



Publication 5830

Modernized e-File (MeF) IS State and Trading Partners Reference Guide

Modernized e-File Release 10.8

The Internal Revenue Service Mission

Provide America's taxpayers top quality service by helping them understand and meet their tax responsibilities and by applying the tax law with integrity and fairness to all.



Table of Contents

1 Interaction with Third Party Transmitters and States.....	8
1.1 Interface Overview	8
1.2 Business Rules	9
1.3 Basic SOAP Message Structure.....	9
1.3.1 Header structure for MIME request without attachment.....	11
1.3.2 Header structure for MTOM request without external attachment	11
1.3.3 Header structure for MTOM request without external attachment	11
1.3.4 Header structure for MIME request with attachment	12
1.4 SOAP Header.....	13
1.4.1 WS-Addressing	13
1.4.2 MeF Header	13
1.4.3 WS-Security for All A2A Web Service Users.....	13
1.4.4 WS-Security for Strong Authentication Users	16
1.5 SOAP Body.....	23
1.5.1 Message Attachments Zip File Format	23
1.5.2 Message Attachments Content Type	23
1.6 MeFHeader Elements	24
1.6.1 element MeFHeaderType/MessageID	26
1.6.2 element MeFHeaderType/RelatesTo	27
1.6.3 element MeFHeaderType/Action	28
1.6.4 element MeFHeaderType/MessageTs	29
1.6.5 element MeFHeaderType/ETIN	30
1.6.6 element MeFHeaderType/SessionKeyCd	31
1.6.7 element MeFHeaderType/TestCd	32
1.6.8 element MeFHeaderType/NotificationResponse	33
1.6.9 element MeFHeaderType/NotificationDesc	33
1.6.10 element MeFHeaderType/NotificationDesc	34
1.6.11 element MeFHeaderType/AppSysID.....	34
1.6.12 element MeFHeaderType/WSDLVersionNum.....	35
1.6.13 element MeFHeaderType/ClientSoftwareTxt	36
1.6.14 attribute Id.....	36
1.7 MeF A2A ErrorExceptionalDetail Elements	36
1.8 MeF A2A Web Services Summary.....	38
1.8.1 Modernized System Infrastructure (MSI) Services	41
1.8.2 Third Party Transmitter Web Services.....	53
1.8.3 State Web Services	79
1.8.4 Exceptions	112
1.9 Attachment Files	113
1.10 IFA Service Messages.....	119
1.11 A2A Web Services Error Handling and Exception Reporting.....	120



1.11.1	XML Gateway Exceptions.....	120
1.11.2	Faults Response	121
2	Data Mapping	126
2.1	Data Mapping for A2A Web Services	126
2.1.1	Namespaces Used for A2A Web Services.....	126
2.1.2	Data Mapping for the IFA Services.....	128
2.1.3	Namespaces Used for the IFA Services	128
3	IFA Interface Specification	129
3.1	Transmitter Operations.....	129
3.2	Transmission File Specification.....	129
4	Example A2A Web Service Messages	131
4.1	MSI Services Messages	131
4.1.1	Login Service Request Message	131
4.1.2	Login Service Response Message	133
4.1.3	Logout Service Request Message.....	135
4.1.4	Logout Service Response Message.....	137
4.1.5	EtinRetrieval Service Request Message.....	138
4.1.6	EtinRetrieval Service Response Message.....	140
4.1.7	EtinStatus Service Request Message	143
4.1.8	EtinStatus Service Response Message	143
4.1.9	GetStateParticipantsList Service Request Message	144
4.1.10	GetStateParticipantsList Service Response Message.....	144
4.2	State Services SOAP Messages.....	146
4.2.1	GetNewSubmissions Service Request Message	146
4.2.2	GetNewSubmissions Service Response Message	148
4.2.3	Get Submissions by Message ID Service Request Message	151
4.2.4	Get Submissions by Message ID Service Response Message	151
4.2.5	Get Submissions Service Request Message	152
4.2.6	GetSubmissions Service Response Message	153
4.2.7	GetSubmission Service Request Message	154
4.2.8	GetSubmission Service Response Message	155
4.2.9	SendSubmissionReceipts Service Request Message	155
4.2.10	SendSubmissionReceipts Service Response Message	156
4.2.11	SendAcks Service Request Message	157
4.2.12	SendAcks Service Response Message	157
4.2.13	GetNewAckNotifications Service Request Message	158
4.2.14	GetNewAckNotifications Service Response Message	158
4.2.15	GetAckNotifications Service Request Message	159
4.2.16	GetAckNotifications Service Response Message	159
4.2.17	GetAckNotifications Service Request Message	160
4.2.18	GetAckNotifications Service Response Message	161
4.2.19	GetSubmissionReconciliationList Service Request Message	162
4.2.20	GetSubmissionReconciliationList Service Response Message	162



