTTP (RFC 2616 [2]). HTTP ist ein Request-Response-Protokoll, d.h. der Client (Web-Browser) schickt eine Anfrage (Request) und erhält darauf eine entsprechende Antwort (Response) vom Server. Die Antwort ist dabei immer nur von dieser einen Anfrage abhängig und damit unabhängig von allen anderen Anfragen bzw. einem inneren Zustand des Servers. Man spricht daher von einem zustandslosen (stateless) Protokoll. Applikationen besitzen jedoch in der Regel einen inneren Zustand, d.h. das Ergebnis einer Aktion ist von vorhergehenden Aktionen abhängig. Da Web-Applikationen in der Regel mehrere Instanzen mit unterschiedlichen Zuständen derselben Applikation zur Verfügung stellen, müssen die einzelnen Anfragen auch der jeweiligen Instanz zugeordnet werden können. Web-Applikationen benötigen daher ein Mechanismus, um die einzelnen Anfragen den entsprechenden Instanzen – so genannten *Sessions* – zuzuordnen. Dieser Mechanismus wird als *Session-Tracking* bezeichnet.

HTTP benutzt als Transportschicht TCP (RFC 793 [3]). HTTP 1.1. schreibt zwar vor, dass für mehrere Anfragen eines Clients die selbe TCP-Verbindung verwendet werden soll, erlaubt aber auch mehr als nur eine TCP-Verbindung pro Client zur gleichen Zeit, bzw. TCP-Verbindungen beliebig zu beenden und neue aufzubauen. Daher ist das Session-Tracking auf Basis der Zuordnung von HTTP-Anfragen zu TCP-Verbindungen ungeeignet.

Auch die unter TCP liegende Kommunikationsschicht IP (RFC 791 [4]) bietet keine ausreichende Möglichkeit, Anfragen eindeutig einem Client zuzuordnen, da sich hinter einer IP-Adresse wiederum viele Clients verbergen können. Dies ist zum Beispiel beim Einsatz von NAT (Network Address Translation), Proxies oder mehreren Clients am selben Host (Single- und Multiuser-Umgebung) der Fall.

Der Server hat damit durch HTTP keine ausreichenden Informationen zur Zuordnung von Anfragen zu einer Session zur Verfügung. Er ist auf die Unterstützung durch den Client angewiesen. Der Client muss dem Server daher mit jeder Anfrage eine Kennung übermitteln, die ihm die Zuordnung zu einer Session erlaubt. Dazu haben sich im Wesentlichen die folgenden Mechanismen etabliert:

- **Cookies** (RFC 2965 [5])  
Der Server übermittelt dem Client Informationen, die der Client dem Server mit jeder weiteren Anfrage wieder präsentiert.
- **Dynamic URL-Rewriteing**  
Die URL jeder Anfrage wird so verändert, dass sie eine eindeutige Kennung der Session enthält.

- **Hidden Form Fields**

Über versteckte Formularfelder wird eine eindeutige Kennung der Session mit jeder Anfrage als Parameter übermittelt.

## 1.2 Authentifizierung

Es gibt verschiedene Schemata HTTP-Anfragen zu authentifizieren. RFC 2617 [6] spezifiziert die beiden folgenden Authentifizierungsschemata:

- **HTTP Basic Authentication**

Zur Authentifizierung werden Benutzername und Passwort übertragen.

- **HTTP Digest Authentication**

Zur Authentifizierung wird ein kryptographischer Hashwert über die angefragte URI, das Benutzerpasswort und eine servergenerierte Nonce übertragen.

Daneben existieren auch noch weitere Authentifizierungsschemata, wie z.B. *NTLM*<sup>1</sup> (nicht dokumentiert) und *Kerberos* (dokumentiert in [7]).

Alle HTTP-Authentifizierungsschemata haben jedoch gemeinsam, dass sie einzelne HTTP-Anfragen authentifizieren. D.h. der Client muss die Authentifizierungsinformationen mit jeder Anfrage an den Server übermitteln und der Server muss diese überprüfen. Je nach Authentifizierungsschema, werden so mit jeder Anfrage sensible Daten übertragen (z.B. Benutzername und Passwort bei HTTP-Basic-Authentication), oder es sind mehrere Anfragen notwendig, um die Authentifizierung durchzuführen (z.B. Challenge-Response bei HTTP Digest Authentication).

Aus diesem Grund werden bei Web-Applikationen, die eine Authentifizierung erfordern, in der Regel nicht einzelne Anfragen, sondern Sessions authentifiziert. D.h. die Authentifizierung erfolgt einmalig für eine einzelne Anfrage, alle weiteren Anfragen, die zur gleichen Session – die auf oben diskutierte Weise verfolgt wird – gehören, werden als authentifiziert betrachtet. Dabei werden die Authentifizierungsinformationen häufig als HTTP-Parameter direkt an die Web-Applikation übermittelt. Dies erlaubt beliebige Verfahren, unabhängig von deren Unterstützung durch den Web-Server, zur Authentifizierung zu verwenden.

Wird nicht mehr jede einzelne Anfrage, sondern die Session authentifiziert, bedeutet das aber auch, dass die zum Session-Tracking verwendeten Daten genauso geschützt werden müssen, wie die Authentifizierungsdaten selbst.

## 1.3 SSL/TLS

Zum Schutz der mit HTTP übertragenen Daten wird in der Regel SSL bzw. dessen Nachfolger TLS verwendet (RFC 2818 [8]). SSL bzw. TLS bietet Nachrichten-Integrität, -Vertraulichkeit und -Authentizität auf Ebene der Transportschicht. D.h. es ermöglicht gegenseitige Authentifizierung und Verschlüsselung der übertragenen Daten. Die Authentifizierung erfolgt meist über X509-Zertifikate. Häufig authentifiziert sich auf diese Weise jedoch nur der Server gegenüber dem Client. Die Authentifizierung des Clients erfolgt meist wieder über die oben besprochenen Mechanismen zur Authentifizierung von HTTP-Sessions, was wiederum einen sicheren Mechanismus zum Session-Tracking erfordert.

---

<sup>1</sup> NTLM – Microsoft™ Windows NT® Lan Manager Protocol

SSL bzw. TLS kennen zwar im Unterschied zu HTTP das Konzept einer Session und wären damit eigentlich in der Lage, das HTTP-Session-Tracking Problem auf sichere Weise zu lösen, allerdings definiert (RFC 2818 [8]) die Verwendung von HTTP über TLS äquivalent zur Verwendung von HTTP über TCP. D.h. dem Client steht es frei, HTTP-Anfragen über eine oder mehrere SSL/TLS-Verbindungen an den Server zu schicken, bzw. diese SSL/TLS-Verbindungen beliebig zu beenden und neue aufzubauen. Da dies nicht nur theoretisch möglich ist, sondern von den gängigen HTTP-Browsern auch so umgesetzt wird, lässt sich HTTP-Session-Tracking auf Basis der SSL/TLS-Session nicht zuverlässig verwenden.

## 1.4 Session-ID

Da, wie oben diskutiert, die Zuordnung von HTTP-Anfragen zu Sessions auf vom Client übermittelten Informationen beruht und Session-Tracking bei Authentifizierung von HTTP-Sessions eine sicherheitskritische Rolle zukommt, müssen diese Informationen geeignet geschützt werden. Da keine weiteren Mechanismen zum Schutz dieser Informationen existieren, beruht der Schutz alleine auf der Geheimhaltung der zur Zuordnung verwendeten eindeutigen Session-Kennung (Session-ID). Jeder der in Kenntnis dieser eindeutigen Session-ID ist, hat die Möglichkeit seine Anfragen so zu kennzeichnen, als würden sie zu der entsprechenden Session gehören. Ist diese Session authentifiziert, erhält ein Angreifer damit die Möglichkeit, ohne Kenntnis der Authentifizierungsinformationen Zugang zu Informationen und Funktionen zu erhalten, für die eine Authentifizierung erforderlich wäre. Dies wird als *Entführen der Session* (Session Hijacking) bezeichnet.

Um ein Ausspähen der Session-ID während der Übertragung zu verhindern, muss diese verschlüsselt (wie oben diskutiert über SSL/TLS) erfolgen. Zusätzlich muss verhindert werden, dass der Angreifer die Session-ID erraten oder aus anderen ihm zur Verfügung stehenden Informationen ableiten kann. Daher wird die Session-ID in der Regel mit einem (kryptographisch sicheren) Zufallszahlengenerator erzeugt.

