Request for comments

Authority public-key distribution protocol

Authors: Tim Moses, Entrust Technologies Corporation,

Ed Simon, Entrust Technologies Corporation,

Please provide comments to Tim Moses at Entrust Technologies, by sending email to tim.moses@entrust.com, placing the text “APD-RFC” in the title..

1. Introduction

This document describes an application-layer protocol by which a subject authority’s public verification key is communicated to a relying party, or an agent representing a community of relying parties, so that those relying parties can verify certificates issued by the subject authority. The transport-layer protocol is unspecified, but the application-layer protocol is most useful when the transport-layer is HTTP. In this case, the subject authority posts its public verification key on an HTTP server and the recipient obtains it using the HTTP “get” method. The application-layer protocol performs a binding between the authority public verification key and the terms and conditions of its use, as specified by the subject authority. The binding is done in such a way that web search engines can locate public verification keys which have been published in conformance with this protocol and which satisfy usage requirements acceptable to the recipient.

Ultimately, policy is defined by communities of relying parties. Therefore, in order to ensure that a subject authority’s offer finds acceptance, in a step that is not described here, it is expected that communities of relying parties will let it be known what constitutes an acceptable policy.

The Authority Public-Key Distribution protocol implements the legal, as well as the technical, requirements for distribution of an authority's public key. In the paper-based world, it is common practice to use a standard template for the legal text, leaving spaces for the specific details of a contract. For example, an EDI trading partner agreement uses generic legal text but leaves spaces within that text so the human parties can write in details, such as the names of the parties, etc..

In the new world of electronic commerce, it becomes possible to automate the negotiation of contracts defined by a standard legal framework. Appropriately, this automation of contract negotiations is called "electronic contracting". As in the paper world, the legal framework is important, but if the legal framework has been standardized, the negotiation becomes simpler because it can focus solely on the variable parts of the contract (e.g.. the offer price).

In the electronic world, XML (together with XSL) allows the static and variable parts of a legal document to be both cleanly separated and re-joined. The separation of the variable information into its own XML instance allows applications to focus on the variable parts of the electronic contract. Yet, if the full legal text (static and variable) is required, an application can simply apply the XSL style sheet (containing the static text) to the XML instance (containing the variable information) to render a complete legal contract.

The protocol described here uses the XSL style-sheet proposal submitted to the W3C on August 27, 1997. XSL was chosen because:

- it can query elements within an XML instance no matter where, in the XML instance, those elements are located; and
- it allows both the querying language and the static text to be contained in a single XSL file.

The XSL style-sheets are a mandatory part of the protocol because it is important that both the offering and accepting parties need not concern themselves with whether the presented legal text really matches that given in this standard. XSL provides an ideal way of encapsulating the static legal text in a format that can be easily processed by XML-aware applications.

2. Protocol overview

There are two considerations that lead to variants of the basic protocol. The first consideration concerns controlling the make-up of the relying party community. Two possibilities are accommodated. In the first possibility, ***any*** recipient is a qualified relying party, and in the second possibility, the subject authority must ***approve*** a recipient before it qualifies to be a relying party.

The second consideration concerns the means by which the relying party achieves the necessary degree of assurance in the certificates issued by the subject authority. Again, two possibilities are accommodated. In the first possibility, relying parties perform their own assessment of the subject authority's suitability. In the second possibility, relying parties rely upon an accreditation authority.

2.1 Qualifying as a relying party

There are two variants of the basic application-layer protocol:

1. a one-pass protocol; and
2. a three-pass protocol.

In the (mandatory) one-pass variant, the subject authority offers its public verification key for use by any relying party provided the use is consistent with the terms and conditions specified in the protocol. It does this by sending the APO.XML message (in accordance with the DTD given in Section 3.1). "Sending the message" may mean posting it on a web site, and having potential relying parties obtain it from the site.

Relying party		Subject authority
	←	APO.XML

The (optional) three-pass variant should be used when relying parties have to be qualified by the subject authority, and the relying party requires a token confirming that it has been qualified. Restricting use of the authority public key to only qualified relying parties is enforced using strong authentication of the relying party to a repository of certificates and CRLs operated by the subject authority; relying parties which are not qualified will be denied access to the repository[1]. Two additional messages are added to the basic protocol: the APA.XML message represents acceptance by the relying party of the conditions applied to the use of the authority public key and supplies a public key by means of which the relying party, or a proxy operated by the relying party, will subsequently authenticate itself to the repository. And the APC.XML message represents confirmation by the subject authority that the relying party is a member of the community of qualified relying parties.

Relying party		Subject authority
	←	APO.XML
APA.XML	→	
	←	APC.XML

2.2 Subject authority accreditation

Where a subject authority has been accredited by one or more accreditation authorities, it may include attribute certificates issued by the accreditation authorities in the APO.XML message. Alternatively, assurance may be

[1] Where it is not acceptable for unqualified entities to obtain a copy of the subject authority's public key, the IETF's PKIX-CMP protocol should be used in place of this one.

obtained from the legal and regulatory framework in which the subject authority operates or the existing relationship between the parties.

3. Protocol definition

The one-pass variant of the application-layer protocol is executed in four steps:

1. The subject authority prepares an XML document in accordance with the XML document type definition (DTD) given in section 3.1, below. This document is a statement of the conditions surrounding the use of the authority public verification key, which may be rendered as an English-language HTML legal document by applying the (mandatory) XSL style-sheet given in section 3.4, below.

2. The resulting XML document is then protected as a PKCS#7 detached signature construct, signed with the subject authority private signature key and including the self-signed certificate of the subject authority. Optionally, the subject authority may include attribute certificates issued to it by accreditation authorities.

3. The resulting structure is communicated to relying parties, preferably by posting it on an HTTP server as a file named APO.XML.

4. The relying party gets the APO.XML file, and then checks:

    a) that the signature construct is valid;

    b) that the self-signed certificate is valid;

    c) that the validation string calculated from the certificate is the same as one obtained for the subject authority by an out-of-band method (see Section 3.7, below);

    d) that it is a valid authority public-key offer, in accordance with the DTD; and

    e) that it satisfies its own requirements for acceptable certification authorities (out of scope).

    If these checks are successful, then the relying party accepts the subject authority's public verification certificate. If any one of the checks is unsuccessful, then the relying party shall reject the subject authority's public verification certificate.

If the offer indicates that relying parties must be qualified, then the three-pass variant is executed. The three-pass variant of the protocol adds the following steps to those specified above:

1. The relying party prepares an APA.XML message in accordance with the XML DTD given in Section 3.2, below;

2. The resulting XML document is then protected as a PKCS#7 detached signature construct, signed with the relying party's private signature key and including the corresponding certificates. The certificates are the

(potentially self-signed) certificate of the relying party authority and a certificate issued by the relying party authority to a proxy which will directly access the subject authority's repository on behalf of the relying party, this latter certificate is conveyed as an authenticated attribute;

3. The resulting structure is communicated to the subject authority using the method specified in the acceptance-method element of the APO.XML file.