4.3	Transmitter Services SOAP Messages	163
4.3.1	SendSubmissions Service Request Message	163
4.3.2	SendSubmissions Service Response Message	166
4.3.3	GetNewAcks Service Request Message.....	168
4.3.4	GetNewAcks Service Response Message.....	168
4.3.5	Get Acks by Message ID Service Request Message	169
4.3.6	Get Acks by Message ID Service Response Message	169
4.3.7	GetAcks Service Request Message	170
4.3.8	GetAcks Service Response Message	171
4.3.9	GetAck Service Request Message	171
4.3.10	GetAck Service Response Message	172
4.3.11	GetNewSubmissionsStatus Service Request Message.....	172
4.3.12	GetNewSubmissionsStatus Service Response Message.....	173
4.3.13	GetSubmissionsStatus Service Request Message	173
4.3.14	GetSubmissionsStatus Service Response Message	174
4.3.15	GetSubmissionStatus Service Request Message	175
4.3.16	GetSubmissionStatus Service Response Message	175
4.3.17	GetNew2290Schedule1 Service Request Message	176
4.3.18	GetNew2290Schedule1s Service Response Message	176
4.3.19	Get2290Schedule1sByMsgID Service Request Message	177
4.3.20	Get2290Schedule1sByMsgID Service Response Message	178
4.3.21	Get2290Schedule1s Service Request Message.....	178
4.3.22	Get2290Schedule1s Service Response Message.....	179
4.3.23	Get2290Schedule1 Service Request Message.....	180
4.3.24	Get2290Schedule1 Service Response Message.....	180
5	Fault Responses and Internal-Use Error Messages	182
5.1	Fault Responses Generated by the XML Gateway	182
5.2	Incoming Message Containing a Virus	186
5.2.1	Example SOAP Fault Response for the Virus Check Performed by the XML Gateway	186
5.2.2	Example Error Message for the Virus Check Performed by the XML Gateway ..	186
5.3	Outgoing Message Containing a Virus	186
5.3.1	Example SOAP Fault Response	187
5.3.2	Example Error Message	187
5.4	Incoming Messages Containing Service Request Argument Errors	187
5.4.1	Service Request Processing Errors	193
5.5	Acknowledgements for Duplicate Messages and Submissions	200
	Glossary	201
	Abbreviations and Acronyms	202



List of Figures

Figure 1-1: MeF Interface Overview	8
Figure 1-2: MeF Basic SOAP Message for HTTP Binding	10
Figure 1-3: MeF Basic SOAP with Attachments Message Structure	10
Figure 1-4: IRS Submission Message Structure	114
Figure 1-5: State Submission Message Structure	115
Figure 1-6: State Submission Response Message Structure	116
Figure 1-7: Acknowledgements Structure	117
Figure 1-8: Receipts Structure	117
Figure 1-9: Acknowledgement Notifications Structure	118
Figure 1-10: Status Records Structure	118
Figure 1-11: Stamped Form 2290 Schedule 1s Attachment Structure	119
Figure 1-12: IFA Transmission File Structure	120
Figure 3-1: Sample Transmission File Message Submitted to IFA.....	130

