OSCI-Transport 1.2

– Korrigenda –

Status: FINAL

OSCI Leitstelle

Bremen, den 10. Juni 2004

Inhaltsverzeichnis

1 Einleitung

Seit der Veröffentlichung der OSCI-Transport 1.2 - Spezifikation wurde eine Reihe von Änderungsvorschlägen unterbreitet. Qualitativ betreffen diese Vorschläge kleinere formale Fehler bis hin zu inhaltlichen Änderungen, die die Abläufe der Szenarien verändern. Diese resultieren zum Teil aus den Erfahrungen bei der Erstellung der Referenzimplementierung, zum Teil wurden sie von den Teilnehmern des Beta-Tests eingebracht.

Diese Korrigenda berücksichtigen Änderungen nur insoweit, wie sie für die praktische Handhabung von OSCI-Transport 1.2 notwendig und sinnvoll und im Rahmen einer Version noch vertretbar erscheinen. Vorschläge, die eine grundlegende Änderung der Struktur der OSCI-Nachrichten zum Gegenstand haben (z.B. SOAP-freies OSCI, s. [DISS 2004]), wurden nicht betrachtet.

Bezüglich der Validierbarkeit der verwendeten Schemata existiert ein bekanntes Problem, das im Rahmen dieser Korrigenda nicht gelöst werden kann. Dieses Problem betrifft Namensraumkollisionen, die sich aus den Ausprägungen von XML-Encryption und XML-Signature ergeben.

Folgende Firmen und Institutionen waren an der Erstellung dieser Korrigenda beteiligt:

- bremen online services GmbH & Co. KG, Bremen
- OSCI Leitstelle, Bremen (Herausgeber)

Weitere Informationen finden Sie im Internet unter der Adresse http://www.osci.de

1.1 Copyright

Die vorliegenden Korrigenda der OSCI-Transport 1.2 - Spezifikation wurden im Auftrag der OSCI-Leitstelle als Herausgeber entwickelt.

Diese Korrigenda sind urheberrechtlich geschützt. Alle Nutzungsrechte liegen bei dem Herausgeber. Herstellern wird zur Implementation von Bürger-, Kommunal-, Intermediär- oder Dienstleistersystemen unentgeltlich ein einfaches Nutzungsrecht eingeräumt. Im Rahmen des genannten Zwecks darf dieses Dokument in unveränderter Form vervielfältigt und zu den nachstehenden Bedingungen verbreitet werden.

Umgestaltungen, Bearbeitungen, Übersetzungen und jegliche Änderungen sind nur nach Rücksprache mit dem Herausgeber zulässig. Kennzeichnungen, Copyright-Vermerke und Eigentumsangaben sind beizubehalten.

Haftung für Mängel dieses Dokuments wird nur bei Vorsatz und grober Fahrlässigkeit übernommen. Hersteller der oben genannten Systeme sind gebeten, Fehler, Unklarheiten oder Interpretationsfreiräume dieser Spezifikation, die die ordnungsgemäße Funktion oder die Interoperabilität behindern, dem Herausgeber zu melden.

Eine Weitergabe dieses Dokuments an Dritte darf nur unentgeltlich, in unveränderter Form und zu den vorstehenden Bedingungen erfolgen.

1.2 Konventionen zur Textauszeichnung

Für diese Korrigenda gelten die gleichen Konventionen wie für die zugrunde liegende Spezifikation:

- Normative Absätze sind hellgrau unterlegt. Beispiel:

  Dieser Absatz ist normativ.

In Zweifelsfällen gelten die Festlegungen in Schemas dieser Spezifikation *vor* normativen Textpassagen dieser Spezifikation. Diese gelten wiederum *vor* normativen Teilen referenzierter Dokumente und diese schließlich *vor* nicht normativen Teilen dieser Spezifikation.

- Änderungen der Korrigenda zur Spezifikation sind innerhalb der normativen Textpassagen **fett** gesetzt.
- Jede Art von Code ist in `Schreibmaschinenschrift` gesetzt.

2 Korrekturen

Im Folgenden werden die einzelnen Korrekturen detailliert beschrieben.