Ist die Session-ID vor Ausspähen bei der Übertragung und Erraten bzw. Ableiten geschützt, bleiben als Angriffspunkte die Orte, an denen die Session-ID gespeichert wird: Client und Server. Der Schutz der Session-ID am Server muss bei der Entwicklung der Server-Applikation bzw. des Sicherheitskonzepts berücksichtigt werden. Die Server-Applikation als Angriffspunkt bzw. deren Schutz wird hier nicht weiter diskutiert, da ein erfolgreicher Angriff auf den Server häufig den direkten Zugang zu den geschützten Informationen ermöglicht. Zudem muss vorausgesetzt werden, dass die Server-Applikation und der Server als solches mit entsprechender Sorgfalt entwickelt und betrieben wird, und dass aufgrund des vorauszusetzenden Sicherheitsbewusstseins bei Betreiber und Entwickler alle vertretbaren und dem Stand der Technik entsprechenden Sicherheitsmaßnahmen, sowohl technisch als auch organisatorisch, ergriffen werden. Umso kritischer ist die Situation am Client. Besonders hier ist demnach für einen entsprechenden Schutz der Session-ID zu sorgen.

Drei Verfahren zur Übermittlung der Session-ID, die im folgenden diskutiert werden, haben sich etabliert:

**Cookies** Cookies, spezifiziert in RFC 2965 [5] - *HTTP State Management Mechanism*, eignen sich – wie schon der Titel der Spezifikation vermuten lässt – am besten für die Übermittlung von Session-Informationen. Cookies ermöglichen dem HTTP-Server in seiner Antwort Informationen (das Cookie) an den Client zu übermitteln, die der Client dann mit jeder weiteren Anfrage wieder an den Server übermittelt. Der Server kann dabei angeben, wie lange das Cookie am Client gespeichert und mit welchen Anfragen es übermittelt werden soll.

Werden Cookies zur Übermittlung der Session-ID eingesetzt, ist es sinnvoll, so genannte temporäre Cookies zu verwenden. Bei temporären Cookies ist die Zeit, die das Cookie im Client gespeichert wird auf Null

gesetzt. Der Client speichert das Cookie dann nicht über eine Browser-Session hinweg. Wird der Browser geschlossen, werden alle temporären Cookies gelöscht. Weiters muss darauf geachtet werden, dass das Cookie nur an den Server übermittelt wird, von dem es erhalten wurde. Dazu ist es wichtig, dass die entsprechenden Informationen mit dem Cookie übermittelt werden.

Leider wurden und werden Cookies auch dazu verwendet, Informationen über die Surfgewohnheiten der Benutzer auszuspähen und sind daher in Verruf geraten. Manche Benutzer haben ihren Web-Browser daher so konfiguriert, dass er gar keine Cookies entgegen nimmt. Um Session-Tracking mittels Cookies zu ermöglichen, müssen aber zumindest temporäre Cookies akzeptiert werden, die dann ausschließlich an den das Cookie ausstellenden Server retourniert werden.

Da Cookies im HTTP-Header als zusätzlicher Parameter übermittelt werden, ist die in ihnen gespeicherte Information normalerweise nicht sichtbar. Um aber Session-Hijacking zu vermeiden, muss vor allem sichergestellt werden, dass nicht Dritte Zugang zum Cookie bzw. dessen Inhalt erlangen, oder dieses übermittelt bekommen.

**Dynamic URL-Rewriteing** Beim Dynamic URL-Rewriteing werden alle in der Antwort vom Web-Server enthaltenen URLs, die wieder auf den gleichen Web-Server zeigen, so modifiziert, dass sie eine Kennung der Session enthalten, also letztlich die Session-ID. Der Web-Server liest dann bei einem Request die Kennung der Session aus der URL in der Anfrage. Diese Methode der Übermittlung der Session-ID funktioniert somit auch dann, wenn Cookies vom Client nicht unterstützt oder akzeptiert werden. Allerdings ist beim URL-Rewriteing die in der URL kodierte Kennung der Session in jeder Anfrage sichtbar. Wird beispielsweise eine URL aus einer Web-Applikation Dritten zugänglich, dann ist damit auch die Kennung der Session für Dritte einsehbar. Dies kann auch aktiv durch einen unbedarften Benutzer erfolgen, wenn er beispielsweise einen Link aus einer Online-Applikation per E-Mail verschickt. Ausserdem werden URLs oft im Browser gespeichert (z.B. in der Browser-History oder als Eingabehilfe in der Adressleiste) und werden auch nach Beenden der Browser-Session nicht gelöscht. Insbesondere, wenn der Browser auch anderen Benutzern zur Verfügung steht, ist daher darauf zu achten, dass die Web-Applikation eine Logout-Funktionalität bietet, die die Zuordnung der verwendeten Session-Kennung aufhebt, da ein Schließen des Browser-Fensters nicht ausreicht.

Durch das Verändern der URLs werden aber auch Caching-Mechanismen unwirksam, da sich auch bei statischen Elementen (z.B. Grafiken) die URLs mit jeder neuen Session verändern.

**Hidden Form-Fields** Auch versteckte Formularfelder können verwendet werden, um die Session-Kennung zusammen mit den Anfragen zurück an den Server zu übermitteln. Damit alle Anfragen an den Server die Session-Kennung enthalten, müssen alle Links über HTML-Formulare realisiert werden. Daher ist diese Methode, abhängig von der Web-Applikation, oft nur eingeschränkt nutzbar. Auch hier ist darauf zu achten, dass die Session-Kennung (Session-ID) durch eine entsprechende Logout-Funktionalität serverseitig wertlos gemacht wird, da Browser oft eine Funktionalität zur Speicherung von Formulardaten bieten.

Eine große Bedrohung für die Geheimhaltung der Session-Informationen in Web-Applikationen stellen Angriffe dar, die auf so genanntem „Cross-Site Scripting“ basieren. Dabei platziert ein Angreifer HTML- oder Script-Code in einer Web-Applikation so, dass dieser beim Zugriff auf die Web-Applikation durch einen anderen Benutzer auf dessen Client zur Ausführung gelangt. Dies ist oft bei dagegen nicht ausreichend abgesicherten Web-Applikationen möglich (z.B. durch das Hinterlassen einer speziell präparierten Nachricht in einem Web-Forum oder Gästebuch). Die Gefahr besteht darin, dass der Script-Code zur Übermittlung

der Session-ID des Opfers an den Angreifer verwendet werden kann und so ein Session-Hijacking möglich macht. Dieser Bedrohung könnte durch einen Session-Tracking Mechanismus, dessen Sicherheit nicht alleine auf der Geheimhaltung einer mit jeder Anfrage zu übermittelnden Session-ID beruht, recht effizient begegnet werden.

## 2 HTTP-Session-Tracking mit Client-Zertifikaten

Je nach verwendetem Verfahren können Session-IDs – und in weiterer Folge authentifizierte Sessions – mehr oder weniger gut geschützt werden. Grundsätzlich ist jedoch ein höheres Maß an Sicherheit beim Session-Tracking nur dann zu erreichen, wenn diese nicht alleine auf der Geheimhaltung der Session-ID beruht.

Wie in Abschnitt 1.3 diskutiert, erlaubt SSL bzw. TLS die gegenseitige Authentifizierung mittels X509-Zertifikaten. Auf Basis des für die Authentifizierung des Clients verwendeten Zertifikats kann somit eine eindeutige Bindung zum Client hergestellt werden. Dabei spielt es auch keine Rolle, dass SSL/TLS-Sessions beliebig neu aufgebaut werden können, da für jede SSL/TLS-Session vom selben Client wieder das gleiche Zertifikat zur Authentifizierung zum Einsatz kommt.

Da die gegenseitige Authentifizierung mittels Zertifikaten auf einem asymmetrischen kryptographischen Verfahren beruht, hängt die Sicherheit auch nicht von der Geheimhaltung einer zu übertragenden Kennung ab. Die dabei zum Einsatz kommenden privaten Schlüssel sind üblicherweise sehr gut geschützt, z.B. unauslesbar auf einer Smartcard oder in einem Software Key-Store, und „verlassen“ den Client somit nicht. Jede HTTP-Anfrage kann so aufgrund des für die Authentifizierung der SSL/TLS-Session verwendeten Client-Zertifikats eindeutig einer Session zugeordnet werden. Session-Hijacking ist dann nur mehr bei Diebstahl des verwendeten privaten Schlüssels möglich. Sollen mehrere gleichzeitige Sessions vom selben Benutzer an einer Web-Applikation möglich sein, dann kann zusätzlich noch auf einen der oben diskutierten Session-Tracking-Mechanismen zurückgegriffen werden. Damit wäre ein Übernehmen der HTTP-Session zwar zwischen den einzelnen SSL/TLS-Sessions des selben Benutzers möglich, daraus ergibt sich jedoch keine zusätzliche Sicherheitsbedrohung. Zur Authentifizierung einer neuen SSL/TLS-Session ist immer der private Schlüssel erforderlich, unabhängig ob eine HTTP-Session-ID der Online-Applikation bereits bekannt ist oder nicht.

Das verwendete Client-Zertifikat zum Aufbau der authentifzierten SSL/TLS-Verbindung muss nicht notwendigerweise mit dem Benutzer bzw. dessen Identität in Verbindung gebracht werden. Wird das Client-Zertifikat nur zur Zuordnung einer HTTP-Anfrage zu einer Session verwendet und erfolgt die Identifikation des Benutzers auf anderem Wege, dann kann ein beliebiges Client-Zertifikat zum Einsatz kommen, solange der zugehörige private Schlüssel unter der Kontrolle des Benutzers bleibt. Damit ist es möglich beliebige Authentifizierungsmechanismen gleichzeitig mit dem auf Client-Zertifikaten basierendem Session-Tracking einzusetzen.

