

<h1>Spezifikation Portal Verbund Protokoll</h1>	Konvention		
	PVP 1.8.9 (2004-07-31)		
	Entwurf öffentlich		
Kurzbeschreibung:	<p>Das Portalverbundsystem ermöglicht das Zusammenwirken von Stammportalen zur Registrierung von Benutzern mit ihren Zugriffsrechten einerseits und Anwendungsportalen zur Überprüfung des berechtigten Zuganges zu Anwendungen andererseits.</p> <p>Die Authentifizierung und Autorisierung kann delegiert werden.</p> <p>Der Aufwand für die Verwaltung der Benutzer wird reduziert und ein Single Sign-On unterstützt.</p>		
Autor:	Rainer Hörbe ¹ (BMI-ITMS)	Projektteam / Arbeitsgruppe:	Arbeitsgruppe Kommunikationsarchitektur
Lokation:	http://reference.e-government.gv.at -> Portalverbund		

Stelle:	vorgelegt am:	angenommen am:
IKT-Board		
Städtebund		
Gemeindebund		
Länder		

¹ Viele Köche verhindern das Einerlei. Dank für Beiträge, Reviews und Feedback (ohne Anspruch auf Vollständigkeit) an: Bernd Martin, Franz Grandits, Harald Stradal, Ignaz Gritschenberger, Gottfried Luef, Günther Schmittner, Peter Pfläging, Peter Pichler und Wolfgang Kremser.

Inhaltsverzeichnis

1. Zweck.....	3
2. Begriffsbestimmung.....	3
3. Grundbestandteile des PVP.....	3
3.1. Metainformationen.....	3
3.2. Authentifizierungsinformationen.....	3
3.3. Autorisierungsinformationen.....	4
3.4. Verrechnungsinformationen.....	4
4. Schreibweise.....	4
5. Grammatik des Portalverbund-Protokolls.....	6
Beschreibung der Parameter.....	8
6. Protokollbindung HTTP.....	12
7. Protokoll-Bindung PVP – SOAP.....	12
7.1. Namespaces.....	12
7.2. Portalverbund pvpToken.....	12
7.3. Fehlermeldungen.....	13
8. Zertifikate.....	14
9. Fehlermeldungen.....	14
Anhang A HTTP Beispiel-Request User Principal.....	15
Anhang B HTTP Beispiel-Request System Principal.....	15
Anhang C Beispiel für Application Chaining.....	16
Anhang D SOAP Beispiel-Request	18
Anhang E Beispiele für Rechte und Rechteparameter.....	19
Anhang F Implementierungshinweise.....	20
a. Betrieb von Anwendungsportalen als Reverse Proxy.....	20
b. Betrieb von Stammportalen als Reverse Proxy.....	20
c. Verifikation von Client-Zertifikaten.....	20
Anhang G Referenzen.....	21
Anhang H Funktionelle Änderungen von Version 1.7 zu 1.8.....	22

1. Zweck

Das Portalverbundsystem ermöglicht die Delegation von Benutzer-Identitäten und Berechtigungen [PV-Whitepaper]. Das Protokoll erweitert die Kommunikation zwischen Stamm- und Anwendungsportalen, indem vertrauenswürdige Aussagen über Authentizität, Autorisierung und Verrechnungsdaten von Benutzern kommuniziert werden.

Autorisierung bedeutet in diesem Zusammenhang, dass einem Benutzer für den Zugriff auf eine Ressource Rechte, Rechteparameter und eine Sicherheitsklasse zugewiesen werden.

Die Kommunikation zwischen den Portalen muss Integrität und Vertraulichkeit gewährleisten.

Darüber hinaus ist PVP für weitere Zwecke vorgesehen:

- Kommunikation zwischen Behörden und Nicht-Behörden auf Grund bilateraler Vereinbarungen
- Kommunikation zwischen internen Stamm- und Anwendungsportalen
- Kommunikation zwischen Anwendungsportalen und Anwendungen

2. Begriffsbestimmung

Die Begriffe sind in [PVV 1.0] definiert.

Ergänzend dazu wird festgelegt:

Verrechnungsdaten

bestehen aus der Identifikation des Rechnungsempfängers, und der Liste der möglichen Kostenstellen und Gebührenstufen des Anwenders im Kontext der Anwendung. Die Auswahl der konkreten Kostenstelle und Gebührenstufe einer Transaktion erfolgt in der Anwendung, nicht im PVP.

3. Grundbestandteile des PVP

Die Information, die im PVP übermittelt werden, bestehen aus Attributen, die nach dem Schema LDAP-gv.at modelliert sind.

3.1. Metainformationen

Die Metainformation enthält die Versionsnummer. Diese wird aufgrund der aktuellen Implementierung des Clients gesetzt.

3.2. Authentifizierungsinformationen

Sind für Nachvollziehbarkeitszwecke für das Applikationsportal notwendig. Sie beinhalten Organisation und Organisationseinheit, sowie Identifikationsmerkmale des Benutzers (gvGid, cn, uid, gvFunction). Informationen sind alle nach dem Schema LDAP-gv.at modelliert.