4. The subject authority gets the APA.XML file, and then checks:

    a) that the signature construct is valid;

    b) that the certificates are valid;

    c) if the relying party's certificate is self-signed, that the validation string calculated from the relying party's certificate is the same as one obtained for the relying party by an out-of-band method;

    d) otherwise, if the relying party certificate is not self-signed, checks that the relying party certificate is valid;

    e) that it is a valid authority public-key acceptance, in accordance with the APA DTD specified in section 3.2, below; and

    f) that it satisfies its own requirements for acceptable relying parties (out of scope).

    If these checks are successful, then the subject authority's repository access controls are modified to accept requests from the relying party's proxy and it sends the confirmation message. If any one of the checks is unsuccessful, then the subject authority does not modify the access controls and it sends the rejection message. The rejection message is not specified.

The following steps are then executed.

1. The subject authority prepares an APC.XML message in accordance with the XML DTD given in Section 3.3, below;

2. The resulting XML document is then protected as a PKCS#7 detached signature construct, signed with the subject authority's private signature key;

3. The resulting structure is communicated to the relying party using the method specified in the confirmation-method element of the APA.XML file; and

4. The relying party gets the APC.XML file, and then checks:

    a) that the signature construct is valid;

    b) that it is a valid authority public-key confirmation, in accordance with the APC DTD specified in section 3.3, below;

    c) that it indicates acceptance of the relying party as a qualified relying party; and

d) that it satisfies its own requirements for evidence of qualification (out of scope).

If these checks are successful, then the relying party accepts the subject authority's public verification certificate and then relies upon certificates issued by the subject authority. If any one of the checks is unsuccessful, then the relying party should not use the subject authority's public key.

3.1 APO.DTD

This section contain the (mandatory) DTD of the authority public-key offer (APO.XML) message.

```
<!--
    THE APO.DTD DOCUMENT TYPE DEFINITION

    This DTD defines how the variable items of an authority public-key offer
    are to be formed.

     When relying party applications receive an Authority Public-Key Offer (APO) in the form of an
     XML instance (conforming to this DTD), they may process the offer
    in accordance with pre-programmed guidelines.  (An example of a
    pre-programmed guideline might be requiring that the value of aggregate
    liability exceed $US 1 000   000.)

    Because the use of an authority public key is a legal, as well as a
    technical, matter, the authority public-key distribution protocol defines
    an XSL style-sheet for rendering the XML instance of an authority public
    key offer as a complete, legal document.
-->


<!--
    The <authority-public-key-offer> element is the root element of the DTD.

    The public-id attribute indicates this specification defining the DTD for
    the XML instance.
-->
<!ELEMENT    authority-public-key-offer    ( logo, parameters )>
<!ATTLIST    authority-public-key-offer
             public-id  CDATA  #FIXED "  rfcXXXX-apo" >

<!--
    Specify the Subject Authority's identity logo in the <logo>
    element.  The content of the <logo> element should be a
    base64-encoded GIF image.

    NOTE: The logo is NOT rendered in the legal contract.

    If the direct relying party is a person, then the logo should
    be displayed whenever a certificate issued by the subject
    authority is correctly validated.
->
<!ELEMENT    logo         (#PCDATA) >
<!--
    The <parameters> element contains the details of the offer.
-->
<!ELEMENT    parameters       (
                               subject-authority,
                              logo?,
                                    subjects,
                               subjects-name-spaces,
                               applications,
                               policy-  oids,
                                 url-certificate-repository,
                                 url-revocation-status,
                               acceptance-method?,
                               notices,
                                 enquiry-address,
                               arbitration?,
                               registration-authority,
                               validation-string) >


<!--
```

```
    Identify the subject authority whose verification public key is offered.
-->
<!ELEMENT    subject-authority    ( #PCDATA )>

<!--
    The <subjects> element identifies the relationship between the subject authority
    and those to whom it issues certificates.
-->
<!ELEMENT    subjects            (   subject+ ) >

<!ELEMENT    subject                    EMPTY >
<!ATTLIST    subject        type    ( employees | contractors | customers |
                                     agents | members | related-authorities |
                                     unrelated-authorities ) #REQUIRED >


<!--
    Use the <subjects-name-spaces> element to specify the portion of the name-space in which
subjects are located.
-->
<!ELEMENT    subjects-name-spaces            ( name-   space+ ) >

<!ELEMENT    name-space                   (#PCDATA) >


<!--
    Specify the acceptable uses of the certificates issued by the
    subject authority.
-->
<!ELEMENT    applications   (
                                 data-encryption |
                            digital-signature |
                            commitment
                                 )+  >


<!--
    Specify an empty "<data-encryption/>" element if data encryption is a valid
    application for the issued certificates.

    Indicate if the non-disclosure clause is to be included in the legal rendition
    by setting the "include-confidentiality-statement" attribute to either yes
    or no.
-->
<!ELEMENT   data-encryption            EMPTY >
<!ATTLIST   data-encryption               include-confidentiality-statement
                                               ( yes | no ) #REQUIRED >


<!--
    Specify an empty "<digital-signature/>" element if digital signing
    (without assumption of liability) is a valid application for the issued certificates.
    For example, if the subjects' certificates are to be used solely for
    authenticating a subject's identity, no assumption of liability on the part of the
    subject authority may be necessary.

    If the Certification Authority is willing to assume liability for
    applications involving digital signatures, specify the <commitment>
    element.
-->
<!ELEMENT    digital-signature        EMPTY >

<!--
    Specify a "<commitment>" element if the subject authority is willing
    to assume liability for applications involving digital signature.
-->
<!ELEMENT    commitment        (per-transaction-limit?,
                                aggregate-limit?) >

<!--
    If the commitment use is allowed, then per-transaction and
    aggregate liability limits may also be specified.

     The <currency> element specifies
     the ISO 4217 three-character code for currency.
-->
<!ELEMENT    per-transaction-limit              (limit, curre  ncy) >
<!ELEMENT    aggregate-limit                 (limit, currency) >

<!ELEMENT    limit                  (#PCDATA) >
<!ELEMENT    currency               (#PCDATA) >
```

```
<!--
    If the issued certificates to which this offer applies are to be limited
     to those containing particular certificate policy identifiers, specify those policy      OIDs.
-->
<!ELEMENT    policy- oids        ( policy- oid+ )>

<!ELEMENT    policy- oid        ( #PCDATA )>


<!--
    Specify the URL by which issued certificates can be retrieved.
-->
<!ELEMENT     url-certificate-repository    ( #PCDATA )>


<!--
    Specify the URL by which the revocation status of issued certificates can
    be retrieved.
-->
<!ELEMENT     url-revocation-status        ( #PCDATA )>


<!--
    Specify the URL by which the offer is to be accepted by the relying party.
-->
<!ELEMENT    acceptance-method        ( #PCDATA )>


<!--
    The <notices> element defines how to register to obtain updates about the offer.
-->
<!ELEMENT    notices                ( notice-title, notice-address )>


<!--
    Specify the title to appear in email sent to the address defined by the
    <notice-address> element.
-->
<!ELEMENT    notice-title            ( #PCDATA )>


<!--
    Specify the email address to which requests to register for notices should be sent.
-->
<!ELEMENT    notice-address          ( #PCDATA )>


<!--
    The email address to which   enquiries about this offer are to be sent.
-->
<!ELEMENT     enquiry-address         ( #PCDATA )>


<!--
    Arbitration may be performed by an arbitrator, acting in
    accordance with governing law specified in the contract.  Alternatively, there
    may be no pre-defined arbitrator.  Use the <arbitrator> element to
     specify the arbitrator and/or governing law.

    If there is to be no arbitration body specified, do not include
    an <arbitration> element.
-->

<!ELEMENT    arbitration             ( arbitrator, governing-law? )>

<!ELEMENT    arbitrator              ( #PCDATA )>

<!ELEMENT    governing-law           ( #PCDATA )>


<!--
    Specify the registration authority which publishes the validation string
     associated with the authority public key being offered.
-->
<!ELEMENT    registration-authority    ( #PCDATA )>


<!--
    Specify the validation string associated with the authority public key
     being offered.
```

```
-->
<!ELEMENT    validation-string         ( #PCDATA )>
```