## 3 MOA-ID

Bei der Authentifizierung mittels Bürgerkarte kommen Zertifikate zum Einsatz. Allerdings, erfolgt die Identifikation des Bürgers nicht über die im Zertifikat enthaltenen Informationen, sondern über die so genannte

*Personenbindung*<sup>2</sup> [9]. Mit MOA-ID [1] steht ein Modul zur Authentifizierung und Identifikation mit der Bürgerkarte zur Verfügung. Der folgende Abschnitt 3.1 beschreibt die Authentifizierung mit MOA-ID und der Bürgerkarte. Abschnitt 3.2 diskutiert die Möglichkeiten zur Realisierung eines Single Sign-On Mechanismus mit MOA-ID.

### 3.1 Authentifizierung

Die Authentifikation und Identifikation eines Bürgers mit MOA-ID erfolgt über ein mit der Bürgerkarte zu signierendes Anmeldeformular, das unter anderem die Personenbindung zu dem zur Signatur verwendeten Schlüsselpaar enthält. Die Bestätigung der erfolgreichen Authentifizierung und die Weitergabe der Personenbindung an die solcherart zugängliche Web-Applikation erfolgt über das *SAML Web Browser SSO Profile* [10].

Abbildung 1 zeigt den Ablauf der Authentifizierung mit MOA-ID im Detail. Der erste Teil zeigt die Authentifizierung mittels Signatur mit der Bürgerkarte, der zweite Teil die Weitergabe der Information über die erfolgte Authentifizierung und die entsprechende Personenbindung an die Online-Applikation. MOA-ID ist in eine MOA-ID-Auth und eine MOA-ID-Proxy Komponente geteilt.

Der Benutzer erhält über einen Link oder Redirect eine Web-Seite von der MOA-ID-Auth Komponente mit einem Formular, das einen `<InfoBoxReadRequest>` zum Auslesen der Personenbindung aus der Bürgerkartenumgebung (BKU) enthält. Die Formulardaten werden direkt an die BKU geschickt und enthalten einen Parameter `DataURL`, der zurück auf die MOA-ID-Auth Komponente verweist. Die BKU schickt als Ergebnis die Personenbindung (`IdentityLink`) an die MOA-ID-Auth Komponente. Die MOA-ID-Auth Komponente überprüft die Personenbindung und schickt bei erfolgreicher Prüfung, einen `<XMLSignRequest>` mit einem zu unterschreibenden Anmeldeformular, das unter anderem die Personenbindung enthält, als Antwort zurück an die BKU. Die BKU stellt das zu unterschreibende Anmeldeformular dar. Unterschreibt der Benutzer mit seiner Bürgerkarte, schickt die BKU das signierte Dokument zurück an die MOA-ID-Auth Komponente, welche die Signatur mit Hilfe von MOA-SP prüft und bei erfolgreicher Prüfung eine entsprechende SAML-Assertion erstellt und speichert. Als Antwort schickt MOA-ID-Auth eine Web-Seite mit einem Redirect auf die Online-Applikation, für die Authentifizierung erfolgt ist. Dieser Redirect enthält ein so genanntes SAML-Artifact: eine Referenz auf die ausgestellte SAML-Assertion. Die Online-Applikation oder die vorgeschaltete MOA-ID-Proxy Komponente kann dann mit dem SAML-Artifact die entsprechende SAML-Assertion bei der MOA-ID-Auth Komponente abholen.

Bei der ersten HTTP-Anfrage holt die Online-Applikation oder die MOA-ID-Proxy Komponente die SAML-Assertion die über das SAML-Artifact referenziert wird von der MOA-ID-Auth Komponente über die entsprechende SOAP-Schnittstelle ab. Die SAML-Assertion enthält dann je nach Konfiguration von MOA-ID Informationen über die erfolgte Authentifizierung und die Personenbindung des authentifizierten Benutzers. Die SAML-Assertion wird nach dem Abholen in der MOA-ID-Auth Komponente gelöscht. Alle weiteren HTTP-Anfragen an die Online-Applikation oder die MOA-ID-Proxy Komponente werden dann der entsprechenden HTTP-Session zugeordnet und als authentifiziert betrachtet. Wird die MOA-ID-Proxy Komponente verwendet, so basiert das Session-Tracking auf Cookies.

Anhang B zeigt ein Beispiel für einen SAML-Request (B.1) und einen entsprechenden SAML-Response (B.2).

---

<sup>2</sup> Die Personenbindung stellt eine Bindung zwischen dem im Zertifikat enthaltenen öffentlichen Schlüssel und einem eindeutigen Ordnungsbegriff für den Bürger (der so genannten *Stammzahl*) her.

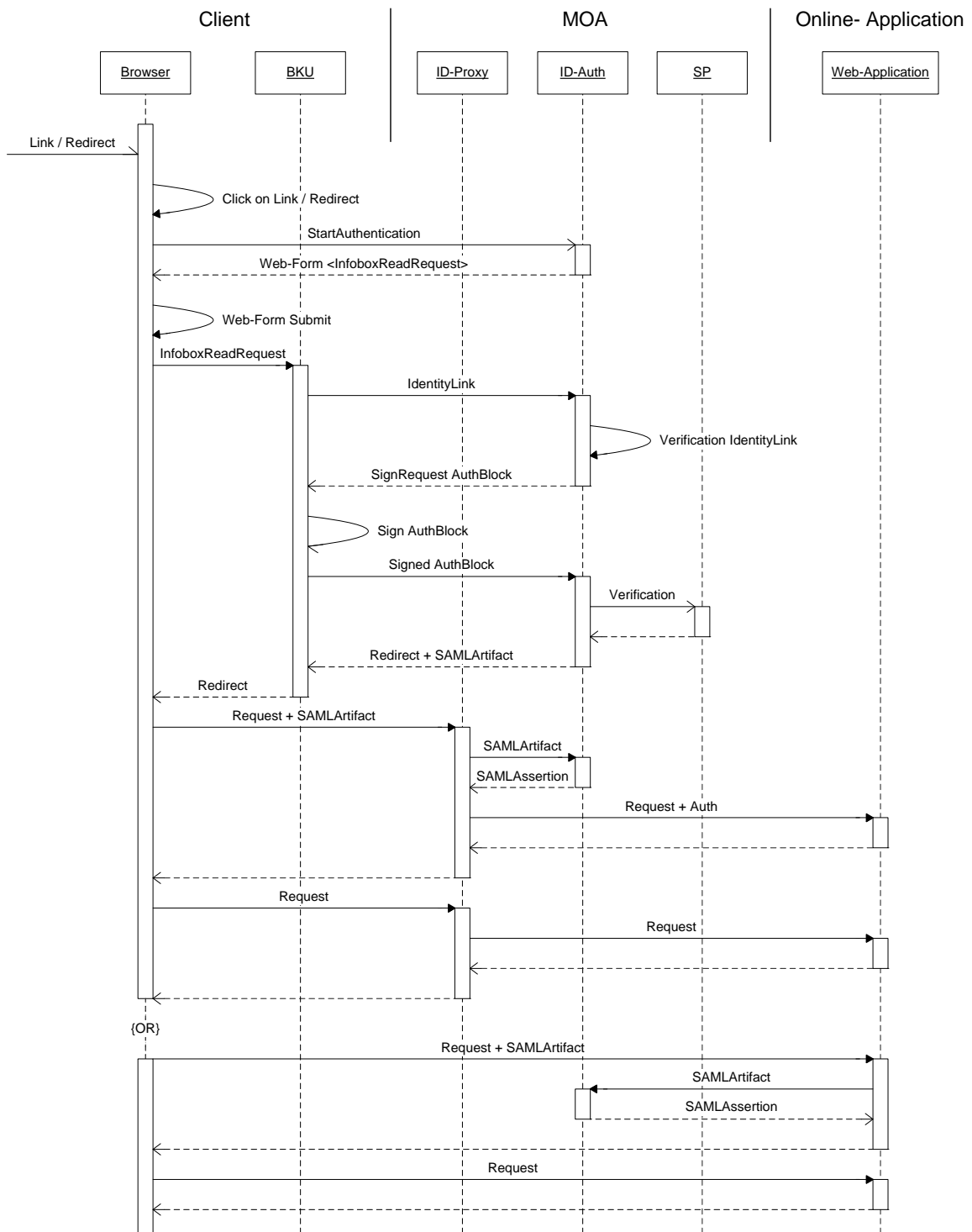


Abb. 1: Sequenz-Diagramm Anmeldung MOA-ID

### 3.2 Single Sign-On

Eine zunächst nahe liegende Möglichkeit, MOA-ID um eine Single Sign-On Fähigkeit zu erweitern, wäre, die SAML-Assertion nach dem Abholen mit dem SAML-Artifact durch die Online-Applikation oder die MOA-ID-Proxy Komponente nicht zu löschen und die Gültigkeit des SAML-Artifacts weiter bestehen zu lassen. Allerdings kommt dem SAML-Artifact eine sicherheitskritische Rolle zu. Jeder der in Kenntnis des SAML-Artifacts ist, kann sich gegenüber der Online-Applikation als authentifizierter Benutzer darstellen. Daher behält das SAML-Artifact nur für einen bestimmten Zeitraum seine Gültigkeit und kann auch nur einmal verwendet werden.