3.3. Autorisierungsinformationen

Sind jene Informationen, die benötigt werden, um dem Benutzer bei der Applikation die Zugriffsberechtigung zu erteilen. Das sind ApplicationRights (Anwendungsrollen) und gvUserRestrictions. Die Einschränkungen können von der Anwendung als ALLOW oder DENY Regeln interpretiert werden. Einschränkungen kann es z.B. für regionale oder Organisationsbezogene Aspekte geben. Authentifizierungs- und Autorisierungsinformation können sich in bei der Organisation und Organisationseinheit unterscheiden; deswegen sind sie extra angeführt. Ein Benutzer kann dadurch im Namen einer anderen Organisation etwas ausführen.

3.4. Verrechnungsinformationen

Dient der Verrechnung von Transaktionsgebühren. Da weder Kostenstelle noch Gebührenstufe für einen Benutzer immer feststeht, gibt es die Möglichkeit durch das Stammportal Werte vorzugeben.

4. Schreibweise

Diese Spezifikation verwendet EBNF (erweiterte Backus-Naur Form). EBNF beschreibt eine Grammatik, mit der die Menge der möglichen Zeichenketten einer „Sprache“ definiert wird.

Diese Schreibweise wird wie folgt definiert:

Name := Regel

Eine (Produktions-) Regel besteht aus einem oder mehreren Elementen. Ein Element ist Literal oder wiederum eine Regel. „:=“ heißt so viel wie „besteht aus“.

Regeln, die in [RFC2616] als „Basic Rules“ definiert sind, werden mit Großbuchstaben geschrieben, wie SPACE, TAB, CRLF, DIGIT, ALPHA, LWS...

Regeln können zur Klarstellung im Fließtext mit spitzen Klammern „<>“ geschrieben werden.

"Literal"

Literale sind durch Anführungszeichen markiert und bedeuten fixen Text, der nicht mehr weiter ersetzt wird. Der Text ist case-insensitive, wenn es nicht anders angegeben ist.

Kommentar

Text nach einem Semikolon bis zum Zeilenende wird als Kommentar betrachtet, z.B. ; Erklärung in die Spezifikation eingebettet

Regel-1 | Regel-2

Elemente, die durch einen vertikalen Strich (|) getrennt sind, sind Alternativen, z.B. "JA" | "NEIN"

!Regel-1 Regel-2!

Elemente innerhalb von Rufzeichen werden als einzelnes Element betrachtet. Z.B. kann die Regel <"bearbeite " ! "alle " | "keine " ! "Anfragen"> zu "bearbeite alle Anfragen" und "bearbeite keine Anfragen" führen. Die übliche Schreibweise mit runden Klammern „()“ wird hier nicht verwendet, um die Lesbarkeit von Parameterlisten zu verbessern.

*Regel +Regel {N}Regel {N-M}Regel

Das Zeichen vor einem Element bedeutet die Anzahl der Wiederholungen des Elements:

*	0 oder mehr
+	1 oder mehr
{N}	N
{N-M}	größer gleich N und kleiner gleich M

[Regel]

Eckige Klammern bedeuten, dass das Element optional ist.

#;Regel

Das #-Zeichen ist eine Kurzschreibweise für eine Liste mit Semikolon als Literal für das Trennzeichen. Die Form <n>#<t> bedeutet mindestens <n> Elemente, die von einem oder mehreren Trennzeichen <t>, und optional LWS getrennt sind.

(*LWS element *(*LWS ";" *LWS element))

kann dargestellt werden durch

1#;element

Null-Elemente sind erlaubt, werden aber nicht gezählt. #Element bedeutet 0 bis unendlich viele Elemente, 1#element 1 ein oder mehrere Elemente.