2.1 Request-Response-Szenario, passiver Empfänger, Protokollierung

Der Ablauf des Szenarios sieht vor, dass der Benutzer 2 für den Nachrichtenrückweg eine MessageId vom Intermediär abruft. Dies ist nicht nur aufwändig, sondern auch nur dann überhaupt möglich, wenn der Benutzer 2 die URL des Intermediärs kennt. Es erscheint sinnvoll, dass der Intermediär diese MessageId dem Bearbeitungsauftrag hinzufügt. Der Ablauf des Szenarios in Abschnitt 3.5.3 würde sich wie folgt vereinfachen (Änderungen in **Fettschrift**):

1. Benutzer 1 sendet einen MessageId-Anforderungsauftrag an den Intermediär.
2. Der Intermediär erzeugt eine neue MessageId und sendet sie in einer MessageId-Anforderungsantwort an Benutzer 1.
3. Benutzer 1 sendet einen Dialoginitialisierungsauftrag an den Intermediär.
4. Der Intermediär reagiert mit einer Dialoginitialisierungsantwort an Benutzer 1.
5. Benutzer 1 sendet einen Abwicklungsauftrag mit Zustellung 1 an den Intermediär.
6. Der Intermediär erzeugt einen Laufzettel 1 zu Zustellung 1 und protokolliert auf diesem den Empfang. Er sendet einen Bearbeitungsauftrag mit Zustellung 1 **und einer neuen MessageId für die Bearbeitungsantwort** an Benutzer 2 und protokolliert die Weiterleitung von Zustellung 1 auf Laufzettel 1.
7. ***[entfällt]***
8. ***[entfällt]***
9. Benutzer 2 sendet eine Bearbeitungsantwort mit Zustellung 2 an den Intermediär.
10. Der Intermediär erzeugt einen Laufzettel 2 zu Zustellung 2 und protokolliert auf diesem den Empfang. Weiterhin protokolliert er den Empfang von Zustellung 1 durch Benutzer 2 auf Laufzettel 1. Er sendet eine Abwicklungsantwort mit Zustellung 2 an Benutzer 1.
11. Benutzer 1 sendet einen Dialogendeauftrag an den Intermediär.
12. Der Intermediär protokolliert den Empfang von Zustellung 2 durch Benutzer 2 auf Laufzettel 2 und sendet eine Dialogendeantwort an Benutzer 1.

Das Schema des Bearbeitungsauftrags muss um die MessageId ergänzt werden. Diese wird der `processDeliveryType`-Definition hinzugefügt:

```
<xsd:complexType name="processDeliveryType">
  <xsd:complexContent>
    <xsd:extension base="osci:DefaultHeaderBlockTemplate">
      <xsd:sequence>
          <xsd:element name="MessageIdResponse" type="osci:MessageIdType"
          minOccurs="0" />
          <xsd:element name="ProcessCardBundle"
          type="osci:ProcessCardBundleType" minOccurs="0" />
```

```
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

2.2 Verschlüsselung

Im normativen Teil des Abschnitts 4.2 wird gefordert, dass Daten für verschiedene Adressaten mit verschiedenen Sitzungsschlüsseln verschlüsselt werden. Mit „Adressaten“ sind hier z.B. nicht Leser im Sinne des OSCI-Rollenmodells gemeint. Die übliche Vorgehensweise, einen Sitzungsschlüssel für mehrere Empfänger zu verschlüsseln, ist zulässig. Der betreffende Absatz wird wie folgt präzisiert:

Es ist sicherzustellen, dass für **verschiedene** Daten, die an unterschiedliche Adressaten gehen, nicht derselbe Sitzungsschlüssel verwendet wird.

Die Verschlüsselung des Sitzungsschlüssels erfolgt mittels RSAES-PKCS1-v1_5 [PKCS 1] (Algorithmus-Identifier: `http://www.w3.org/2001/04/xmlenc#rsa-1_5` [XENC, Abschnitt 5.4.1]). Die Modullänge des verwendeten RSA-Schlüsselpaares muss dabei mindestens 1024 Bit betragen.

2.3 Zertifikatsprüfungen

Im normativen Teil des Abschnitts 4.4 wird für die Online-Gültigkeitsprüfung lediglich die Überprüfung gefordert, ob ein Zertifikat widerrufen wurde. Darüber hinaus muss jedoch auch geprüft werden, ob das Zertifikat dem Aussteller bekannt ist und von ihm ausgestellt wurde. Der betreffende normative Absatz wird daher wie folgt ergänzt:

Die folgenden Prüfungen sind jeweils vorzunehmen:

- Mathematische Prüfung der Signatur des Zertifikats: Der Hashwert über das Zertifikat wird erneut berechnet und formatiert. Das Ergebnis muss mit der entschlüsselten Signatur des Zertifikats übereinstimmen.
- Offline-Gültigkeitsprüfung: Geprüft wird, dass der Zeitpunkt der Prüfung innerhalb des Gültigkeitszeitraums liegt, der im Zertifikat angegeben ist.
- Online-Gültigkeitsprüfung: Geprüft wird, dass das Zertifikat **vom Aussteller ausgestellt und** zum Zeitpunkt der Prüfung nicht widerrufen ist.

Entsprechend muss die Schemadefinition dahingehend geändert werden, dass Kombinationen der verschiedenen Prüfverfahren (z.B. CRL/LDAP) im `OnlineResult`-Eintrag dargestellt werden können (Abschnitt 6.3). Um beliebige Kombinationen zu ermöglichen, wird die Definition wie folgt geändert:

```
<xsd:complexType name="OnlineResultType">
  <xsd:sequence>
    <xsd:element name="OSCP" type="xsd:base64Binary" minOccurs="0" />
    <xsd:element name="CRL" type="xsd:dateTime" minOccurs="0" />
    <xsd:element name="LDAP" type="xsd:base64Binary" minOccurs="0" />
    </xsd:element>
  </xsd:sequence>
  <xsd:attribute name="Result" use="required">
    <xsd:simpleType>
      <xsd:restriction base="xsd:string">
        <xsd:enumeration value="ok" />
        <xsd:enumeration value="revoked" />
```

```
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:attribute>
</xsd:complexType>
```

Zusätzlich erhalten die Definitionen der `MathResult`- und `OfflineResultType`-Elemente (Abschnitt 6.3) einen möglichen Attributwert „indeterminate“ für den Fall, dass die betreffende Prüfung aus irgendeinem Grund nicht durchgeführt werden konnte:

```
<xsd:complexType name="MathResultType">
  <xsd:attribute name="Result" use="required">
    <xsd:simpleType>
      <xsd:restriction base="xsd:string">
        <xsd:enumeration value="ok" />
        <xsd:enumeration value="corrupted" />
        <xsd:enumeration value="indeterminate" />
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:attribute>
</xsd:complexType>

<xsd:complexType name="OfflineResultType">
  <xsd:attribute name="Result" use="required">
    <xsd:simpleType>
      <xsd:restriction base="xsd:string">
        <xsd:enumeration value="valid" />
        <xsd:enumeration value="invalid" />
        <xsd:enumeration value="indeterminate" />
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:attribute>
</xsd:complexType>
```

Die Definition des `CertType`-Elements (Abschnitt 6.3) erhält einen zusätzlichen möglichen Attributwert „accredited“ für Zertifikate akkreditierter Herausgeber:

```
<xsd:complexType name="CertTypeType">
  <xsd:attribute name="Type" use="required">
    <xsd:simpleType>
      <xsd:restriction base="xsd:string">
        <xsd:enumeration value="advanced" />
        <xsd:enumeration value="qualified" />
        <xsd:enumeration value="accredited" />
        <xsd:enumeration value="unknown" />
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:attribute>
</xsd:complexType>
```

Um die Zuordnung von Zertifikaten zu erleichtern, wird schließlich die `InspectionType`-Definition (ebenfalls Abschnitt 6.3) um das Subject des Zertifikats ergänzt:

```
<xsd:complexType name="InspectionType">
  <xsd:sequence>
    <xsd:element name="Timestamp" type="osci:TimestampType" />
    <xsd:element name="X509SubjectName" type="xsd:string" />
    <xsd:element name="X509IssuerName" type="xsd:string" />
    <xsd:element name="X509SerialNumber" type="xsd:integer" />
    <xsd:element name="CertType" type="osci:CertTypeType" />
    <xsd:element name="MathResult" type="osci:MathResultType" />
    <xsd:element name="OfflineResult" type="osci:OfflineResultType" />
    <xsd:element name="OnlineResult" type="osci:OnlineResultType"
     minOccurs="0" />
  </xsd:sequence>
</xsd:complexType>
```

2.4 Reaktionsvorschriften

Der normative Abschnitt in Abschnitt 5 wird um folgenden Satz ergänzt:

Ist die Auftragsnachricht verschlüsselt, so muss der Supplier die Auftragsantwort ebenfalls verschlüsseln. Ist die Auftragsnachricht signiert, so muss der Supplier die Auftragsantwort ebenfalls signieren.

Entsprechend wird die Tabelle im Abschnitt 5.2 um die folgende Warnung ergänzt:

Schritt	Situation	osci:Code
8	Der Empfänger hat eine unverschlüsselte Auftragsantwort gesendet	3803

2.5 Rückmeldungen bezüglich des Signaturzertifikats des Empfängers

Die Tabelle im Abschnitt 5.2 enthält keine Rückmeldungen für die Prüfung des Signaturzertifikats des Empfängers. Die Tabelle wird um die folgenden Zeilen ergänzt:

Schritt	Situation	osci:Code
7	Signierzertifikat des Empfängers ist zeitlich ungültig	3708
7	Signatur über das Signierzert ifikat des Empfängers ist fehlerhaft	9711
7	Signierzertifikat des Empfängers ist widerrufen	9712

2.6 Schema: Fehlender Addressee-Signaturzertifikatseintrag in der `NonIntermediaryCertificatesTemplate`-Definition

Die Definition des `NonIntermediaryCertificatesTemplate` (Abschnitt 6.3) muss lauten:

```
<xsd:complexType name="NonIntermediaryCertificatesTemplate"
 abstract="true">
  <xsd:complexContent>
    <xsd:extension base="osci:GeneralHeaderBlockTemplate">
      <xsd:sequence>
        <xsd:element name="CipherCertificateOriginator"
         type="osci:CertificateType" minOccurs="0" />
        <xsd:element name="CipherCertificateOtherAuthor"
         type="osci:CertificateType"
         minOccurs="0" maxOccurs="unbounded" />
        <xsd:element name="CipherCertificateAddressee"
         type="osci:CertificateType" minOccurs="0" />
        <xsd:element name="CipherCertificateOtherReader"
         type="osci:CertificateType"
         minOccurs="0" maxOccurs="unbounded" />
        <xsd:element name="SignatureCertificateOriginator"
         type="osci:CertificateType" minOccurs="0" />
        <xsd:element name="SignatureCertificateOtherAuthor"
         type="osci:CertificateType"
         minOccurs="0" maxOccurs="unbounded" />
        <xsd:element name="SignatureCertificateAddressee"
         type="osci:CertificateType" minOccurs="0" />
      </xsd:sequence>
      <xsd:attribute ref="soap:actor"
       fixed="http://www.w3.org/2001/12/soap-envelope/actor/none"
       use="required" />
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

2.7 Schema: Ableitung des `DefaultBodyBlockTemplate`-Elements

Die Definition des `DefaultBodyBlockTemplate`-Elements (Abschnitt 6.3) muss lauten:

```
<xsd:complexType name="DefaultBodyBlockTemplate" abstract="true">
    <xsd:complexContent>
      <xsd:extension base="osci:GeneralBodyBlockTemplate">
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
```

2.8 Schema: `ContentContainer`-Elemente

Die Definition des `ContentContainerType`-Elements lässt nur ein `Content`-Element zu. Die Definition wird wie folgt geändert:

```
<xsd:complexType name="ContentContainerType">
  <xsd:sequence>
    <xsd:element ref="ds:Signature"
     minOccurs="0" maxOccurs="unbounded" />
    <xsd:element name="Content" type="osci:ContentType"
     minOccurs="0" maxOccurs="unbounded" />
    <xsd:element ref="xenc:EncryptedData"
     minOccurs="0" maxOccurs="unbounded" />
  </xsd:sequence>
  <xsd:anyAttribute namespace="##any" />
</xsd:complexType>
```

Die zusätzliche Attributdefinition erleichtert die Verwaltung der `ContentContainer`-Elemente.

2.9 Bedeutung des Elements `fetchDelivery`

Das Element `fetchDelivery` zeigt einen Zustellungs**abhol**auftrag an (Abschnitt 6.2).