List of Tables

Table 1-1: Strong Authentication Request Message Elements	17
Table 1.6-2 Element MessageID Details	26
Table 1.6-3 Element RelatesTo Details	28
Table 1.6-4 Element Action Details.....	29
Table 1.6-5 Element MessageTs Details	29
Table 1.6-6 Element ETIN Details	30
Table 1.6-7 Element SessionKeyCd Details	31
Table 1.6-8 Element TestCd Details	32
Table 1.6-9 Element NotificationResponse Details	33
Table 1.6-10 Element NotificationDesc Details.....	33
Table 1.6-11 Element NotificationDt Details	34
Table 1.6-12 Element AppSysID Details.....	35
Table 1.6-13 Element WSDLVersionNum Details	35
Table 1.6-14 Element ClientSoftwareTxt Details	36
Table 2.1-1: A2A WSDL and XML Schema Files	126
Table 2.1-2: A2A Target Namespaces.....	126
Table 2.1-3: A2A XML Namespace Prefixes	127
Table 2.1-4: IFA XML Schema Files	128



Table 2.1-5: IFA Target Namespaces	128
Table 2.1-6: IFA XML Namespace Prefixes.....	128
Table 5.1-1: XML Gateway Threats and Fault Responses	185
Table 5.4-1: Service Request Argument Errors and Fault Responses.....	188
Table 5.4-2: Service Request Argument Errors and Fault Responses.....	193
Table 5.4-3: Service Request Processing Errors and Fault Responses	195
Table 5.4-4: Error Codes, Log Messages, Fault Responses, and Fault Types	197



1 Interaction with Third Party Transmitters and States

1.1 Interface Overview

As illustrated in Figure 1-1, MeF Interface Overview, there are two data communication channels between the IRS external Third Party Transmitters and States and the Modernized e-File (MeF) system.

- **The Internet Filing Application (IFA) channel** allows a human operator to transmit prepared SOAP message files using a Web browser. Section 3 covers upload and download formats for transmitters using the IFA channel
- **The Application-to-Application (A2A) channel** allows application software to directly interact with MeF, without requiring a human operator, using the Web Services request-response model. Most current MeF traffic uses this channel, and most of this document focuses on this channel