5. Grammatik des Portalverbund-Protokolls

Das Protokoll besteht aus eine Liste von abstrakten Parametern, wobei jeder Parameter als Key-Value-Paar in der EBNF definiert ist. Die resultierenden Keys haben das Format BEREICH-Attribut, wobei BEREICH entweder AUTHENTICATE, AUTHORIZE oder ACCOUNTING ist. Zusätzlich gibt es noch den Key *Version*, der als Metainformation zu keinem BEREICH gehört.

```
pvp-Parameters := pvp-Version pvp-Authentication [pvp-Authorization] [pvp-
  Accounting] [pvp-Chained-parameters]
```

```
pvp-Version:= "Version: " pvp-Version-Numbers
```

```
pvp-Version-Numbers := "1.0" | "1.1" | "1.2" | "1.8"
```

```
pvp-Authentication := pvp-Participant pvp-Principal
```

```
pvp-Participant := Auth-Participant [Auth-OuDomain]
```

```
Auth-OuDomain := "AUTHENTICATE-gvOuDomain: " +CHAR
```

```
Auth-Participant := "AUTHENTICATE-participantId: " +CHAR
```

```
pvp-Principal := Auth-User-Principal | Auth-System-Principal
```

```
Auth-User-Principal := Auth-UserId Auth-Cn Auth-Gid [Auth-Function]
  Auth-OuId Auth-Ou [Auth-SecClass]
```

```
Auth-System-Principal := Auth-UserId Auth-Cn Auth-OuId Auth-Ou [Auth-SecClass]
```

```
pvp-Authorization := [Auz-ActingOrg] Auz-Roles
```

```
pvp-Accounting := Acc-invoiceRecptId Acc-CostCenterIdList Acc-ChargeCodeList
```

```
pvp-Chained-parameters := pvp-Version pvp-Authentication [pvp-Authorization] (es
  gilt zusätzlich unten stehende Zusatzregel)
```

```
Auth-UserId := "AUTHENTICATE-userId: " +CHAR ;
```

```
Auth-Cn := "AUTHENTICATE-cn: " +OCTET
```

```
Auth-Gid := "AUTHENTICATE-gvGid: " +CHAR
```

```
Auth-OuId := "AUTHENTICATE-gvOuId: " +CHAR
```

```
Auth-Ou := "AUTHENTICATE-ou: " +OCTET
```

```
Auth-Function := "AUTHENTICATE-gvFunction: " +OCTET
```

```
Auth-SecClass := "AUTHENTICATE-gvSecClass: " "0" | "1" | "2" | "3"
```

```
Auz-ActingOrg := Auz-OuId Auz-Ou
```

```
Auz-OuId := " AUTHORIZE -gvOuId: " +CHAR
```

```
Auz-Ou := " AUTHORIZE -ou: " +OCTET
```

```
Auz-Roles := "AUTHORIZE-Roles: "
  1#;Auz-Right["("##,Auz-RoleParameter)"] [;]
```

Auz-Right := +OCTET

Auz-RoleParameter := Auz-Parameter"=" +OCTET

Auz-Parameter := "GKZ" | "DST" | "BL" | "gvOuId" | +CHAR

Acc-InvoiceRecptId := "ACCOUNTING-InvoiceRecptId: " +CHAR

Acc-CostCenterIdList := "ACCOUNTING-gvCostCenterId: " Acc-CostCenterIdItem

Acc-CostCenterIdItem := ["<default>"] Acc-CostCenterIdValue
[, #, Acc-CostCenterIdValue] [, "<user defined>"]

Acc-CostCenterIdValue := {1-25}!ALPHA | DIGIT | SPACE | "-" | "_" | "/"!

Acc-ChargeCodeList := "ACCOUNTING-gvChargeCode: " Acc-ChargeCodeItem

Acc-ChargeCodeItem := ["<default>"] Acc-ChargeCodeValue
[, #, Acc-ChargeCodeValue]

Acc-ChargeCodeValue := DIGIT [DIGIT]

Zusatzregel für die Produktion von pvp-Chained-parameters:

Nach der Ausführung der Produktionsregeln gilt für die HTTP-Bindung:

Bei den Keys, die für das Application Chaining vom vorhergehenden Request übernommen werden, wird "nn-" vorangestellt. Dabei ist nn eine laufende Nummer für die Reihenfolge der Verarbeitung beginnend bei 01 für den Endbenutzer. Die Anzahl der Stufen im Chaining ist auf 2 beschränkt.

D.h., dass der letzte Benutzer in der Kette (der also die Operation am Anwendungsportal direkt ausführt) immer die Key-Value-Paare ohne laufende Nummer hat, und die zeitlich vorhergehenden Schritte beginnend beim Endbenutzer durchnummeriert werden.²

Mit der Versionsnummer wird im Request definiert, welche Version des PVP verwendet wird:

PVP-Version		Dokumentversion
1	0	1.4 (BMI-Gateway-Protokoll)
1	1	1.5.3
1	2	1.6, 1.7
1	8	Dieses Dokument

Der Client MUSS die Protokoll-Version senden, die im Client implementiert ist. Der Server MUSS per Default den Fehler 511 setzen, wenn eine höhere Version im Request enthalten ist, als im Server implementiert ist. Wenn ältere Serverversionen in bestimmten Fällen dennoch die höhere Clientversion unterstützen, wird im Server manuell eine Ausnahmeregel für den Client definiert, um die höhere Version zu verarbeiten.

Die erste Stelle der Version gibt die Hauptversion an, nach dem Punkt folgt die Unterversion. Unterversionen des Protokolls sind aufwärtskompatibel.

² Dadurch wird eine Rückwärtskompatibilität mit Anwendungsportalen erreicht, die die zusätzlichen Parameter des Application Chainings nicht protokollieren.