3.2 APA.DTD

This section contains the (optional) DTD of the authority public-key acceptance message. The acceptance message supplies a relying party's proxy public key which will form the basis for strong authentication of the relying party's requests for certificates and/or CRLs from the subject authority's repository. In addition, it forms the request by the candidate relying party to be qualified as a relying party.

```
<!--
    THE APA.DTD DOCUMENT TYPE DEFINITION

    This DTD defines how the variable items of an authority public-key
    acceptance are to be formed.

    Because the use of an authority public key is a legal, as well as a
    technical, matter, the authority public-key distribution protocol
    defines an XSL style-sheet for rendering the XML instance of an
    authority public-key acceptance as a complete, legal document.
-->


<!--
    The <authority-public-key-acceptance> element is the root element
    of the DTD.

    The public-id attribute indicates this specification defining the DTD
    for the XML instance.
-->
<!ELEMENT    authority-public-key-acceptance    ( parameters )>
<!ATTLIST    authority-public-key-acceptance
               public-id  CDATA  #FIXED "   rfcXXXX-apa" >

<!--
    The <parameters> element contains the details of the acceptance.
-->
<!ELEMENT    parameters     (
                            relying-party,
                            subject-authority,
                            applications,
                            policy-  oids,
                            notice-address,
                            confirmation-method,
                            registration-authority,
                            validation-string
                            ) >

<!--
    Specify the party accepting the offer of the Certification Authority's
    verification public key.
-->
<!ELEMENT    relying-party            ( #PCDATA )>

<!--
    Specify the subject authority whose verification public key is to be accepted.
-->
<!ELEMENT    subject-authority        ( #PCDATA )>

<!--
    Specify the uses to which the certificates issued by
    the subject authority will be put.  The set of applications must be a sub-set of those offered
in the subject authority's APO.XML document.

    See the APO.DTD for details about the <applications> element.
-->
<!ELEMENT    applications   (
                            data-encryption |
                            digital-signature |
                            commitment
                            )+  >
```

```
<!ELEMENT    data-encryption            EMPTY >

<!ELEMENT    digital-signature        EMPTY >

<!ELEMENT    commitment     (
                            per-transaction-limit?,
                            aggregate-limit?
                            ) >

<!ELEMENT    per-transaction-limit           (limit, currency) >
<!ELEMENT    aggregate-limit                 (limit, currency) >

<!ELEMENT    limit                 (#PCDATA) >
<!ELEMENT    currency              (#PCDATA) >


<!--
    The offer may apply only to certificates that contain particular certificate policy
     identifiers.

    Specify which policy  OIDs will be used.  The
    set of policy  OIDs specified here must be a sub-set of those
    in the subject  authority's APO.XML document.
-->
<!ELEMENT    policy- oids       ( policy- oid+ )>
<!ELEMENT    policy- oid        ( #PCDATA )>

<!--
    Specify the email address to which the subject authority
    should send updates concerning its offer.
-->
<!ELEMENT    notice-address          ( #PCDATA )>

<!--
    Specify the URL by which the confirmation of the acceptance
     should be sent.
-->
<!ELEMENT    confirmation-method     ( #PCDATA )>

<!--
    Specify the registration authority publishing the validation string associated
    with the relying party's verification public key.
-->
<!ELEMENT    registration-authority  ( #PCDATA )>

<!--
    Specify the validation string associated with the relying party's verification
    certificate.
-->
<!ELEMENT    validation-string       ( #PCDATA )>
```

3.3 APC.DTD

This section contains the (optional) DTD of the authority public-key confirmation message. The confirmation message either supplies the relying party with a non-repudiation token indicating that it is a qualified relying party or a token indicating that its request to be recognized as a qualified relying party has been rejected.

```
<!--
    THE APC.DTD DOCUMENT TYPE DEFINITION

    This DTD defines how the variable items of an authority public-key
    confirmation are to be formed.

    Because the use of an authority public key is a legal, as well as a
    technical, matter, the authority public-key distribution protocol
    defines an XSL style-sheet for rendering the XML instance of an authority
    public-key confirmation as a complete, legal document.
-->


<!--
```

```
    The <authority-public-key-confirmation> element is the root element
    of the DTD.

    The public-id attribute indicates this specification defining the DTD
    for the XML instance.
-->
<!ELEMENT    authority-public-key-confirmation    ( parameters )>
<!ATTLIST    authority-public-key-confirmation
                          public-id  CDATA  #FIXED "   rfcXXXX-apc" >

<!--
    The <parameters> element contains the details of the confirmation.
-->
<!ELEMENT    parameters     (
                          subject-authority,
                          relying-party,
                          start-date,
                          end-date,
                          policy-  oids,
                          applications
                          ) >

<!ELEMENT    subject-authority    ( #PCDATA )>
<!ELEMENT    relying-party        ( #PCDATA )>

<!--
    Specify the period of qualification in the form “YYYY MMM DD     HH:mm UCT”.
-->
<!ELEMENT    start-date      ( #PCDATA )>
<!ELEMENT    end-date        ( #PCDATA )>

<!--
    Specify the policy  OIDs which may be used by the
    relying party.
-->
<!ELEMENT    policy- oids       ( policy- oid+ )>
<!ELEMENT    policy- oid        ( #PCDATA )>

<!--
    Specify the uses for which the relying party may use the
    issued certificates.
-->
<!ELEMENT    applications   (
                          data-encryption |
                          digital-signature |
                          commitment
                          )+  >

<!ELEMENT    data-encryption            EMPTY >

<!ELEMENT    digital-signature        EMPTY >

<!ELEMENT    commitment     (
                          per-transaction-limit?,
                          aggregate-limit?
                          ) >

<!ELEMENT    per-transaction-limit           (limit, currency) >
<!ELEMENT    aggregate-limit                 (limit, currency) >

<!ELEMENT    limit                 (#PCDATA) >
<!ELEMENT    currency              (#PCDATA) >
```