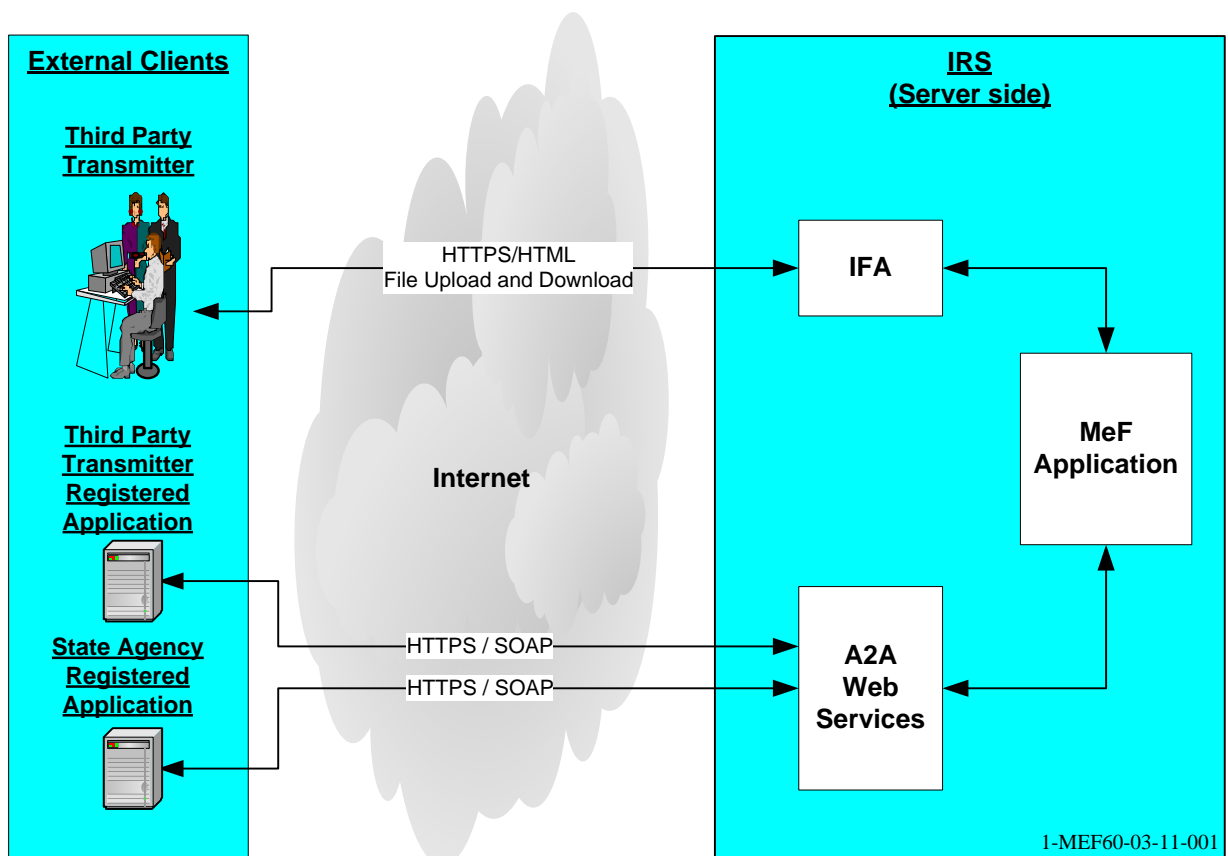


Figure 1-1: MeF Interface Overview

For both channels, data is exchanged between the external clients and the MeF system via Simple Object Access Protocol (SOAP) messages or files. Data is transmitted using the HyperText Transfer Protocol (HTTP) transport mechanism over a Secure Sockets Layer (SSL) HTTP Secure (HTTPS) connection. The SOAP data structures are specified in this document.



1.2 Business Rules

A number of business rules governing submission types and the usage of the two channels are summarized below.

- A transmitter can submit IRS submissions, state submissions, and IRS Exempt Organization (EO) submissions with request for redacted copies for states via the A2A Web Services and IFA channels.
- A state agency can only use the A2A Web Services to communicate with the MeF system. States are not allowed to use the IFA channel. Therefore, states must retrieve submissions using A2A, even if the transmitter sent the state submission to MeF using IFA.
- There are two possible usage patterns for A2A Web Services interactions:
- A session can be established that spans multiple requests, all of which use the same user credentials. This pattern can be used for all types of requests; and is in general the preferred pattern. Some requests, including all Send requests, must use this pattern.
- A single request can be submitted at a time, without establishing a session. This pattern applies only to Get requests and should only be used when only that one request is needed to accomplish the business need.

1.3 Basic SOAP Message Structure

As described earlier, the MeF system uses two channels to exchange SOAP-structured data. The Web Services A2A channel uses SOAP messages, while the IFA channel uses files with SOAP data structure. Beginning with MeF Release 6.2, every eXtensible Markup Language (XML) message (SOAP messages, federal and state submission manifests and submissions) must begin with an XML declaration that looks like `<?xml version="1.0" encoding="UTF-8"?>`. The declaration must always be the first XML statement on the first line of the file beginning in column 1. There may be other optional elements present, but the above format – including the elements `version="1.0"` and `encoding="UTF-8"` – must be present and contain exactly the values `1.0` and `UTF-8`. Beginning with Release 9.0, MeF A2A will perform WSDL verification – XML documents that do not strictly adhere to the WSDL/Schema definitions will be rejected at the MeF A2A portal.

The following sections describe the basic SOAP message structure for the Web Services A2A channel. The submission manifests and the submissions are described in Section 1.9.2.

This section describes the logical structure of basic messages with a SOAP Header and SOAP Body blocks within a SOAP message Envelope. A SOAP message contains one SOAP Header and one SOAP Body within one SOAP envelope. The SOAP Header contains the Web Services (WS) Addressing (WS-Addressing), WS-Security, and MeFHeader elements. The SOAP Body contains the MeF Message elements and attachment information (if any). See Figure 1-2, MeF Basic SOAP Message for HTTP Binding, and Figure 1-3, MeF Basic SOAP with Attachments Message Structure.