Beschreibung der Parameter

Name	Wert (einzeilig, Werte in [] optional)
Version	"1.0" "1.1" "1.2" "1.8" (die vom Client implementierte PVP-Version)
AUTHENTICATE-..	
participantId	Org-ID des Portalverbund-Teilnehmers, bei dem der Benutzer registriert ist. Siehe auch Kapitel 8 „Zertifikate“
gvOuDomain	<i>veraltet! Ersetzt durch participantId</i> Organisations-Domäne des Benutzers. Entweder Internet-Domäne (z.B. magwien.gv.at) oder LDAP: Domain/dn bei System-Principals: Organisationsdomäne des Anwendungsverantwortlichen
UserID	UserID, mit der der Benutzer am Stammportal authentifiziert ist. (LDAP: gvOrgPerson/uid) oder abgekürzte Bezeichnung des System-Principals in der Form Anwendung.Subsystem.
cn	Name des Benutzers (LDAP: gvOrgPerson/cn) oder des System-Principals in der Form Anwendung.Subsystem
gvGid	Global Identifier des Benutzers LDAP: gvOrgPerson/gvGid
gvOuid	Stammdienststelle: Eindeutige Kennung für die Organisationseinheit des Benutzers (LDAP: gvOrgUnit/gvOuid) bei System-Principals: Organisationseinheit des Anwendungsverantwortlichen ³
Ou	Stammdienststelle: Verwaltungskennzeichen [VKZ] der mit AUTHENTICATE-gvOuid bezeichneten Organisationseinheit (LDAP: gvOrgUnit/ou). Bei System-Principals: Organisationseinheit des Anwendungsverantwortlichen
gvFunction	Entspricht Funktion in gvPersonFunction. Verpflichtend, wenn für eine Person Funktionen definiert sind. LDAP: gvPersonFunction/gvFunction

³ Zur Begriffsdefinition siehe „Portalverbundvereinbarung“

	Sicherheitsstufe des Benutzers nach [SecClass] Fehlt dieser Header, wird die Sicherheitsklasse „1“ angenommen. ⁴
AUTHORIZE- ...	
gvOuid	Auftraggebende Dienststelle: Eindeutige Kennung für die Organisationseinheit des Benutzers (LDAP: gvOrgUnit/gvOuid) bei System-Principals: Organisationseinheit des Anwendungsverantwortlichen ⁵
Ou	Auftraggebende Dienststelle: Verwaltungskennzeichen [VKZ] der mit AUTHENTICATE-gvOuid bezeichneten Organisationseinheit (LDAP: gvOrgUnit/ou). Bei System-Principals: Organisationseinheit des Anwendungsverantwortlichen
roles:	Anwendungsrechte, optional mit Rechte-Parametern. LDAP: gvApplicationRight/cn cn (Rechte) und gvUserRestriction (Rechte-Parameter)

⁴ Ausnahme: Beim Server portal.bmi.gv.at ist der Default-Wert 2, um Rückwärtskompatibilität mit bestehenden Anwendungen zu gewährleisten.

⁵ Zur Begriffsdefinition siehe „Portalverbundvereinbarung“

ACCOUNTING- ...	
InvoiceRecptId:	Org-ID des Rechnungsempfängers , zur Definition siehe [VKZ]
CostCenterId:	Liste der für den Benutzer vorgegebenen Kostenstellencodes. Beispiele: ABC123<default>,DEF456 Der Benutzer hat die Kostenstellen ABC123 und DEF456 zur Auswahl, wobei ABC123 der Vorgabewert ist. ABC123 Der Benutzer hat die Kostenstelle ABC123 fix vorgegeben. ABC123<default>, DEF456,<user defined> Der Benutzer hat die Kostenstellen ABC123 und DEF456 zur Auswahl, wobei ABC123 der Defaultwert ist. Außerdem kann er weitere Kostestellen frei eingeben.
ChargeCode:	Liste der für den Benutzer vorgegebenen Codes für Transaktionsgebühr, wobei 0 gebührenfrei bedeutet. Beispiele: 1 Der Benutzer hat die Gebührenstufe in der Anwendung fix vorgegeben 0<default>,1 Der Benutzer hat die Gebührenstufe 0 in der Anwendung vorgegeben, kann aber (über eine Auswahlliste) auch den Wert 1 eingeben.

- Die mitgelieferten Benutzerdaten SOLLEN vom Anwendungsportal protokolliert werden, und es SOLL überprüft werden, ob die Rechte des Benutzers in der Menge der für die Organisation gültigen Rechte enthalten ist.
- Die Verrechnungsparameter CostCenterId und ChargeCode werden der Anwendung als Vorgabewerte übergeben. Bei Mehrfachwerten kann dem Benutzer eine Auswahl angeboten werden.
- Wenn für eine Anwendung bereits eine andere Verrechnungsart spezifiziert ist, als unter pvp-Accounting oben definiert, wird für die Anwendung eine Übergangsfrist verlautbart, bis die hier definierte Verrechnungsart verbindlich ist.
- Beim Application Chaining dient die Übergabe der Parameter nicht der Prüfung von Rechten während der Verarbeitung, sondern zur Protokollierung, um später die datenschutzrechtlich verantwortliche Person feststellen zu können.

Bei den vorangegangenen Stufen im Application Chaining reicht es, folgende Parameter zu übergeben:

- VERSION
- AUTHENTICATE-participantId (wenn innerhalb der Kette unterschiedlich)
- AUTHENTICATE-userId
- AUTHENTICATE-gvGid (bei User Principals)
- AUTHENTICATE-cn
- AUTHENTICATE-gvOuId
- AUTHENTICATE-ou
- AUTHORIZE-roles
- AUTHORIZE-cn
- AUTHORIZE-gvOuId

6. Protokollbindung HTTP

In diesem Abschnitt wird definiert, wie das PVP an das HTTP-Protokoll [RFC2616] gebunden wird.