3.4 APO.XSL

This section contains the (mandatory) style-sheet for rendering an authority public-key offer as an English-language legal document in HTML. Other languages and renditions may be defined in separate documents. In the event of a dispute, this rendition will be considered the basis of the contract between the subject authority and the relying party. An example rendition is given in Section 4.

```
<!--
    THE APO.XSL STYLE-SHEET
```

```
    This XSL style sheet is intended to be associated with an authority public
    key offer XML document.  When this style sheet is applied to such a
    document, the result is an HTML rendition of a complete offer.

    The format of this style-sheet is based on the XSL submission made to the
    W3C ("A Proposal for  eXtensible Style Language (XSL)" dated 1997
    August 27).
-->

<xsl>
    <rule>
        <root/>
        <HTML><HEAD><TITLE>authority-public-key-offer</TITLE></HEAD>
        <BODY>
        <children/>
        </BODY></HTML>
    </rule>


<!-- The 'parameters' rule contains the generic legal text of the offer and an
     "element selector" for most parameters. -->
    <rule>
        <target-element  type="parameters"/>

        <H1>AUTHORITY PUBLIC-KEY OFFER</H1>

        <H2>Introduction</H2>
        <P>
        <select-elements><target-element   type="subject-authority"/>
        </select-elements>,
        hereinafter referred to as the 'subject authority',
        hereby offers its certification authority public verification key for
        the purpose of validating public-key certificates that may be issued
        by it from time to time.
        </P>

        <H2>Community</H2>
        <P>
        The subject authority affirms that the subjects identified in
        certificates issued by it may be considered to be
        <select-elements><target-element   type="subjects"/>
          </select-e lements>
        of the subject authority.
        </P>

        <P>
        Subject names in certificates issued by the subject authority are
        subordinate to the following name(s):
        <select-elements><target-element   type="subjects-name-spaces"/>
        </select-elements>.
        </P>

        <H2>Application</H2>
        <P>
        The proper use of certificates issued by the subject authority under
        the terms of this offer is limited to
        <select-elements><target-element   type="applications"/>
        </select-elements>.
        </P>

        <!-- If there is a <per-transaction-limit> element,
               select rule [1] below.  -->
        <select-elements  from="descendants">
          <element  type="applications">
          <element  type="commitment">
        <target-element  type="per-transaction-limit"/>
          </element>
          </element>
        </select-elements>

        <!-- If there is a <aggregate-limit> element,
               select rule [2] below.  -->
        <select-elements  from="descendants">
          <element  type="applications">
          <element  type="commitment">
          <target-element  type="aggregate-limit"/>
          </element>
          </element>
        </select-elements>

        <H2>Restrictions</H2>
```

```
<P>
This offer applies only to certificates which include a
 certificatePolicies extension which contains, inter    alia, the following
policy identifier(s):
<select-elements><target-element   type="policy-oids"/></select-elements>.
</P>

<H2>Certificate retrieval</H2>
<P>
Certificates issued by the subject authority may be retrieved using the
following method:
<select-elements><target-element   type="url-certificate-repository"/>
</select-elements>.
Access to this repository may be controlled (see below).
</P>

<H2>Processing</H2>
<P>
Certificates issued by the subject authority must be processed in
accordance with the certificate processing procedure of
ISO 9594-8 (1997), including verification of revocation status.
Revocation status may be obtained using the following method:
<select-elements><target-element   type="url-revocation-status"/>
</select-elements>.
Access to this method may be controlled (see below).
</P>

<!-- If the "include-confidentiality-statement" attribute of the
       <data-encryption> element (if any), is set to "yes",
       select rule [3] below.  -->
<select-elements  from="descendants"  mode="conf">
<element  type="applications">
<target-element  type="data-encryption">
<attribute  name="include-confidentiality-statement"   value="yes"/>
  </target-element>
</element>
</select-elements>

<!-- If there is an <acceptance-method> element,
       select rule [4] below.  -->
<select-elements><target-element   type="acceptance-method"/>
</select-elements>

<!-- Select rule [5] below for the <notices> element.  -->
<select-elements><target-element   type="notices"/>
</select-elements>

<H2>Contact details</H2>
<P>
 Enquiries concerning this offer may be submitted by sending email to
<select-elements><target-element   type="enquiry-address"/>
</select-elements>.
</P>

<!-- If there is an <arbitration> element,
       select rule [6] below.  -->
<select-elements><target-element   type="arbitration"/>
</select-elements>

<H2>Offer validation</H2>
<P>
To be considered valid, this offer must be encapsulated as a
 signedData construct in accordance with PKCS#7.
The PKCS#7 data construct must include a valid signature of the
subject authority. This signature must be verified using the public key
and any other constraints (including, but not limited to, the
signature algorithm identifier, the certificate validity period and
the certificate revocation status) contained in the certificate whose
issuer and subject both identify the subject authority and which is
also contained in the PKCS#7 data construct. It shall also be verified
that the validation string calculated from the public key contained in
the certificate matches the validation string associated with the
subject authority as published by the
<select-elements><target-element   type="registration-authority"/>
</select-elements>.
</P>
<P>
Note: the validation string shall be
<select-elements><target-element   type="validation-string"/>
  </select-elements>.
</P>
```

```
    <P>END OF OFFER</P>
</rule>


<!-- The following rules format the <subject> elements.  -->
<rule>
    <target-element  type="subject"  position="first-of-type"/>
    < eval> getAttribute("type")</ eval>
</rule>

<rule>
    <target-element  type="subject"/>
    , < eval> getAttribute("type")</ eval>,
</rule>

<rule>
    <target-element  type="subject"  position="last-of-type"/>
    and < eval> getAttribute("type")</ eval>
</rule>


<!-- The following rules format the contents of the <applications>
      element.  -->
<rule>
    <target-element  type="data-encryption"  position="first-of-any"/>
    data encryption
</rule>

<rule>
    <target-element  type="digital-signature"  position="first-of-any"/>
    digital signature
</rule>

<rule>
    <target-element  type="commitment"  position="first-of-any"/>
    commitment
</rule>


<rule>
    <target-element  type="data-encryption"/>
    , data encryption,
</rule>

<rule>
    <target-element  type="digital-signature"/>
    , digital signature,
</rule>

<rule>
    <target-element  type="commitment"/>
    , commitment,
</rule>


<rule>
    <target-element  type="data-encryption"  position="last-of-any"/>
    and data encryption
</rule>

<rule>
    <target-element  type="digital-signature"  position="last-of-any"/>
    and digital signature
</rule>

<rule>
    <target-element  type="commitment"  position="last-of-any"/>
    and commitment
</rule>


<!-- The following rules format <policy-  oid> and <name-space>
      elements are nicely formatted.  -->
<rule>
    <target-element  type="policy-oid"  position="first-of-type"/>
    <target-element  type="name-space"  position="first-of-type"/>

    <children/>
</rule>

<rule>
```

```
    <target-element  type="policy-oid"/>
    <target-element  type="name-space"/>

    , <children/>,
</rule>

<rule>
    <target-element  type="policy-oid"  position="last-of-type"/>
    <target-element  type="name-space"  position="last-of-type"/>

    and <children/>
</rule>


<!-- Rule [1]: format the <per-transaction-limit> element. -->
<rule>
    <target-element  type="per-transaction-limit"/>
    <P>
    The subject authority accepts liability to a limit of
    <children/> per transaction for damages resulting directly
       from reliance on its certificates when processed as described
       in this offer.
    </P>
</rule>

<!-- Rule [2]: format the <per-transaction-limit> element. -->
<rule>
    <target-element  type="aggregate-limit"/>
    <P>
    The subject authority accepts liability to a total limit of
    <children/> for damages resulting directly
       from reliance on its certificates when processed as described
       in this offer.
    </P>
</rule>

<rule>
    <target-element  type="limit"/>
    "<children/>"
</rule>

<rule>
    <target-element  type="currency"/>
    <children/>
</rule>


 <!-- Rule [3]:  If the "include-confidentiality-statement" attribute
        of the <data-encryption> element (if any), is set to "yes",
         do this rule when the  mode="conf" selector above is called.  -->
<rule>
    <element  type="applications"  mode="conf">
        <target-element  type="data-encryption">
              <attribute  name="include-confidentiality-statement"   value="yes"/>
          </target-element>
    </element>
    <H2>Non-disclosure</H2>
    <P>
    The subject authority declares that information provided to its
    subjects in encrypted form, and clearly marked 'confidential',
    will be treated in accordance with the subject authority's
    procedures for handling its own confidential information.
    </P>
</rule>


<!-- Rule [4]: format the <acceptance-method > element. -->
<rule>
    <target-element  type="acceptance-method"/>
    <H2>Access control</H2>
    <P>
    Requests for access must be submitted using the following method:
    <children/>.
    </P>
</rule>


<!-- Rule [5]: format the <notices> element. -->
<rule>
    <target-element  type="notices"/>
    <H2>Notices</H2>
```

```
        <P>
        From time to time, the subject authority may issue notices which may,
        inter  alia, invalidate this offer. In order to receive such notices,
        it is recommended that the user of certificates issued by the
        subject authority register with the subject authority by sending email
        with the title
        "<select-elements>
           <target-element  type="notice-title"/>
        </select-elements>"
        to
        <select-elements>
           <target-element  type="notice-address"/>
        </select-elements>.
        </P>
    </rule>


    <!-- Rule [6]: format the <arbitration> element.
          (This rule calls the t  wo rules following it.  -->
    <rule>
        <target-element  type="arbitration"/>
        <H2>Disputes</H2>

        <select-elements>
        <target-element  type="arbitrator"/>
        </select-elements>

        <select-elements>
        <target-element  type="governing-law"/>
        </select-elements>

    </rule>

    <rule>
        <target-element  type="arbitrator"/>
        <P>
        Any disputes arising out of the use of certificates issued by the
        subject authority shall be arbitrated by
        <children/>.
        </P>
    </rule>

    <rule>
        <target-element  type="governing-law"/>
        <P>
        The governing law for resolution of disputes shall be that in effect
           in the jurisdiction of <children/>.
        </P>
    </rule>


</xsl>
```