Eine weitere Möglichkeit wäre, dass MOA-ID einen bereits authentifizierten Benutzer anhand einer HTTP-Session wiedererkennt und ein SAML-Artifact für die gespeicherte SAML-Assertion ausstellt, ohne den kompletten Authentifizierungsprozess erneut durchzuführen. Dabei kommt dann aber dem Session-Tracking-Mechanismus des MOA-ID eine gleich sicherheitskritische Rolle, wie einem wiederverwendbarem Artifact. Gelingt es einem Angreifer, die HTTP-Session mit MOA-ID zu stehlen, kann er sich beliebige gültige SAML-Artifacts ausstellen lassen. Nun könnte man für das Session-Tracking auf den in Abschnitt 2 vorgeschlagenen Mechanismus zurückgreifen, allerdings bleibt hier noch immer die sicherheitskritische Rolle des SAML-Artifacts bestehen. Daher schlägt das vorliegende Dokument im folgenden Abschnitt einen alternativen Mechanismus vor, dessen Sicherheit nicht auf der Geheimhaltung von SAML-Artifacts oder Session-IDs, sondern auf asymmetrischen kryptographischen Verfahren beruht.

## 4 MOA-ID Single Sign-On mit Client-Zertifikaten

Aufbauend auf dem in Abschnitt 2 beschriebenen Mechanismus zum sicheren Session-Tracking mit Client-Zertifikaten, ermöglicht die hier vorgestellte Erweiterung sicheres Single Sign-On mit MOA-ID.

Bei der im Abschnitt 3.1 erläuterten Authentifizierung mit MOA-ID wird ein SAML-Artifact vom Client an die Online-Applikation als Nachweis über die erfolgreiche Authentifizierung übergeben. Die Online-Applikation erhält mit diesem SAML-Artifact von MOA-ID eine zugehörige SAML-Assertion, die die erfolgreiche Authentifizierung bestätigt. Werden Client-Zertifikate zur Authentifizierung der SSL/TLS-Session und in Folge zur Erhöhung der Sicherheit des HTTP-Session-Tracking eingesetzt, dann können diese Client-Zertifikate von der Online-Applikation (bzw. der MOA-ID-Proxy Komponente) auch verwendet werden, um eine entsprechende SAML-Assertion für diesen Client bei MOA-ID-Auth abzufragen. Damit müssen keine SAML-Artifacts von MOA-ID zum Client und vom Client weiter an die Online-Applikation übertragen werden.

Die Information über die Identität des Benutzers erhält die Online-Applikation nicht aus dem zur SSL/TLS-Authentifizierung verwendeten Client-Zertifikat, sondern aus der SAML-Assertion von MOA-ID-Auth. Daher kann ein beliebiges Zertifikat zur SSL/TLS-Authentifizierung eingesetzt werden, solange dieses Zertifikat von MOA-ID eindeutig dem authentifizierten Benutzer zugeordnet werden kann. Daher spielt es keine Rolle, ob das Zertifikat eine Information über die Identität des Benutzers enthält oder nicht. Es müssen daher nicht zwingend Zertifikate der Bürgerkarte zum Einsatz kommen. Möglich wäre damit auch die Verwendung von Software-Zertifikaten, die sich leichter in die verwendeten Web-Browser integrieren lassen. Zur Steigerung des Komforts für den Benutzer, kann ein solches Software-Zertifikat auch direkt von MOA-ID bei der Anmeldung ausgestellt werden.

Abbildung 2 zeigt den Ablauf der Authentifizierung mit MOA-ID Single Sign-On. Die Authentifizierung bei MOA-ID erfolgt auf die in Abschnitt 3.1 beschriebene Weise mit der Bürgerkarte. Der Client muss jedoch

bei der Authentifizierung ein SSL-Client-Zertifikat verwenden, besitzt er kein solches Zertifikat, kann dieses auch direkt von MOA-ID ausgestellt werden. MOA-ID erstellt nach erfolgreicher Authentifizierung eine SAML-Assertion für das entsprechende Client-Zertifikat. Diese SAML-Assertion behält ihre Gültigkeit, solange der Client mit MOA-ID eine Session aufrecht erhält bzw. bis er sich aktiv ausloggt. Praktisch sollte dazu ein Browser-Fenster eine Seite von MOA-ID anzeigen, die einen Logout-Button enthält und über einen automatischen Redirect regelmäßig neu geladen wird. Beim Laden der Seite wird von MOA-ID ein Timer zurückgesetzt. Läuft der Timer ab, weil die Seite nicht mehr neu geladen wurde (z.B. weil das Browser-Fenster geschlossen wurde), wird die Session beendet und die zugehörige SAML-Assertion gelöscht. Klickt der Benutzer auf den Logout-Button, wird die zugehörige SAML-Assertion sofort gelöscht.

Greift der Benutzer auf eine Online-Applikation zu und verwendet zur Authentifizierung der SSL/TLS-Session ein Client-Zertifikat, stellt die Online-Applikation eine Anfrage für eine SAML-Assertion für das entsprechende Zertifikat an MOA-ID. Erhält die Online-Applikation eine SAML-Assertion für das entsprechende Zertifikat, so kann sie den Benutzer als authentifiziert betrachten und erhält die Informationen über die Identität des Benutzers aus der SAML-Assertion. Anhang A definiert aufbauend auf dem *SAML Assertion Query/Request Profile* [10] ein Profil zur Abfrage der SAML-Assertion für ein SSL/TLS-Client-Zertifikat von MOA-ID über SOAP.

Ob das ganze Zertifikat, oder welche Merkmale des Zertifikats zur Anfrage für eine SAML-Assertion herangezogen werden sollen, muss noch untersucht werden. Im wesentlichen hängt es davon ab, welche Informationen einfach von einem vorgeschalteten SSL/TLS-Endpunkt (z.B. Apache httpd mit mod\_ssl, IIS) an die nachgeschaltete Online-Applikation (z.B. Apache Tomcat) weitergegeben werden können. Auf jeden Fall müssen diese Informationen geeignet sein, das zugehörige Zertifikat eindeutig zu identifizieren.

## 5 Zusammenfassung

Durch geeignete Erweiterung von MOA-ID ist es möglich, eine Bindung für die Dauer der Anmeldung bei einer Online-Applikation zwischen dem zum Session-Tracking eingesetzten Zertifikat und dem authentifizierten Bürger herzustellen. Damit ist es einerseits möglich, ein erhöhtes Maß an Sicherheit für die Online-Applikation, als auch einen Single Sign-On (SSO) Mechanismus zu realisieren. Durch eine entsprechende Erweiterung der MOA-ID-Proxy Komponente können auch bestehende Web-Applikationen von dieser Erweiterung profitieren. Abschnitt 4 beschreibt diese Erweiterung, Anhang A spezifiziert das zugehörige SAML-Profil.

Für die entsprechende Erweiterung von MOA-ID muss die SOAP-Schnittstelle auf die SAML-Version 2.0 aktualisiert werden. Dabei könnte aber zur Wahrung der Kompatibilität mit bestehenden Online-Applikation auch weiterhin die SAML-Version 1.0 parallel auf der selben Schnittstelle angeboten werden.