- Die PVP-Parameter werden über benutzerdefinierte HTTP-Header mitgegeben.
- Die Namen der HTTP-Header werden mit dem Präfix X- (für benutzer-definierte Header) versehen.
- Bei Trennzeichen ",;()=" in den Werten der HTTP-Header (z.B. X-AUTHORIZE-roles) SOLLTE Whitespace vermieden werden, KANN aber vorkommen.
- HTTP MUSS mit TLS oder SSL3.0 gesichert werden, wobei Client-Zertifikate verpflichtend sind.
- Wenn die Verrechnungsdaten vom Stamm- oder Anwendungsportal protokolliert werden sollen, muss die Anwendung die vom Benutzer eingegebenen Werte als die Cookies `x-gvCostCenterId` und `x-gvChargeCode` übergeben, damit sie für das Anwendungsportal lesbar sind. Die Cookies bleiben nur für die Dauer einer HTTP-Transaktion erhalten.
- Jede HTTP-Transaktion wird für sich authentifiziert, da das HTTP-Protokoll stateless ist. Ein Session-Ticket Mechanismus wie bei Kerberos ist derzeit nicht vorgesehen.⁶
- Der Fehlercode wird als HTTP-Code zurück gegeben

7. Protokoll-Bindung PVP – SOAP

Die Protokollbindung für SOAP verwendet und erweitert die Spezifikation Web Services Security [WS-Security].

7.1. Namespaces

Folgende Namespaces werden in diesem Dokument verwendet:

Prefix	Namespace
P	http://egov.gv.at/pvp1
S	http://www.w3.org/2001/12/soap-envelope
wsse	http://schemas.xmlsoap.org/ws/2002/04/secext

7.2. Portalverbund pvpToken

Das `<wsse:security>` Element ist ein Header-Block, um sicherheitsrelevante Informationen zur Nachricht zu übergeben. Es wird zur Übergabe der PVP-Parameter

⁶ Um keinen Performance-Nachteil zu erhalten, wird server-seitig ein Caching der Authentifizierungs-Transaktion empfohlen.

an ein Anwendungsportal um das Element <P:pvToken> erweitert. Die in 5. Grammatik des Portalverbund-Protokolls definierten Bereiche der Keys sind im Gegensatz zur HTTP-Bindung nicht im Namen jedes Elements enthalten, sondern werden zur hierarchischen Gliederung der Elemente in <authorize>, <authenticate> und <accounting> verwendet. Die Struktur ist im [PVP-Schema] beschrieben.

7.3. Fehlermeldungen

PVP-Fehlercodes werden als SOAP Faults zurückgegeben.

Namespace= *urn:ZMRServices*

Der Faultcode setzt sich aus den Zeichen "F" und dem Fehlercode (siehe 9 Fehlermeldungen) zusammen.

Zusätzlich werden für die SOAP-Bindung folgende Fehlermeldungen definiert:

Faultcode	Faultstring	Beschreibung
F480	Invalid SOAP Header	Kein gültiger SOAP-Header gefunden
F481	Missing Security (WS-Security) token in SOAP-Header	Die PVP-Parameter müssen in einem WSSE:Security-Token eingebettet sein

8. Zertifikate

Im Portalverbundsystem sind Verwaltungszertifikate oder kommerzielle Zertifikate registrierter ZDAs zu verwenden. Das Zertifikat identifiziert den Stammportalbetreiber.

Ab PVP-Version 1.8 werden Teilnehmer durch den PVP-Parameter participantId gekennzeichnet. Anwendungsportale SOLLEN jedoch Stammportale mit vorhergehenden PVP-Versionen unterstützen, wo der Teilnehmer identisch mit dem Stammportalbetreiber ist.

9. Fehlermeldungen

Fehler-Code	Beschreibung
402	Für diese Funktion ist eine Verrechnung erforderlich, aber das Header-Feld XXXX fehlt (XXXX ist eines aus ACCOUNTING-gvInvoiceRecptId, ACCOUNTING-gvCostCenterId oder ACCOUNTING-gvChargeCode)
440	Mandatory PVP-Header XXXX fehlt
441	Werte in AUTHORIZE-roles haben ungültiges Format
442	Kein zulässiges Recht in AUTHORIZE-roles
450	ACCOUNTING-gvInvoiceRecptId: ungültiger Wert oder Verrechnungskonto gesperrt
451	Ungültiger Wert für ACCOUNTING-gvChargeCode
461	Sicherheitsklasse (gvSecClass) muss mindestens 1 sein
462	Sicherheitsklasse (gvSecClass) muss mindestens 2 sein
463	Sicherheitsklasse (gvSecClass) muss 3 sein
490	Zertifikatsüberprüfung fehlgeschlagen. Grund: XXXXXXXXXXXXXXXXXXXX (z.B.: ungültige Root-CA, Zertifikat abgelaufen, Zertifikat nicht beim Portal registriert)
491	HTTP wird nicht unterstützt – es muss HTTPS verwendet werden
492	Keine Berechtigung für diese Anwendung im Anwendungsportal definiert
511	PVP-Version nicht unterstützt