3.5 APA.XSL

This section contains the (optional) style-sheet for rendering an authority public-key acceptance as an English-language legal document in HTML. Other languages and renditions may be defined in separate documents. In the event of a dispute, this rendition will be considered the basis of the contract between the subject authority and the relying party.

```
<!--
    THE APA.XSL STYLE-SHEET

    This XSL style sheet is intended to be associated with an authority
    public-key acceptance XML document.  When this style sheet is applied
    to such a document, the result is an HTML rendition of a complete
    acceptance.

    The format of this style-sheet is based on the XSL submission made
    to the W3C ("A Proposal for   eXtensible Style Language (XSL)" dated
    1997 August 27).
-->
<xsl>
<rule>
    <root/>
```

```
        <HTML><HEAD><TITLE>authority-public-key-acceptance</TITLE></HEAD>
        <BODY>
        <children/>
        </BODY></HTML>
    </rule>


<!--
    The 'parameters' rule contains the generic legal text of the
    acceptance and an "element selector" for most parameters.
-->
    <rule>
        <target-element  type="parameters"/>
        <H1>AUTHORITY PUBLIC-KEY ACCEPTANCE</H1>

        <H2>Introduction</H2>
        <P>
        <select-elements><target-element   type="relying-party"/>
        </select-elements>,
        hereinafter referred to as the 'relying party', hereby accepts the
        conditions of use imposed upon its public key by
        <select-elements><target-element   type="subject-authority"/>
        </select-elements>.
        </P>

        <P>
        The relying party commits to use certificates issued by the subject
        authority in accordance with this acceptance only for the following
        applications:
        <select-elements><target-element   type="applications"/>
        </select-elements>.
        </P>

        <select-elements  from="descendants">
        <element  type="applications">
        <element  type="commitment">
        <target-element  type="per-transaction-limit"/>
        </element>
        </element>
        </select-elements>

        <select-elements  from="descendants">
        <element  type="applications">
        <element  type="commitment">
        <target-element  type="aggregate-limit"/>
        </element>
        </element>
        </select-elements>

        <P>
        The relying party commits to use only those certificates identified by
        the following object identifier(s):
        <select-elements><target-element   type="policy-oids"/>
        </select-elements>.
        </P>

        <select-elements><target-element   type="arbitration"/>
        </select-elements>

        <H2>Notices</H2>
        <P>
        Notices issued by the subject authority should be sent by email to the
        following email address:
        <select-elements><target-element   type="notice-address"/>
        </select-elements>.
        </P>

        <H2>Confirmation of acceptance</H2>
        <P>
        Confirmation of qualification as a relying party under the terms of
        the associated offer should be sent to the relying party by the
        following method:
        <select-elements><target-element   type="confirmation-method"/>
        </select-elements>.
        </P>

        <H2>Acceptance validation</H2>
        <P>
        To be considered valid, this acceptance must be encapsulated as
        a  signedData construct in accordance with PKCS#7.
        The PKCS#7 data construct must include a valid signature of the
```

```
    relying party.
    This signature must be verified using the public key and any
    other constraints
    (including, but not limited to, the signature algorithm identifier,
    the certificate validity period and the certificate revocation
    status)
    contained in the certificate whose issuer and subject both identify
    the relying party and which is also contained in the PKCS#7 data
    construct.
    It shall also be verified that the validation string calculated
    from the certificate matches the
    validation string associated with the relying party as published
    by the
    <select-elements><target-element   type="registration-authority"/>
    </select-elements>.
    </P>
    <P>
    Note: the validation string shall be
    <select-elements><target-element   type="validation-string"/>
    </select-elements>.
    The verification certificate of the proxy that may request
    certificate and revocation
    information from the subject authority on behalf of the relying
    party is included as an authenticated attribute in the PKCS#7
    data construct.
    </P>
    <P>END OF ACCEPTANCE</P>
</rule>


<rule>
    <target-element  type="data-encryption"  position="first-of-any"/>
    data encryption
</rule>

<rule>
    <target-element  type="digital-signature"  position="first-of-any"/>
    digital signature
</rule>

<rule>
    <target-element  type="commitment"  position="first-of-any"/>
    commitment
</rule>

<rule>
    <target-element  type="data-encryption"/>
    , data encryption,
</rule>

<rule>
    <target-element  type="digital-signature"/>
    , digital signature,
</rule>

<rule>
    <target-element  type="commitment"/>
    , commitment,
</rule>


<rule>
    <target-element  type="data-encryption"  position="last-of-any"/>
    and data encryption
</rule>

<rule>
    <target-element  type="digital-signature"  position="last-of-any"/>
    and digital signature
</rule>

<rule>
    <target-element  type="commitment"  position="last-of-any"/>
    and commitment
</rule>


<rule>
    <target-element  type="policy-oid"  position="first-of-type"/>
    <children/>
</rule>
```

```
    <rule>
        <target-element  type="policy-oid"/>, <children/>,
    </rule>

    <rule>
        <target-element  type="policy-oid"  position="last-of-type"/>
        and <children/>
    </rule>

    <rule>
        <target-element  type="acceptance-method"  position="last-of-type"/>
        and <children/>
    </rule>


    <rule>
        <target-element  type="per-transaction-limit"/>
        <P>
        The relying party acknowledges that the subject authority may be held
        liable to a limit of
        <children/> per transaction for damages resulting directly
        from reliance on its certificates.
        </P>
    </rule>

    <rule>
        <target-element  type="aggregate-limit"/>
        <P>
        The relying party acknowledges that the subject authority may be held
        liable to a limit of
        <children/> total for damages resulting directly
        from reliance on its certificates.
        </P>
    </rule>

    <rule>
        <target-element  type="limit"/>
        "<children/>"
    </rule>

    <rule>
        <target-element  type="currency"/>
        <children/>
    </rule>


    <rule>
        <target-element  type="arbitration"/>
        <H2>Disputes</H2>

        <select-elements>
        <target-element  type="arbitrator"/>
        </element>
        </select-elements>

        <select-elements>
        <target-element  type="governing-law"/>
        </element>
        </select-elements>

    </rule>

    <rule>
        <target-element  type="arbitrator"/>
        <P>
        Any disputes arising out of the use of certificates issued by the
        subject authority shall be arbitrated by
        <children/>.
        </P>
    </rule>

    <rule>
        <target-element  type="governing-law"/>
        <P>
        The governing law for resolutions of disputes shall be that in effect
        in the jurisdiction of <children/>.
        </P>
    </rule>

</xsl>
```