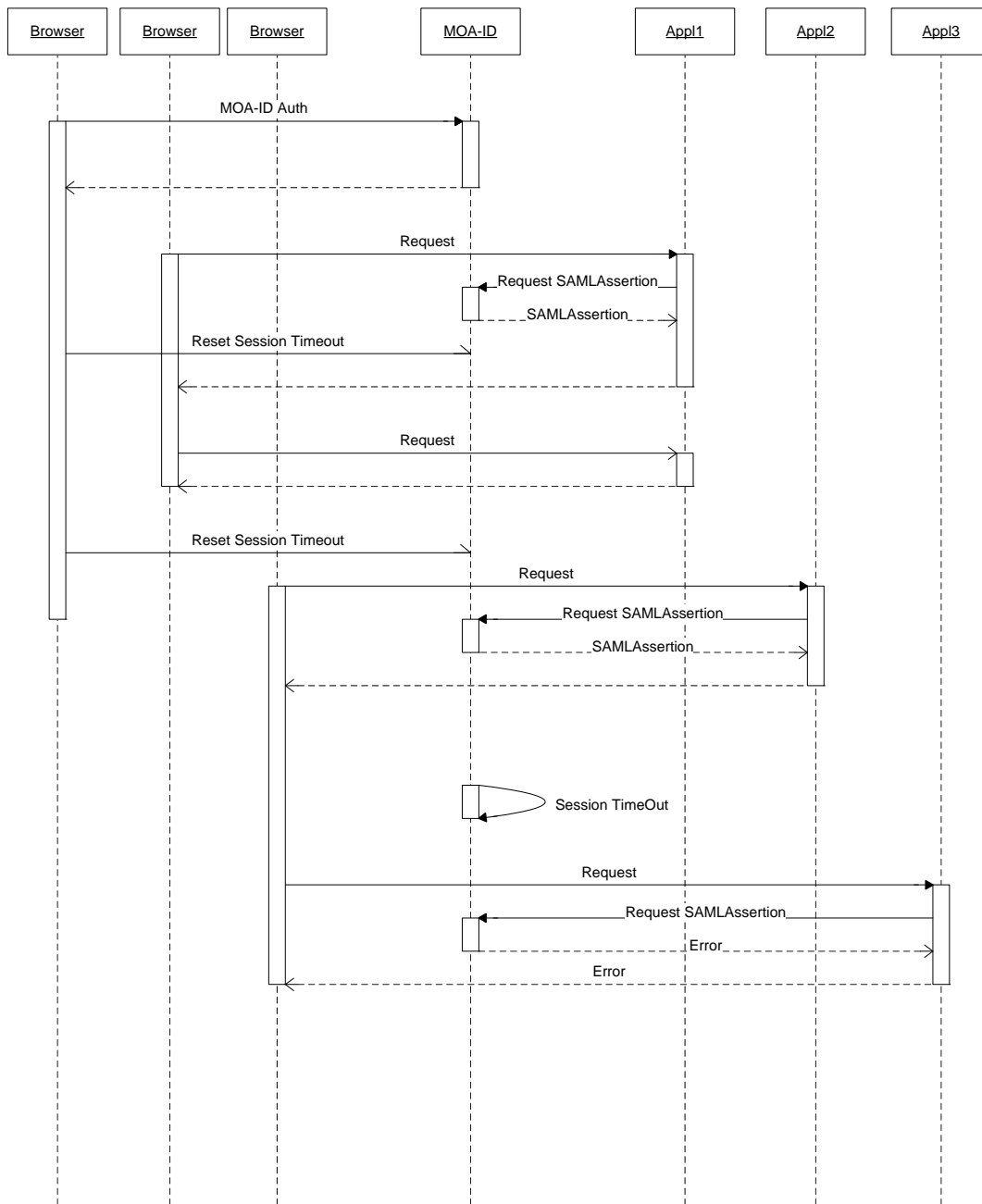


Abb. 2: Sequenz-Diagramm MOA-ID Single Sign-On

## A MOA-ID Assertion Query/Request Profile

[11] definiert ein Protokoll zum Abfragen existierender Assertions. [10] definiert mit dem *Assertion Query/Request Profile* ein Profil zur Verwendung dieses Protokolls mit einer synchronen Bindung, wie z.B. der SOAP-Bindung (definiert in [12]). Dieses Profil baut auf dem *Assertion Query/Request Profile* auf und definiert eine eingeschränkte Menge zulässiger Abfragetypen.

### A.1 Required Information<sup>3</sup>

**Identification:** `http://draft.egiz.gv.at/namespace/moa/20060517`

**Contact Information:** `post@egiz.gv.at`

**Description:** Restricts the *Assertion Query/Request Profile* defined in [10] to a limited number of possible request types.

**Updates:** None.

### A.2 Profil Beschreibung

Dieses Profil stellt eine Einschränkung des *Assertion Query/Request Profile* [10] dar. Aufbauend auf den in [11] definierten Abfragen existierender Assertions definiert es eine eingeschränkte Menge zulässiger Abfragetypen.

#### A.2.1 <AuthnQuery> mit Client-Zertifikat

Die <AuthnQuery> mit Client-Zertifikat dient zur Abfrage einer Assertion für eine per Client-Zertifikat authentifizierte SSL/TLS-Session.

Der Abfragende schickt eine <AuthnQuery> Nachricht mit einem <Subject>-Element. Das <Subject>-Element muss genau ein <SubjectConfirmation>-Element enthalten. Der Wert für das Method-Attribut der <SubjectConfirmation> ist `http://draft.egiz.gv.at/namespace/moa/20060517/AuthnQuery`. Das <SubjectConfirmation>-Element muss genau ein <SubjectConfirmationData>-Element vom Typ `saml:KeyInfoConfirmationDataType` enthalten. Das <ds:KeyInfo>-Element des <SubjectConfirmationData>-Elements muss genau ein <ds:X509Data>-Element enthalten. Das <ds:X509Data>-Element muss entweder ein <ds:X509IssuerSerial>-Element oder <ds:X509Certificate> enthalten.

```

1 <AuthnQuery ID="AQ1028939021" IssueInstant="2006-05-16T17:23:32+02:00" Version="2.0" xmlns="urn:
  oasis:names:tc:SAML:2.0:protocol" xmlns:ds="http://www.w3.org/2000/09/xmldsig#" xmlns:xenc="
  http://www.w3.org/2001/04/xmlenc#" xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion">
2   <saml:Subject>
3     <saml:SubjectConfirmation Method="http://draft.egiz.gv.at/namespace/moa/20060517/AuthnQuery">
4       <saml:SubjectConfirmationData xsi:type="saml:KeyInfoConfirmationDataType">
5         <ds:KeyInfo>
6           <ds:X509Data>
7             <ds:X509Certificate>MIIE3zCCA8egAwIBAgIBSjANBgkqhkiG9w0BAQUFADCB6jELMAkGA1UEBhMCQVQx
8     ...

```

<sup>3</sup> Diese Information wird von [12] gefordert und ist daher in englischer Sprache erfasst.

```

9 /q5XMiJRcEdhUCdlvyBgot/imGSHzQkIWabNdmollF+C9khQgEf0dyjb0MzqTYiJ
10 iQUq</ds:X509Certificate>
11     </ds:X509Data>
12     </ds:KeyInfo>
13   </saml:SubjectConfirmationData>
14 </saml:SubjectConfirmation>
15 </saml:Subject>
16 </AuthnQuery>

```

## B Examples

### B.1 MOA-ID 1.3 SAML-Request

```

1 <samlp:Request IssueInstant="2006-05-16T17:23:32+02:00" MajorVersion="1" MinorVersion="0" RequestID
2   ="6907489010263635536" xmlns:samlp="urn:oasis:names:tc:SAML:1.0:protocol">
3   <samlp:AssertionArtifact>AAFIGuyJxAVdvZ5oCCXqCtqxoUKKmgTpSmG1CfvNQmMtiY0TvUhb44rw</samlp:
4     AssertionArtifact>
5 </samlp:Request>

```

### B.2 MOA-ID 1.3 SAML-Response

```

1 <?xml version="1.0" encoding="ISO-8859-15"?>
2 <samlp:Response InResponseTo="" IssueInstant="2006-05-16T17:23:39+02:00" MajorVersion="1"
3   MinorVersion="0" ResponseID="-821345920806276425" xmlns:saml="urn:oasis:names:tc:SAML:1.0:
4     assertion" xmlns:samlp="urn:oasis:names:tc:SAML:1.0:protocol">
5   <samlp:Status>
6     <samlp:StatusCode Value="samlp:Success" />
7     <samlp:StatusMessage>Anfrage erfolgreich beantwortet</samlp:StatusMessage>
8   </samlp:Status>
9   <saml:Assertion AssertionID="-3283308340509532424" IssueInstant="2006-05-16T17:23:32+02:00"
10     Issuer="https://inspiron:8443/moa-id-auth/" MajorVersion="1" MinorVersion="0" xmlns:pr="http:
11       //reference.e-government.gv.at/namespace/persondata/20020228#" xmlns:saml="urn:oasis:names:
12         tc:SAML:1.0:assertion" xmlns:si="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsi="http:
13           //www.w3.org/2001/XMLSchema-instance">
14     <saml:AttributeStatement>
15       <saml:Subject>
16         <saml:NameIdentifier NameQualifier="urn:publicid:gv.at:wbpk+FN+468924i">l95mPf5Ilmr/
17           05rIvPcdyDRwOVY</saml:NameIdentifier>
18         <saml:SubjectConfirmation>
19           <saml:ConfirmationMethod>http://reference.e-government.gv.at/namespace/moa/20020822#cm</
20             saml:ConfirmationMethod>
21           <saml:SubjectConfirmationData>
22             <saml:Assertion AssertionID="any" IssueInstant="2006-05-16T17:23:19+02:00" Issuer="
23               XXXOtto XXXOttakringer" MajorVersion="1" MinorVersion="0" xmlns:pr="http://
24                 reference.e-government.gv.at/namespace/persondata/20020228#" xmlns:saml="urn:oasis:
25                   names:tc:SAML:1.0:assertion">
26             <saml:AttributeStatement>
27               <saml:Subject>
28                 <saml:NameIdentifier>https://inspiron:8443/moa-id-auth/</saml:NameIdentifier>
29               </saml:Subject>
30               <saml:Attribute AttributeName="wbPK" AttributeNamespace="http://reference.e-
31                 government.gv.at/namespace/moa/20020822#">
32                 <saml:AttributeValue>
33                   <pr:Identification>
34                     <pr:Value>l95mPf5Ilmr/05rIvPcdyDRwOVY</pr:Value>
35                     <pr:Type>urn:publicid:gv.at:wbpk+FN+468924i</pr:Type>
36                   </pr:Identification>
37                 </saml:AttributeValue>
38               </saml:Attribute>
39               <saml:Attribute AttributeName="OA" AttributeNamespace="http://reference.e-
40                 government.gv.at/namespace/moa/20020822#">