Fehlerbedingungen sind im Text möglichst detailliert zu beschreiben, etwa durch die Referenz des betroffenen Headers und die Art der Bedingung (z.B. „Header `Version` fehlt“, „Wert für `gvSecClass` zu groß“)

Anhang A HTTP Beispiel-Request User Principal

Beispiel für einen HTTP Header bei einem Request eines Stammportals ohne Verrechnungsdaten. Die Organisation wird über eine OuDomain identifiziert:

```
POST /abc.gv.at/portal/servlet/ HTTP/1.1
Host: portal.abc.gv.at
Accept-Encoding: gzip, deflate
User-Agent: Mozilla (5.0 Linux)
Connection: close
X-Version: 1.8
X-AUTHENTICATE-participantId: AT:L6:1234789
X-AUTHENTICATE-UserId: 4711240761@gemeinden.stmk.gv.at
X-AUTHENTICATE-cn: Max Mustermann
X-AUTHENTICATE-gvGid: 4711240761
X-AUTHENTICATE-gvOuDomain: beispielgemeinden.stmk.gv.at
X-AUTHENTICATE-gvOuId: AT:L9:12345
X-AUTHENTICATE-Ou: GGA-60477
X-AUTHENTICATE-gvFunction: SB
X-AUTHENTICATE-gvSecClass: 2
X-AUTHORIZE-roles: Beispielrolle(GKZ=60477,GKZ=60479);
Content-Type: application/x-www-form-urlencoded
Content-Length: 788
```

In diesem Fall ist der Benutzer berechtigt, das Recht *Beispielrecht* für die Gemeinden 60477 und 60479 auszuüben.

Anhang B HTTP Beispiel-Request System Principal

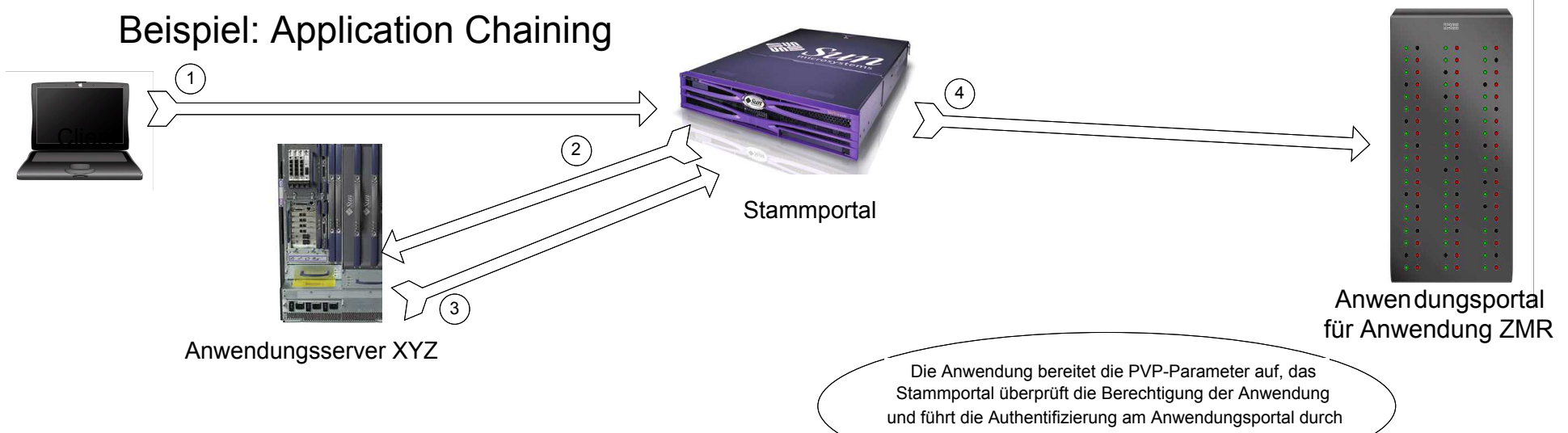
Die Organisation wird in dem Beispiel über eine Org-Id identifiziert:

```
POST /bmi.gv.at/portal/xyz HTTP/1.1
Host: portal.bmi.gv.at
User-Agent: JSSE 1.0.1
X-Version: 1.8
X-AUTHENTICATE-participantId: AT:L9:MA2412
X-AUTHENTICATE-UserId: omr-appuser@wien.gv.at
X-AUTHENTICATE-cn: OMR
X-AUTHENTICATE-gvOuId: AT:L9:12345
X-AUTHENTICATE-Ou: GGA-90101
X-AUTHENTICATE-gvSecClass: 2
X-AUTHORIZE-roles: Beispielrolle(GKZ=60477,GKZ=60479);
Content-Type: text/xml
Content-Length: 788
```