3.6 APC.XSL

This section contains the (optional) style-sheet for rendering an authority public-key confirmation as an English-language legal document in HTML. Other languages and renditions may be defined in separate documents. In the event of a dispute, this rendition will be considered the basis of the contract between the subject authority and the relying party.

```
<!--
    THE APC.XSL STYLE-SHEET


    This XSL style sheet is intended to be associated with an authority
    public-key confirmation XML document.  When this style sheet is
    applied to such a document, the result is an HTML rendition of a
    complete confirmation.

    The format of this style-sheet is based on the XSL submission made to
    the W3C ("A Proposal for   eXtensible Style Language (XSL)" dated
    1997 August 27).
-->
<xsl>
    <rule>
        <root/>
        <HTML><HEAD><TITLE>authority-public-key-confirmation</TITLE></HEAD>
        <BODY>
        <children/>
        </BODY></HTML>
    </rule>

<!-- The 'parameters' rule contains the generic legal text of the confirmation
     and an "element selector" for most parameters. -->
    <rule>
        <target-element  type="parameters"/>

        <H1>AUTHORITY PUBLIC-KEY CONFIRMATION</H1>

        <H2>Introduction</H2>
        <P>
        <select-elements><target-element   type="subject-authority"/>
        </select-elements>,
        hereinafter referred to as the 'subject authority',
        hereby confirms that the request by
        <select-elements><target-element   type="relying-party"/>
        </select-elements>
        to be a qualified relying party has been accepted.
        </P>
        <P>
        The qualification applies to the period starting on
        <select-elements><target-element   type="start-date"/></select-elements>
        and ending on
        <select-elements><target-element   type="end-date"/></select-elements>.
        <P>
        </P>
        The qualification applies to certificates identified by the following
        policy object identifiers:
        <select-elements><target-element   type="policy-oids"/>
        </select-elements>.
        </P>
        <P>
        The qualification applies to certificates for the following
        applications:
        <select-elements><target-element   type="applications"/>
        </select-elements>.
        </P>

        <select-elements  from="descendants">
        <element  type="applications">
        <element  type="commitment">
        <target-element  type="per-transaction-limit"/>
        </element>
        </element>
        </select-elements>

        <select-elements  from="descendants">
```

```
    <element  type="applications">
    <element  type="commitment">
    <target-element  type="aggregate-limit"/>
    </element>
    </element>
    </select-elements>

    <select-elements><target-element   type="arbitration"/>
    </select-elements>

    <H2>Confirmation validation</H2>
    <P>
    To be considered valid, this confirmation must be encapsulated as
    a  signedData construct in accordance with PKCS#7.
    The PKCS#7 data construct must include a valid signature of the
    subject authority.
    This signature must be verified using the public key supplied in
    the associated authority public-key offer.
    </P>
    <P>END OF CONFIRMATION</P>
</rule>


<rule>
    <target-element  type="data-encryption"  position="first-of-any"/>
    data encryption
</rule>

<rule>
    <target-element  type="digital-signature"  position="first-of-any"/>
    digital signature
</rule>

<rule>
    <target-element  type="commitment"  position="first-of-any"/>
    commitment
</rule>

<rule>
    <target-element  type="data-encryption"/>
    , data encryption,
</rule>

<rule>
    <target-element  type="digital-signature"/>
    , digital signature,
</rule>

<rule>
    <target-element  type="commitment"/>
    , commitment,
</rule>


<rule>
    <target-element  type="data-encryption"  position="last-of-any"/>
    and data encryption
</rule>

<rule>
    <target-element  type="digital-signature"  position="last-of-any"/>
    and digital signature
</rule>

<rule>
    <target-element  type="commitment"  position="last-of-any"/>
    and commitment
</rule>

<rule>
    <target-element  type="policy-oid"  position="first-of-type"/>
    <children/>
</rule>

<rule>
    <target-element  type="policy-oid"/>, <children/>,
</rule>

<rule>
    <target-element  type="policy-oid"  position="last-of-type"/>
    and <children/>
```

```
    </rule>

    <rule>
        <target-element  type="acceptance-method"  position="last-of-type"/>
        and <children/>
    </rule>


    <rule>
        <target-element  type="per-transaction-limit"/>
        <P>
        The subject authority accepts liability to a limit of
        <children/> per transaction for damages resulting directly
        from reliance on its certificates when processed as described
        in the associated offer.
        </P>
    </rule>

    <rule>
        <target-element  type="aggregate-limit"/>
        <P>
        The subject authority accepts liability to a total limit of
        <children/> for damages resulting directly
        from reliance on its certificates when processed as described
        in the associated offer.
        </P>
    </rule>

    <rule>
        <target-element  type="limit"/>
        "<children/>"
    </rule>

    <rule>
        <target-element  type="currency"/>
        <children/>
    </rule>

    <rule>
        <target-element  type="arbitration"/>
        <H2>Disputes</H2>

        <select-elements>
        <target-element  type="arbitrator"/>
        </element>
        </select-elements>

        <select-elements>
        <target-element  type="governing-law"/>
        </element>
        </select-elements>

    </rule>

    <rule>
        <target-element  type="arbitrator"/>
        <P>
        Any disputes arising out of the use of certificates issued by the
        subject authority shall be arbitrated by
        <children/>.
        </P>
    </rule>

    <rule>
        <target-element  type="governing-law"/>
        <P>
        The governing law for resolutions of disputes shall be that in effect
        in the jurisdiction of <children/>.
        </P>
    </rule>


</xsl>
```