```

```

28     <saml:AttributeValue>https://inspiron:8443/WBPKDemo/WBPKDemo/</saml:
29     AttributeValue>
30   </saml:Attribute>
31   <saml:Attribute AttributeName="Geburtsdatum" AttributeNamespace="http://reference.e
32     -government.gv.at/namespace/moa/20020822#">
33     <saml:AttributeValue>1973-01-01</saml:AttributeValue>
34   </saml:Attribute>
35 </saml:AttributeStatement>
36 <dsig:Signature Id="signature-1147793003-29789274-8574" xmlns:dsig="http://www.w3.org
37   /2000/09/xmldsig#">
38   <dsig:SignedInfo>
39     <dsig:CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n
40     -20010315"/>
41     <dsig:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#ecdsa-
42     sha1"/>
43     <dsig:Reference Id="signed-data-reference-0-1147793003-29789274-25360" URI="">
44       <dsig:Transforms>
45         <dsig:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-
46         signature"/>
47         <dsig:Transform Algorithm="http://www.w3.org/TR/1999/REC-xslt-19991116">
48           <xsl:stylesheet version="1.0" xmlns:pr="http://reference.e-government.gv.at
49             /namespace/persondata/20020228#" xmlns:saml="urn:oasis:names:tc:SAML
50             :1.0:assertion" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
51             <xsl:template match="/" xmlns="http://www.w3.org/1999/xhtml">
52               <html xmlns="http://www.w3.org/1999/xhtml">
53                 <head>
54                   <title>Signatur der Anmeldedaten</title>
55                 </head>
56                 <body>
57                   <h1>Signatur der Anmeldedaten</h1>
58                   <p/>
59                   <h4>Mit meiner elektronischen Signatur beantrage ich, <b>
60                     <xsl:value-of select="//@Issuer"/>
61                     </b>,
62                   geboren am
63                   <xsl:value-of select="substring(//saml:Attribute[@AttributeName='Geburtsdatum']/saml:AttributeValue
64                     ,9,2)"/>.<xsl:value-of select="substring(//saml:Attribute[@AttributeName='Geburtsdatum']/saml:
65                     AttributeValue,6,2)"/>.<xsl:value-of select="substring(//saml:Attribute[@AttributeName='
66                     Geburtsdatum']/saml:AttributeValue,1,4)"/>, den Zugang zur gesicherten Anwendung.</h4>
67                   <p/>
68                   <h4>Datum und Uhrzeit: <xsl:value-of select="substring(//
69                     @IssueInstant,9,2)"/>.<xsl:value-of select="substring(//
70                     @IssueInstant,6,2)"/>.<xsl:value-of select="substring(//
71                     @IssueInstant,1,4)"/>, <xsl:value-of select="substring(//
72                     @IssueInstant,12,2)"/>:<xsl:value-of select="substring(//
73                     @IssueInstant,15,2)"/>:<xsl:value-of select="substring(//
74                     @IssueInstant,18,2)"/>
75                   </h4>
76                   <xsl:if test="//saml:Attribute[@AttributeName='wbPK']">
77                     <h4>wbPK(*): <xsl:value-of select="//saml:Attribute[
78                       @AttributeName='wbPK']/saml:AttributeValue/pr:Identification/
79                       pr:Value"/>
80                   </h4>
81                   <p/>
82                   <hr/>
83                   <font size="2">(*) wbPK: Das <i>wirtschaftsbereichsspezifische
84                     Personenkennzeichen</i> wird aus den jeweiligen Stammzahlen
85                     des Bürgers und des Wirtschaftsunternehmens berechnet und
86                     ermöglicht eine eindeutige Zuordnung des Bürgers zum
87                     Wirtschaftsunternehmen.</font>
88                   </xsl:if>
89                 </body>
90               </html>
91             </xsl:template>
92           </xsl:stylesheet>
93         </dsig:Transform>
94       </dsig:Reference>
95     </dsig:SignatureMethod>
96   </dsig:SignedInfo>
97 </dsig:Signature>

```

```
        WithComments"/>
72      </dsig:Transforms>
73      <dsig:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
74      <dsig:DigestValue>EoQWQcJX8cTPY7Eb2u30jy45apc=</dsig:DigestValue>
75    </dsig:Reference>
76    <dsig:Reference Id="etsi-data-reference-0-1147793003-29789274-9504" Type="http://
      uri.etsi.org/01903/v1.1.1#SignedProperties" URI="#etsi-data-object
      -0-1147793003-29789274-4813">
77      <dsig:Transforms>
78        <dsig:Transform Algorithm="http://www.w3.org/2002/06/xmldsig-filter2">
79          <xpf:XPath Filter="intersect" xmlns:xpf="http://www.w3.org/2002/06/xmldsig-
            filter2">id('etsi-data-object-0-1147793003-29789274-4813')/node()</xpf:
            XPath>
80        </dsig:Transform>
81      </dsig:Transforms>
82      <dsig:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
83      <dsig:DigestValue>bKG00J3TF1Kh7fZ/t4+MxsXk4XA=</dsig:DigestValue>
84    </dsig:Reference>
85  </dsig:SignedInfo>
86  <dsig:SignatureValue>TIHX7BvnyJpeD9ZNYjH+zWna/98vTIWk4ZJFXxMH/eWZKk8nAspeUsAKS0Lg
87  YbE6</dsig:SignatureValue>
88  <dsig:KeyInfo>
89    <dsig:X509Data>
90      <dsig:X509Certificate>MIIDqjCCAhKgAwIBAgIRMqzlyU0Hg2/
        DH0mTBwYfKAwwDQYJKoZIhvcNAQEF
91      BQAwwTTEsMBAGAlUEAwWJv1NpZyBDQSAyMSowKAYDVQQKDCFIYXVwdHZlcmJh
92      bmQgw7ZzdGVyci4gU296aWFSdmVycy4xCzAJBgNVBAYTAkFUMB4XDTA1MTAw
93      NzEzMjAwMFoXDTEwMTAwNzEzMjAwMFoWTEfMBOGA1UEAwWFFhYTY3R0byBY
94      WFhPdHRha3JpbmdlcjEgMCGA1UECgwhSGF1cHR2ZXJiYw5kIMO2c3RlcnIu
95      IFNvemlhbHZlcnMuMQ0wCwYDVQQQLDARWU21nMQswCQYDVQQGEWJBVDBJMBMG
96      ByqGSM49AgEGCqQSM49AwEBAzIABDPAT1VLpmjMmbnFuMuT2YJM0TQiBbQx
97      AzKtLXqyBlVpQLYF8VZP5Bz6dOyX+mKmAaOBwzCBwDATBgNVHSMEDDAKgAhI
98      aXVcPK8jKJARBgNVHQ4ECgQIRGKobglRPwQwDgYDVR0PAQH/BAQDAgbAMBYG
99      AlUdIAQPMA0wCwYJKiGACgEEAWYAMEMGCCsGAQUFBwEBBDcwNTAzBggrBgEF
100     BQcwAYYnaHR0cDovL29jc3AuZW5hcmQuc296aWFSdmVyc21jaGVydw5nLmF0
101     MCKGA1UdEQQIMCCBHmRhbml1bGUuYm9zY2hldHRvQGNoaXBRYXJ0ZS5hdDAN
102     BgkqhkiG9w0BAQUFAAOCAQEAg1J7pVqymABeXMrwoKcWmV9GS5eutsIGfbQt
103     ezTzG96Tt/dXnW5s5FVynLPKrvU6eX31jln+WP0sVM2j84s6XYchYn/1slMP
104     aFGVUX3BqPZ2kvZFDEoa/9+3bBR9nsXkkd0GpF/YqYAg9NKzivXm6Iq81Zn9
105     uWpXn7ax+gYCjBwgVH8V5WjyWz7b9Ls7COSVUPFz+8WQI9TGSz8hZvJPHPr
106     5vRfpi2+kw0tZECzq34KUXYgQlwoKhJ17R5QpXb2ewYd73beNYH+j48gJd3w
107     KbezpaFsBGUjK/3/g/jQx2r2wqPqSaKg14PDYq4r5/KU3xAY5RGadWktpHH
108     2tqIsCWzIPXE4mzjFXvZ5PqMT11Df1GBF/AzYc4yy0nbgR4xRjv3blChoh
109     XK2Ww/MggUWHZPyLgw55JfRbk9CFVoGdUWXqQAV7XcQYCYZGK1TPzJBe1Q8G
110     4eaVcCebb47IPQZopGEiaGmG+PxhBDZ5uVcZykxs8mzWiSVKwby+MVH2
111   </dsig:X509Certificate>
112   </dsig:X509Data>
113   </dsig:KeyInfo>
114   <dsig:Object Id="etsi-data-object-0-1147793003-29789274-4813">
115     <etsi:QualifyingProperties Target="#signature-1147793003-29789274-8574" xmlns:
        etsi="http://uri.etsi.org/01903/v1.1.1#">
116       <etsi:SignedProperties>
117         <etsi:SignedSignatureProperties>
118           <etsi:SigningTime>2006-05-16T15:23:23Z</etsi:SigningTime>
119         <etsi:SigningCertificate>
120           <etsi:Cert>
121             <etsi:CertDigest>
122               <etsi:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1
                "/>
123               <etsi:DigestValue>nEzNXfz00F2v9F4BoQmVx4q1A=</etsi:DigestValue>
124             </etsi:CertDigest>
125           <etsi:IssuerSerial>
126             <dsig:X509IssuerName>C=AT,O=Hauptverband österr. Sozialvers.,CN=VSig
                CA 2</dsig:X509IssuerName>
127             <dsig:X509SerialNumber>17243936570799158538778671906756073826316</
                dsig:X509SerialNumber>
128           </etsi:IssuerSerial>