Anhang C Beispiel für Application Chaining

(siehe nächste Seite. Anmerkung: Im Beispiel wurde der Parameter X-AUTHENTICATE-gvGid ausgelassen, was nicht korrekt ist)

Beispiel: Application Chaining



Übergabe der PVP-Parameter

Browser an Stammportal ①	Stammportal an Anwendung XYZ ②	Anwendung XYZ an Anwendung ZMR ③④
<pre>User-Agent: Mozilla/5.0 (Win... Content-Type: application/ x-www-form-urlencoded Content-Length: 789</pre> <p>Beim Application Chaining werden Benutzer und Rolle zur Protokollierung mit einem Präfix versehen an die nächste Anwendung weitergegeben</p>	<pre>User-Agent: Mozilla/5.0 (Windows; U; ... X-Version: 1.8 X-AUTHENTICATE-ParticipantId: AT:L6:13 X-AUTHENTICATE-UserId: fxmeier@stmk.gv.at X-AUTHENTICATE-cn: Franz X. Meier X-AUTHENTICATE-gvOuid: AT:L6:123 X-AUTHENTICATE-Ou: L6AL-FA1B X-AUTHORIZE-roles: XYZ-Sachbearbeiter X-GVOPZONE: Prod X-AUTHENTICATE-gvSecClass: 1 Content-Type: application/x-www-form-urlencoded Content-Length: 789</pre>	<pre>User-Agent: JAIK-JCE2.6.1/JSSE 1.0.1 X-Version: 1.8 X-01-Version: 1.8 X-AUTHENTICATE-ParticipantId: AT:L6:13 X-01-AUTHENTICATE-UserId: fxmeier@stmk.gv.at X-01-AUTHENTICATE-cn: Franz X. Meier X-01-AUTHENTICATE-gvOuid: AT:L6:123 X-01-AUTHENTICATE-Ou: L6AL-FA1B X-01-AUTHORIZE-roles: XYZ-Sachbearbeiter(); X-AUTHENTICATE-UserId: apu-1@stmk.gv.at X-AUTHENTICATE-cn: ApplicationUser Apu1 X-AUTHENTICATE-gvOuid: AT:L6:123 X-AUTHENTICATE-Ou: L6AL-FA1B X-AUTHORIZE-roles: ZMR-Behoerdenabfrage X-GVOPZONE: Prod X-AUTHENTICATE-gvSecClass: 2 Content-Type: text/xml Content-Length: 2197</pre>

Anhang D SOAP Beispiel-Request

```
<S:Envelope
  xmlns:P="http://portal.bmi.gv.at/ref/pvp1.xsd"
  xmlns:S="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:wsa="http://schemas.xmlsoap.org/ws/2002/03/addressing"
  xmlns:wsse="http://schemas.xmlsoap.org/ws/2002/04/secext"
<S:Header>
  <wsse:Security>
    <P:pvpToken version="1.8">
      <P:authenticate>
        <P:participantId>AT:L6:994</P:participantId>
        <userPrincipal>
          <userId>fmeier@stmk.gv.at</userId>
          <cn>F. Meier</cn>
          <gvOuId>AT:L6:1299</gvOuId>
          <ou>L6AL-F2/c</ou>
          <gvSecClass>2</gvSecClass>
          <gvGid>Uh05RG++kla0TsVY+CU=</gvGid>
          <gvFunction>SB</gvFunction>
        </userPrincipal>
      </P:authenticate>
      <P:authorize>
        <P:role value="ZMR-Fremdenbehoerdenanfrage">
          <P:param>
            <P:key>GKZ</P:key>
            <P:value>60100</P:value>
          </P:param>
        </P:role>
      </P:authorize>
    </P:pvpToken>
  </wsse:Security>
</S:Header>
<S:Body>
  . . .
</S:Body>
</S:Envelope>
```

Anhang E Beispiele für Rechte und Rechteparameter

Durch die zweidimensionale Darstellung von Rechten können die meisten Berechtigungssysteme mit angemessenem Aufwand abgebildet werden. Rechte an Anwendungen sind hier im Allgemeinen bereits aggregierte Einzelrechte, die in der Literatur oft als Rollen bezeichnet werden.

9.1.1.Einfaches Berechtigungsschema

Recht 1: Anwendungsadministrator

Recht 2: Sachbearbeiter

Recht 3: Abfrageberechtigter

9.1.2.Komplexeres Berechtigungsschema

Recht 1: Anwendungsadministrator

Recht 2: Sachbearbeiter (OE)

Recht 3: Abfrageberechtigter

In diesem Fall dürfen Sachbearbeiter nur für bestimmte Organisationseinheiten Geschäftsfälle erledigen. Dafür wird eine Liste von Oes übergeben, für die der Benutzer die Rechte hat.

Das Modell ist auch für andere Einschränkungen oder explizite Berechtigungen einer Rolle anwendbar, etwa nach geografischen Gesichtspunkten.