3.7 Validation string algorithm

The validation string is calculated from the binary form of the authority's self-signed certificate by operating upon it, starting at the initial 0X30 and finishing with the final bit of the signature, with the SHA-1 hash algorithm. The right-most 8 bytes of the resulting digest are discarded. The left-most 3 bits of each of the remaining 12 bytes are discarded. The remaining twelve 5-bit values are represented as alphanumeric characters according to the following table.

00000 → A
00001 → B
… omitting I
11000 → Z
11001 → 3
11010 → 4
…
11111 → 9

Finally, the alphanumeric string is divided into three sub-strings, each of four characters, and the sub-strings are separated by hyphens. For example:

A4HY-8KLN-9T3M

This validation string is distributed out of band to the protocol. It may be distributed in software, printed on advertisements, letterhead and business cards, or it may be included in a register prepared by a third party, such as an industry association.

4. Examples (informative)

4.1 APO.XML

This section contains a sample authority public-key offer in XML.

```
<?xml version="1.0"?>
<!DOCTYPE authority-public-key-offer SYSTEM "   apo.dtd">

<authority-public-key-offer   public-id="rfcXXXX-apo">

<parameters>

<subject-authority>XYZ Corporation</subject-authority>

<subjects>
<subject  type="employees"/>
<subject  type="contractors"/>
</subjects>

<subjects-name-spaces>
<name-space>Name Space 1</name-space>
<name-space>Name Space 2</name-space>
</subjects-name-spaces>

<applications>

<data-encryption  include-confidentiality-  statement="yes"/>

<digital-signature/>

<commitment>
```

```
<per-transaction-limit>
<limit>1000</limit>
<currency>USD</currency>
</per-transaction-limit>

<aggregate-limit>
<limit>10,000,000</limit>
<currency >USD</currency>
</aggregate-limit>

</commitment>

</applications>

<policy- oids>
<policy- oid>{1 2 3 4}</policy- oid>
<policy- oid>{1 2 3 5}</policy- oid>
<policy- oid>{1 2 3 6}</policy- oid>
</policy- oids>

<url-certificate-repository>
http:// www.xyzcorp.com/ certs.html
</url-certificate-repository>

<url-revocation-status>
http:// www.xyzcorp.crls.html
</url-revocation-status>

<acceptance-method>mailto:access@xyzcorp.com</acceptance-method>

<notices>
<notice-title>Notice Title</notice-title>
<notice-address>notice@xyzcorp.com</notice-address>
</notices>

<enquiry-address>enquiry@xyzcorp.com</enquiry-address>

<arbitration>
<arbitrator>Arbitrators Incorporated</arbitrator>
<governing-law>State of Mind</governing-law>
</arbitration>

<registration-authority>Industry Association</registration-authority>

<validation-string>VALI-DATE-DSTR</validation-string>

</parameters>

</authority-public-key-offer>
```

4.2 Authority Public-Key Offer rendition

This section contains the text of the authority public-key offer which results from operating upon the sample APO.XML file in Section 4.1 with the APO.XSL file in Section 3.4.

AUTHORITY PUBLIC-KEY OFFER

Introduction

XYZ Corporation, hereinafter referred to as the 'subject authority', hereby offers its certification authority public verification key for the purpose of validating public-key certificates that may be issued by it from time to time.

Community

The subject authority affirms that the subjects identified in certificates issued by it may be considered to be employees and contractors of the subject authority.

Subject names in certificates issued by the subject authority are subordinate to the following name(s): Name Space 1 and Name Space 2.

Application

The proper use of certificates issued by the subject authority under the terms of this offer is limited to data encryption, digital signature, and commitment .

The subject authority accepts liability to a limit of "1000" USD per transaction for damages resulting directly from reliance on its certificates when processed as described in this offer.

The subject authority accepts liability to a total limit of "10,000,000" USD for damages resulting directly from reliance on its certificates when processed as described in this offer.

Restrictions

This offer applies only to certificates which include a certificatePolicies extension which contains, inter alia, the following policy identifier(s): {1 2 3 4}, {1 2 3 5}, and {1 2 3 6}.

Certificate retrieval

Certificates issued by the subject authority may be retrieved using the following method: http://www.xyzcorp.com/certs.html. Access to this repository may be controlled (see below).

Processing

Certificates issued by the subject authority must be processed in accordance with the certificate processing procedure of ISO 9594-8 (1997), including verification of revocation status. Revocation status may be obtained using the following method: http://www.xyzcorp.com/crls.html. Access to this method may be controlled (see below).

Non-disclosure

The subject authority declares that information provided to its subjects in encrypted form, and clearly marked 'confidential', will be treated in accordance with the subject authority's procedures for handling its own confidential information.

Access control

Requests for access must be submitted using the following method: mailto:access@xyzcorp.com.

Notices

From time to time, the subject authority may issue notices which may, inter alia, invalidate this offer. In order to receive such notices, it is recommended that the user of certificates issued by the subject authority register with the subject authority by sending email with the title "Notice Title" to notice@xyzcorp.com.

Contact details

Enquiries concerning this offer may be submitted by sending email to enquiry@xyzcorp.com.

Disputes

Any disputes arising out of the use of certificates issued by the subject authority shall be arbitrated by Arbitrators Incorporated.

The governing law for resolution of disputes shall be that in effect in the jurisdiction of State of Mind.

Offer validation

To be considered valid, this offer must be encapsulated as a signedData construct in accordance with PKCS#7. The PKCS#7 data construct must include a valid signature of the subject authority. This signature must be verified using the public key and any other constraints (including, but not limited to, the signature algorithm identifier, the certificate validity period and the certificate revocation status) contained in the certificate whose issuer and subject both identify the subject authority and which is also contained in the PKCS#7 data construct. It shall also be verified that the validation string calculated from the public key contained in the certificate matches the validation string associated with the subject authority as published by the Industry Association.

Note: the validation string shall be VALI-DATE-DSTR.