2.10 Schema: Definition des `ControlBlockType` - Elements der Annahmeantwort

Die Definition des `ControlBlockType` - Elements (Abschnitt 6.6.16) muss lauten:

```
<xsd:complexType name="ControlBlockType">
    <xsd:complexContent>
      <xsd:restriction base="osci:ControlBlockTemplate">
        <xsd:sequence>
          <xsd:element name="Response" type="xsd:string" minOccurs="1" />
          <xsd:element name="Challenge" type="xsd:string"
           minOccurs="0" maxOccurs="0" />
        </xsd:sequence>
        <xsd:attribute name="ConversationId" type="osci:Number"
         use="prohibited" />
        <xsd:attribute name="SequenceNumber" type="osci:Number"
         use="prohibited" />
      </xsd:restriction>
    </xsd:complexContent>
  </xsd:complexType>
```

2.11 Auswahlregeln im Laufzettelabholauftrag

Die Auswahlregeln für den Laufzettelabholauftrag (Abschnitt 6.6.11) werden um Möglichkeiten erweitert, die zurückzugebenden Laufzettel weiter einzugrenzen, nämlich

- Laufzettel für Nachrichten, die noch nie vom Intermediär abgeholt bzw. empfangen wurden,

- Laufzettel für Nachrichten, die an den Absender des Laufzettelabholauftrags als Empfänger (Addressee) gesendet wurden.
- Laufzettel für Nachrichten, die der Absender des Laufzettelabholauftrags als Absender (Originator) versendet hat.

Diese Regeln werden über zwei zusätzliche Attribute der `osci:RecentModification`- und `osci:ReceptionOfDelivery` -Elemente, `Role` und `NoReception` gesteuert:

6. Besitzen die Elemente `osci:RecentModification` bzw. `osci:ReceptionOfDelivery` ein Attribut „`Role`“ mit dem Wert „`Addressee`“, so werden die Auswahlregeln nur auf Laufzettel von Nachrichten angewandt, die an den Absender des Laufzettelabholauftrags als Empfänger gerichtet wurden. Besitzt das Attribut „`Role`“ den Wert „`Originator`“, so werden nur Laufzettel von Nachrichten zurückgegeben, die der Absender des Laufzettelabholauftrags versendet hat.
7. Besitzen die Elemente `osci:RecentModification` bzw. `osci:ReceptionOfDelivery` ein Attribut „`NoReception`“ mit dem Wert „`true`“, so werden nur Laufzettel von Nachrichten zurückgegeben, zu denen keine Empfangsbestätigung des Empfängers vorliegt.

Die Definition des `fetchProcessCardType`-Elements ändert sich hiermit wie folgt:

```
<xsd:complexType name="SelectionDateType">
  <xsd:sequence>
    <xsd:element type="xsd:dateTime" />
  </xsd:sequence>
  <xsd:attribute name="NoReception" type="xsd:boolean" use="optional"/>
  <xsd:attribute name="Role" type="xsd:string" use="optional">
    <xsd:simpleType>
      <xsd:restriction base="xsd:string">
        <xsd:enumeration value="Addressee" />
        <xsd:enumeration value="Originator" />
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:attribute>
</xsd:complexType>

<xsd:complexType name="fetchProcessCardType">
  <xsd:complexContent>
    <xsd:extension base="osci:DefaultBodyBlockTemplate">
      <xsd:sequence>
        <xsd:element name="SelectionRule" minOccurs="0">
          <xsd:complexType>
            <xsd:choice>
              <xsd:element name="ReceptionOfDelivery"
               type="SelectionDateType" />
              <xsd:element name="RecentModification"
               type="SelectionDateType" />
              <xsd:element name="MessageId" type="osci:MessageIdType"
               maxOccurs="unbounded" />
            </xsd:choice>
          </xsd:complexType>
        </xsd:element>
        <xsd:element name="Quantity" minOccurs="0">
          <xsd:complexType>
```

```
            <xsd:attribute name="Limit" type="xsd:positiveInteger"
             use="required" />
          </xsd:complexType>
        </xsd:element>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

3 Literaturverzeichnis

[DISS 2004] F. Hüwel: Validität, Anwendbarkeit und Praktikabilität des Standards OSCI-Transport; Diss. FernUniversität Hagen 2004. Online verfügbar unter http://www.huewel.de/VAP.pdf