```

```

129         </etsi:Cert>
130     </etsi:SigningCertificate>
131     <etsi:SignaturePolicyIdentifier>
132         <etsi:SignaturePolicyImplied/>
133     </etsi:SignaturePolicyIdentifier>
134 </etsi:SignedSignatureProperties>
135 <etsi:SignedDataObjectProperties>
136     <etsi:DataObjectFormat ObjectReference="#signed-data-reference
        -0-1147793003-29789274-25360">
137         <etsi:MimeType>text/html</etsi:MimeType>
138     </etsi:DataObjectFormat>
139 </etsi:SignedDataObjectProperties>
140 </etsi:SignedProperties>
141 </etsi:QualifyingProperties>
142 </dsig:Object>
143 </dsig:Signature>
144 </saml:Assertion>
145 <saml:Assertion AssertionID="szo.bmi.gv.at-AssertionID112869913240743" IssueInstant
        ="2005-10-07T15:32:12+01:00" Issuer="http://portal.bmi.gv.at/ref/szo/issuer"
        MajorVersion="1" MinorVersion="0" xmlns:dsig="http://www.w3.org/2000/09/xmldsig#"
        xmlns:ecdsa="http://www.w3.org/2001/04/xmldsig-more#" xmlns:pr="http://reference.e-
        government.gv.at/namespace/persondata/20020228#" xmlns:saml="urn:oasis:names:tc:
        SAML:1.0:assertion" xmlns:si="http://www.w3.org/2001/XMLSchema-instance">
146 <saml:AttributeStatement>
147     <saml:Subject>
148         <saml:SubjectConfirmation>
149             <saml:ConfirmationMethod urn:oasis:names:tc:SAML:1.0:cm:sender-vouches</saml:
                ConfirmationMethod>
150             <saml:SubjectConfirmationData>
151                 <pr:Person si:type="pr:PhysicalPersonType">
152                     <pr:Identification>
153                         <pr:Value>l95mPf5IImr/O5rIvPcdyDRwOVY</pr:Value>
154                         <pr:Type urn:publicid:gv.at:wbpk+FN+468924i</pr:Type>
155                     </pr:Identification>
156                     <pr:Name>
157                         <pr:GivenName>XXXOtto</pr:GivenName>
158                         <pr:FamilyName primary="undefined">XXXOttakringer</pr:FamilyName>
159                     </pr:Name>
160                     <pr:DateOfBirth>1973-01-01</pr:DateOfBirth>
161                 </pr:Person>
162             </saml:SubjectConfirmationData>
163         </saml:SubjectConfirmation>
164     </saml:Subject>
165     <saml:Attribute AttributeName="CitizenPublicKey" AttributeNamespace="urn:publicid:
        gv.at:namespaces:identitylink:1.2">
166         <saml:AttributeValue>
167             <ecdsa:ECDSAKeyValue>
168                 <ecdsa:DomainParameters>
169                     <ecdsa:NamedCurve URN="urn:oid:1.2.840.10045.3.1.1"/>
170                 </ecdsa:DomainParameters>
171                 <ecdsa:PublicKey>
172                     <ecdsa:X Value="1268935989905593888842497408361401244425865965241629946115"
                            si:type="ecdsa:PrimeFieldElemType"/>
173                     <ecdsa:Y Value="1242583556546726918377254031913220431062044823430235334145"
                            si:type="ecdsa:PrimeFieldElemType"/>
174                 </ecdsa:PublicKey>
175             </ecdsa:ECDSAKeyValue>
176         </saml:AttributeValue>
177     </saml:Attribute>
178     <saml:Attribute AttributeName="CitizenPublicKey" AttributeNamespace="urn:publicid:
        gv.at:namespaces:identitylink:1.2">
179         <saml:AttributeValue>
180             <ecdsa:ECDSAKeyValue>
181                 <ecdsa:DomainParameters>
182                     <ecdsa:NamedCurve URN="urn:oid:1.2.840.10045.3.1.1"/>
183                 </ecdsa:DomainParameters>
184             <ecdsa:PublicKey>