Anhang F Implementierungshinweise

a. Betrieb von Anwendungsportalen als Reverse Proxy

Das Anwendungsportal KANN als Reverse Proxy mit mehreren Anwendungen an einem Virtuellen Host betrieben werden. Dadurch wird die Zertifikatsverwaltung vereinfacht. Aus dieser Architektur ergeben sich folgende Konsequenzen:

- Jeder Anwendung einen URL-Raum zugewiesen, sodass aus dem URL eindeutig die Anwendung abgeleitet werden kann.
- Anwendungen sollten Ressourcen nur über relative Pfade adressieren.
- Cookies mit Domain-Parametern müssen korrekt behandelt werden⁷
- Cookies aller Anwendungen des Stammportals haben einen gemeinsamen Namensraum. Sollte das zu Problemen⁸ führen, können zusätzliche Virtual Hosts die Namensräume trennen.

b. Betrieb von Stammportalen als Reverse Proxy

Ein Stammportal KANN als Reverse Proxy betrieben werden.

Im Pfad wird die Domäne des Anwendungsportals wiederholt, um eindeutige Pfade zu gewährleisten. Dadurch kann ein Stammportal als Reverse Proxy mit einem Virtuellen Host betrieben werden, auch wenn Zugriffe auf unterschiedliche Anwendungsportale erfolgen.

Z.B. : <https://portal.org-b.gv.at/org-a.gv.at/app1/>

c. Verifikation von Client-Zertifikaten

Bei der Implementierung von Servern ist darauf zu achten, dass für TLS-Client-Zertifikate eine Hostname-Verifikation durchgeführt wird.

⁷ Der Domain-Name beim Response auf den des Portals, und beim Request auf den der Anwendung umgesetzt werden.

⁸ Namensraumkonflikte können z.B. entstehen, wenn 2 Java-Anwendungen JSESSIONID Cookies verwenden, und im gleichen Application Server ausgeführt werden, und dadurch der Cookie-Namen nicht getrennt geändert werden kann. Benutzer, die zwischen den Anwendungen wechseln, können die Sessions durcheinander bringen.

Anhang G Referenzen

[PortalV-PKI]

<http://portal.bmi.gv.at/ref/> -> PKI

[PV-Whitepaper]

Hörbe, Rainer: Portal Verbund Whitepaper 2002-02-28

<http://reference.e-government.gv.at> -> Portalverbund

[PVP-Schema]

Hörbe, Liehmann, Martin: XML-Schema für den Protalverbund: pvp1.xsd
(2004-07-01)

<http://reference.e-government.gv.at> -> Portalverbund

[PVV 1.0]

Connert, Grandits, Kotschy, Posch, Siegl: Vereinbarung über die einzuhaltenden Rahmenbedingungen bei der Einrichtung und Benützung eines E-Government Portalverbundsystems (21.11.2002)

<http://reference.e-government.gv.at> - Empfehlungen

[RFC2616]

R. Fielding & al.: Hypertext Transfer Protocol -- HTTP/1.1

<ftp://ftp.isi.edu/in-notes/rfc2616.txt>

[SecClass]

Hörbe, Rainer: Sicherheitsklassen im Portalverbund-System, 15.10.2002

<http://reference.e-government.gv.at> – Empfehlungen

[PVP-Schema]

[VKZ]

Grandits, Franz: Verwaltungskennzeichen:

<http://reference.e-government.gv.at>. Dokument VKZ 1.1.0 (Entwurf vom 15.5.2003)

[WS-Security]

Specification: Web Services Security, Version 1.0 05 April 2002

<http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0.pdf>

Anhang H Funktionelle Änderungen von Version 1.7 zu 1.8

- Unterscheidung zwischen natürlicher Person und System als Benutzer
- Erweiterung für Application Chaining
- Einführung der SOAP-Bindung
- Definition, wie die Versionsnummer zu verwenden ist. In diesem Zusammenhang wurde auch die Fehlernummer 511 eingeführt
- Stammportale sind nun mandantenfähig, ohne pro Mandant ein separates Zertifikat zu benötigen. Statt dessen wird der Parameter participantId verwendet.
- X-AUTHENTICATE-gvOuDomain ist optional, wenn in X-AUTHENTICATE-participantID die Org-Id laut [VKZ] übergeben wird
- Abschnitt Zertifikate wurde geändert
- X-AUTHENTICATE-ou ist mit der Verfügbarkeit des VKZ auch mit diesem zu befüllen, nicht mehr mit dem cn der OE. X-AUTHENTICATE-gvOuid ist nicht mehr optional
- Änderung der Bezeichnung "RegionalRestrictions" in Rollenparameter. Erweiterung der Grammatik: Auz-Parameter kann beliebige CHAR-Werte enthalten
- Korrektur der Grammtik des Parameters "Version"
- Anforderungen an Reverse Proxies definiert
- Redefinition des Begriffs „Verrechnungsdaten“ und Erläuterung von CostCenterId und ChargeCode, Korrektur der Grammatik
- Überarbeitung des Abschnitts „Zweck“
- Fußnote zu X-AUTHENTICATE-gvSecClass: Der Satz „Später angeschaltete Portale müssen den Header mitliefern.“ wurde gestrichen, da er im Widerspruch dazu steht, dass der Parameter optional ist.