END OF OFFER

4.3 APA.XML

This section contains a sample authority public-key acceptance in XML.

```
<?xml version="1.0"?>
<!DOCTYPE authority-public-key-acceptance SYSTEM "apa.dtd">

<authority-public-key-acceptance public-id="rfcXXXX-apa">

<parameters>

<relying-party>ABC Corporation</relying-party>

<subject-authority>XYZ Corporation</subject-authority>

<applications>

<data-encryption/>

<digital-signature/>

<commitment>

<per-transaction-limit>
<limit>1000</limit>
<currency>USD</currency>
</per-transaction-limit>

<aggregate-limit>
<limit>10,000,000</limit>
<currency>USD</currency>
</aggregate-limit>

</commitment>

</applications>

<policy-oids>
<policy-oid>{1 2 3 4}</policy-oid>
<policy-oid>{1 2 3 5}</policy-oid>
</policy-oids>
```

```
<notice-address>notice@abccorp.com</notice-address>
<confirmation-method>Confirmation Method</mailto:confirmation@abccorp.com>
<registration-authority>Registration Authority</registration-authority>
<validation-string>ABCD-EFGH-JKLM</validation-string>
</parameters>
</authority-public-key-acceptance>
```

4.4 Authority Public-Key Acceptance rendition

This section contains the text of the authority public-key acceptance which results from operating upon the sample APA.XML file in Section 4.3 with the APA.XSL file in Section 3.5.

AUTHORITY PUBLIC-KEY ACCEPTANCE

Introduction

ABC Corporaton., hereinafter referred to as the 'relying party', hereby accepts the condition of use imposed upon its public key by XYZ Corporation.

The relying party commits to use certificates issued by the subject authority in accordance with this acceptance only for the following applications: data encryption, digital signature, and commitment .

The relying party acknowledges that the subject authority may be held liable to a limit of "1000" USD per transaction for damages resulting directly from reliance on its certificates.

The relying party acknowledges that the subject authority may be held liable to a limit of "10,000,000" USD total for damages resulting directly from reliance on its certificates.

The relying party commits to use only those certificates identified by the following object identifier(s): {1 2 3 4} and {1 2 3 5}.

Notices

Notices issued by the subject authority should be sent by email to the following email address: notice@acbcorp.com.

Confirmation of acceptance

Confirmation of qualification as a relying party under the terms of the associated offer should be sent to the relying party by the following method: confirmation@abccorp.com.

Acceptance validation

To be considered valid, this acceptance must be encapsulated as a signedData construct in accordance with PKCS#7. The PKCS#7 data construct must include a valid signature of the relying party. This signature must be verified using the public key and any other constraints

(including, but not limited to, the signature algorithm identifier, the certificate validity period and the certificate revocation status) contained in the certificate whose issuer and subject both identify the relying party and which is also contained in the PKCS#7 data construct. It shall also be verified that the validation string calculated from the certificate matches the validation string associated with the relying party as published by the Registration Authority.

Note: the validation string shall be ABCD-EFGH-JKLM. The verification certificate of the proxy that may request certificate and revocation information from the subject authority on behalf of the relying party is included as an authenticated attribute in the PKCS#7 data construct.

END OF ACCEPTANCE

4.5 APC.XML

This section contains a sample authority public-key confirmation in XML.

```
<?xml version="1.0"?>
<!DOCTYPE authority-public-key-confirmation SYSTEM "apc.dtd">

<authority-public-key-confirmation  public-id="rfcXXXX-apc">

<parameters>

<subject-authority>XYZ Corporation</subject-authority>

<relying-party>ABC Corporation</relying-party>

<start-date>1998 Jan 01 00:00 UCT</start-date>
<end-date>2000 Dec 31 23:59 UCT</end-date>

<policy-oids>
<policy-oid>{1 2 3 4}</policy-oid>
<policy-oid>{1 2 3 5}</policy-oid>
<policy-oid>{1 2 3 6}</policy-oid>
</policy-oids>

<applications>

<data-encryption/>

<digital-signature/>

<commitment>

<per-transaction-limit>
<limit>1000</limit>
<currency>USD</currency>
</per-transaction-limit>

<aggregate-limit>
<limit>10,000,000</limit>
<currency>USD</currency>
</aggregate-limit>

</commitment>

</applications>

</parameters>
</authority-public-key-confirmation>
```

4.6 Authority Public-Key Confirmation rendition

This section contains the text of the authority public-key confirmation which results from operating upon the sample APC.XML file in Section 4.5 with the APC.XSL file in Section 3.6.

AUTHORITY PUBLIC-KEY CONFIRMATION

Introduction

XYZ Corporation, hereinafter referred to as the 'subject authority', hereby confirms that the request by ABC Corporation to be a qualified relying party has been accepted.

The qualification applies to the period starting on 1998 Jan 01 00:00 UCT and ending on 2000 Dec 31 23:59 UCT.

The qualification applies to certificates identified by the following policy object identifiers: {1 2 3 4}, {1 2 3 5}, and {1 2 3 6}.

The qualification applies to certificates for the following applications: data encryption , digital signature, and commitment .

The subject authority accepts liability to a limit of "1000" USD per transaction for damages resulting directly from reliance on its certificates when processed as described in the associated offer.

The subject authority accepts liability to a total limit of "10,000,000" USD for damages resulting directly from reliance on its certificates when processed as described in the associated offer.

Confirmation validation

To be considered valid, this confirmation must be encapsulated as a signedData construct in accordance with PKCS#7. The PKCS#7 data construct must include a valid signature of the subject authority. This signature must be verified using the public key supplied in the associated authority public-key offer.

END OF CONFIRMATION

5. Notes (Informative)

5.1 Cross-certification

The relying party may act as an agent for a community of relying parties by issuing a cross-certificate for the public verification key of the subject authority. The contents of the authority public-key offer may form the basis for the contents of the cross-certificate.

1. Community - The relying party is entitled to rely on legal provisions which apply to the subject authority and its subscriber community when they are bound in the relationship declared in the “subjects” element. If the subject authority identifies that it issues certificates to unrelated authorities under this policy, then the cross-certificate should include a basic constraints extension with a skip certs value of zero. If the subject authority identifies that it does not issue certificates to unrelated authorities, then the relying party may omit the basic constraints extension. In this case, the subject authority accepts responsibility for ensuring that all subsequent authorities in the certificate validation path adhere to the conditions laid out in the offer.

2. Restrictions - The relying party authority should include in its cross-certificate a certificate policies extension which lists its own identifiers for the certificate policies implied by the contents of the offer. It should also include a policy mappings extensions which asserts the mappings between its identifiers and those listed in the “Restrictions” section. It should also include a policy constraints extension, identifying that its relying parties must find explicit certificate policies in the certificates issued by the subject authority. The certificate policies and policy mappings extensions may be marked non-critical, but the policy constraints extension should be marked critical.

3. Name constraints - The relying party authority should include in its cross-certificate a name constraints extension restricting acceptable certificates to that portion of the name space identified by the “subjects-name-spaces” element.