```

```
185         <ecdsa:X Value="542003903456552843010188713067144025199717915522743477407"
186             si:type="ecdsa:PrimeFieldElemType"/>
187         <ecdsa:Y Value="447232787244807255655353982171129068312004543525848553645"
188             si:type="ecdsa:PrimeFieldElemType"/>
189     </ecdsa:PublicKey>
190 </ecdsa:ECDSAKeyValue>
191 </saml:AttributeValue>
192 </saml:Attribute>
193 </saml:AttributeStatement>
194 <dsig:Signature>
195   <dsig:SignedInfo>
196     <dsig:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-c14n
197       #"/>
198     <dsig:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
199     <dsig:Reference URI="">
200       <dsig:Transforms>
201         <dsig:Transform Algorithm="http://www.w3.org/TR/1999/REC-xpath-19991116">
202           <dsig:XPath>not (ancestor-or-self::pr:Identification)</dsig:XPath>
203         </dsig:Transform>
204         <dsig:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-
205           signature"/>
206       </dsig:Transforms>
207       <dsig:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
208       <dsig:DigestValue>qrrl0lSASs6maGm2tsVFvbEwHS0=</dsig:DigestValue>
209     </dsig:Reference>
210     <dsig:Reference Type="http://www.w3.org/2000/09/xmldsig#Manifest" URI="#manifest
211       ">
212       <dsig:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
213       <dsig:DigestValue>grxi03fYt8NzzbGImn0upSJnGYE=</dsig:DigestValue>
214     </dsig:Reference>
215   </dsig:SignedInfo>
216   <dsig:SignatureValue>QIzVXCAoFnrw61sX0cb0Ztp0AN/hR6vaakUimtJlSr+
217     aaBkZCg0ZvbSFo7fbRK7EEe54qrLCbna
218     25+7jaIrEchVr94rA0bXKZc96syu7y3QaeLW4s4IiwhrX0ueMYpdIRhxVGMGR4B6yMDeTGZE/Xn77
219     cp00tLzxGA9o/DGtXIA= </dsig:SignatureValue>
220   <dsig:KeyInfo>
221     <dsig:X509Data>
222       <dsig:X509Certificate>
223         MIIETCCA5mgAwIBAgICXDkdQYJKoZIhvcNAQEFBQAwZ8xCzAJBgNVBAYTAKFUMUgWRgYDVQQK
224         Ez9BLVRydXN0IEIEdlcy4gZi4gU2l1jAGVyaGVpdHNzeXN0ZW11IGltIGVsZWt0ci4gRGF0ZW52ZXJr
225         ZWhyIEEdtYkxIjAgBgNVBAsTGWEtc2lnbi1jb3Jwb3JhdGUtbGlnaHQtdmExIjAgBgNVBAMTGWET
226         c2lnbi1jb3Jwb3JhdGUtbGlnaHQtdmExHhcNMjE5MTk0OTQ5Wjc3ZGVzIGdmLiBnaXRnaWVkcYkZlIjAg
227         Y2h1dHprb21taXNzaW9uMRQwFgYDVQDEw9OAwTvbGF1cyBTY2h3YWIxXDAKBG9NBWAwTARyLjCB
228         nzANBgkqhkiG9w0BAQEFAA0BJQAQYkCgYEAorTi0cYq4MHF45zWaF3AnA+bUfF+ZSxaXc0CLM2A
229         gioFvylfD592WrrSYuIfcNDpiziFwPnnpJqOwoPnP4Bhzn2FUeKhxhXipsHdSRuwOMrq/CQU8wUjW
230         nefo4EZyZEo4wTsVzkrhF90k10oQPa2J+r14/bcKZ3fc1h6aDEoxWsCAwEAAaOCAacwggjMAkG
231         A1UdEwQMAAwEQYDVRO0BAoECEFFvos8m+GnTMF0GA1UdIARWFMQWUgYHkiGAEQEOATBHMEUGCCSG
232         AQUFBwIBFjlodHRwOi8vd3d3LmEtDHJ1c3QuYXQvZG9jcy9jcc9hc2lnbi1nb3Zlcm5tZW50LXN1
233         cnZlci5wZGYwEwYDVRO0BAwCoAITp5/lC/JHx8wfwYIKwYBBQUHAQEeczBxMCCGCCSGAQUFBzAB
234         hhtodHRwOi8vb2Nzc5hLXRydXN0LmF0L29jcc3AwRgYIKwYBBQUHMAKGOMh0dHA6Ly93d3cuYS10
235         cnVzdC5hdC9jZlJ0cy9hLXNpZ24tY29ycG9yYXRlLWxpZ2h0LTAxYS5jcnQwDgYDVRO0PAQH/BAQD
236         AgeAMG4GA1UdHwRnMGUwY6BhoF+GXWxkYXA6Ly9sZGFwLmEtDHJ1c3QuYXQvbn3U9YS1zaWduLWNv
237         cnBvcnF0ZS1saWdodC0wMScxvPUETVHJ1c3QsYz1BVD9jZlJ0aWZpY2F0ZlJldm9jYXRpb25saXN0
238         PzA0BgcqKAAKAQEBBAMBAf8wDQYJKoZIhvcNAQEFBQADggEBAEWCrcjcxUw64Bp35nPsaIaKSfhEQ
239         J1kX0urJ+72wrOIb5tY0MfRI7mhMlpUWNYpxe21pBSE2vg6NRhstCNEkPqtOzpcnwqWZ3VrqrQuI
240         ag+sePhVp1vwOVH/GCSWjma3RB9hlz011jFhL4hgDQV+SkqG5IPxel0DeAly56hrWNIpogNf13qF
241         ZGmDCfnaYB+nYa0Chyo8JqwJGLoDucRpWdvlum7Px+2d/n/94KnPqwI5X4aFAzcgJ0nL9LoVEni
242         7ka7DScxPeGfQklBNhWTFwgBqTTULIZ5hePe+QHYtv+Y/1QcK3mRSjzUs4NFXJLGDWCKvxTDVeud
243         5B7SJ3xtJWM=
244       </dsig:X509Certificate>
245     </dsig:X509Data>
246   </dsig:KeyInfo>
247   <dsig:Object>
248     <dsig:Manifest Id="manifest">
249     <dsig:Reference URI="">
```

```
246         <dsig:Transforms>
247             <dsig:Transform Algorithm="http://www.w3.org/TR/1999/REC-xpath-19991116">
248                 <dsig:XPath>not(ancestor-or-self::dsig:Signature)</dsig:XPath>
249             </dsig:Transform>
250         </dsig:Transforms>
251         <dsig:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
252         <dsig:DigestValue>Jkykc2EQhVWN2mPBfx90jHyxwPs=</dsig:DigestValue>
253         </dsig:Reference>
254     </dsig:Manifest>
255 </dsig:Object>
256 </dsig:Signature>
257 </saml:Assertion>
258 </saml:SubjectConfirmationData>
259 </saml:SubjectConfirmation>
260 </saml:Subject>
261 <saml:Attribute AttributeName="PersonData" AttributeNamespace="http://reference.e-government.
262     gv.at/namespace/persondata/20020228#">
263     <saml:AttributeValue>
264         <pr:Person xmlns:pr="http://reference.e-government.gv.at/namespace/persondata/20020228#"
265             xsi:type="pr:PhysicalPersonType">
266             <pr:Identification>
267                 <pr:Value>195mPf5I1mr/05rIvPcdyDRwOVY=</pr:Value>
268                 <pr:Type>urn:publicid:gv.at:wbpk+FN+468924i</pr:Type>
269             </pr:Identification>
270             <pr:Name>
271                 <pr:GivenName>XXXOtto</pr:GivenName>
272                 <pr:FamilyName primary="undefined">XXXOttakringer</pr:FamilyName>
273             </pr:Name>
274             <pr:DateOfBirth>1973-01-01</pr:DateOfBirth>
275             </pr:Person>
276         </saml:AttributeValue>
277     </saml:Attribute>
278     <saml:Attribute AttributeName="isQualifiedCertificate" AttributeNamespace="http://reference.e
279         -government.gv.at/namespace/moa/20020822#">
280         <saml:AttributeValue>>false</saml:AttributeValue>
281     </saml:Attribute>
282     <saml:Attribute AttributeName="bkuURL" AttributeNamespace="http://reference.e-government.gv.
283         at/namespace/moa/20020822#">
284         <saml:AttributeValue>http://127.0.0.1:3495/http-security-layer-request</saml:AttributeValue
285         >
286     </saml:Attribute>
287 </saml:AttributeStatement>
288 </saml:Assertion>
289 </samlp:Response>
```

## Referenzen

- [1] ARGE SPEZIFIKATION MOA: *Spezifikation Module für Online Applikationen (MOA) – ID*. Konvention. [http://www.cio.gv.at/onlineservices/basicmodules/moa-id/specification/MOA\\_ID\\_1.3\\_20060315.pdf](http://www.cio.gv.at/onlineservices/basicmodules/moa-id/specification/MOA_ID_1.3_20060315.pdf). Version: 1.3-20060315, Abruf: 27.5.2006
- [2] FIELDING, R. ; GETTYS, J. ; MOGUL, J. ; FRYSTYK, H. ; MASINTER, L. ; LEACH, P. ; BERNERS-LEE, T.: *Hypertext Transfer Protocol – HTTP/1.1*. RFC 2616 (Draft Standard). <http://www.ietf.org/rfc/rfc2616.txt>. Version: Juni 1999 (Request for Comments). – Updated by RFC 2817
- [3] POSTEL, J.: *Transmission Control Protocol*. RFC 793 (Standard). <http://www.ietf.org/rfc/rfc793.txt>. Version: September 1981 (Request for Comments). – Updated by RFC 3168
- [4] POSTEL, J.: *Internet Protocol*. RFC 791 (Standard). <http://www.ietf.org/rfc/rfc791.txt>. Version: September 1981 (Request for Comments). – Updated by RFC 1349
- [5] KRISTOL, D. ; MONTULLI, L.: *HTTP State Management Mechanism*. RFC 2965 (Proposed Standard). <http://www.ietf.org/rfc/rfc2965.txt>. Version: Oktober 2000 (Request for Comments)
- [6] FRANKS, J. ; HALLAM-BAKER, P. ; HOSTETLER, J. ; LAWRENCE, S. ; LEACH, P. ; LUOTONEN, A. ; STEWART, L.: *HTTP Authentication: Basic and Digest Access Authentication*. RFC 2617 (Draft Standard). <http://www.ietf.org/rfc/rfc2617.txt>. Version: Juni 1999 (Request for Comments)
- [7] JAGANATHAN, K. ; ZHU, L. ; BREZAK, J.: *Kerberos based HTTP Authentication in Windows*. Internet-Draft draft-jaganathan-kerberos-http-01.txt (Informational). <http://www.ietf.org/internet-drafts/draft-jaganathan-kerberos-http-01.txt>. Version: Juli 2005 (Internet-Draft)
- [8] RESCORLA, E.: *HTTP Over TLS*. RFC 2818 (Informational). <http://www.ietf.org/rfc/rfc2818.txt>. Version: Mai 2000 (Request for Comments)
- [9] HOLLOSI, A. ; KARLINGER, G.: *XML-Definition der Personenbindung*. <http://www.buergerkarte.at/konzept/personenbindung/spezifikation/20050214/Personenbindung-20050214.pdf>. Version: 1.2.2, Abruf: 27.5.2006
- [10] S. CANTOR ET AL.: *Profiles for the OASIS Security Assertion Markup Language (SAML) V2.0*. <http://docs.oasisopen.org/security/saml/v2.0/saml-profiles-2.0-os.pdf>. Version: 2.0, März 2005 (OASIS SSTC)
- [11] S. CANTOR ET AL.: *Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0*. <http://docs.oasisopen.org/security/saml/v2.0/saml-core-2.0-os.pdf>. Version: 2.0, März 2005 (OASIS SSTC)
- [12] S. CANTOR ET AL.: *Bindings for the OASIS Security Assertion Markup Language (SAML) V2.0*. <http://docs.oasisopen.org/security/saml/v2.0/saml-bindings-2.0-os.pdf>. Version: 2.0, März 2005 (OASIS SSTC)