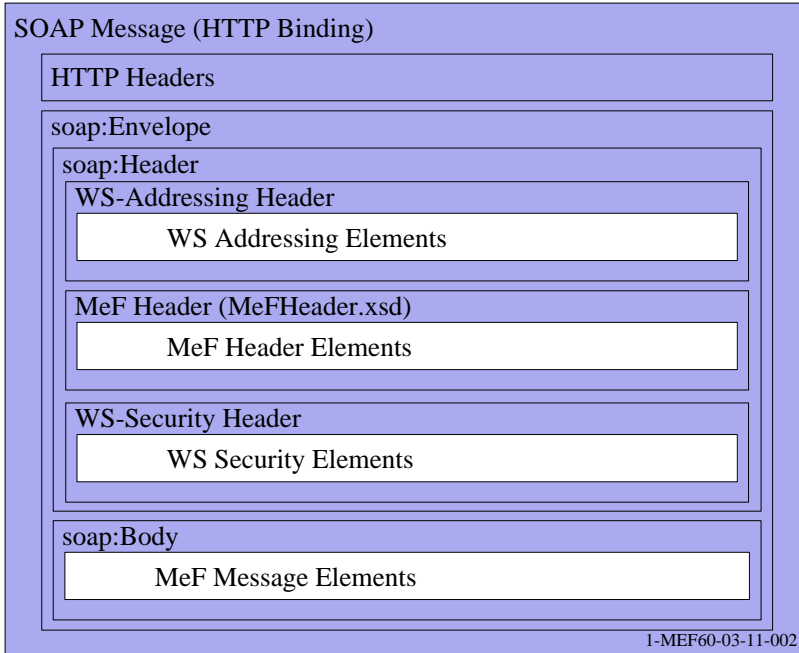


Figure 1-2: MeF Basic SOAP Message for HTTP Binding

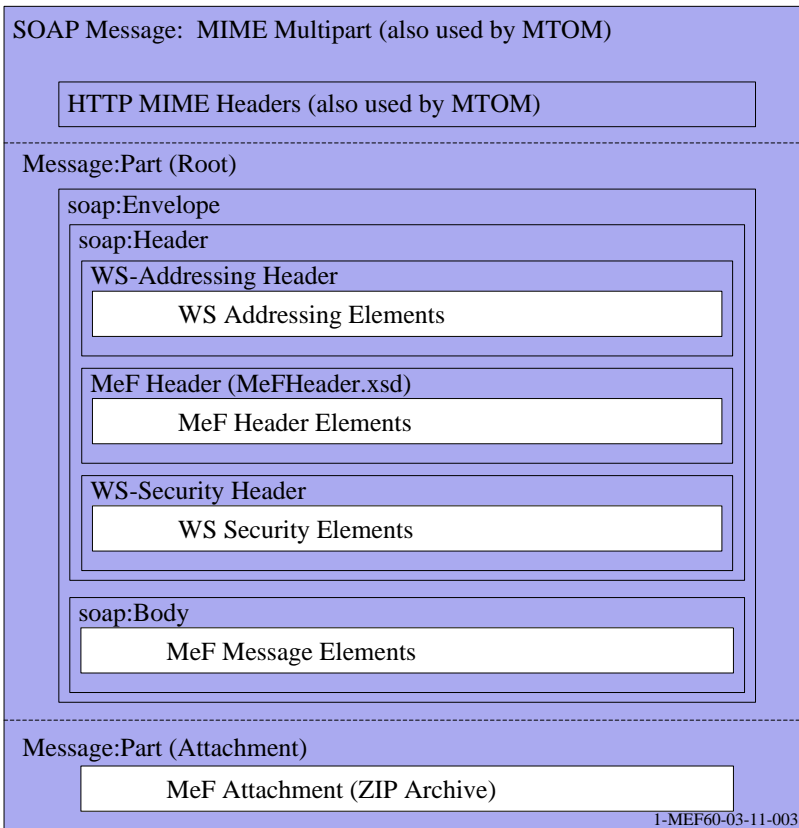


Figure 1-3: MeF Basic SOAP with Attachments Message Structure



1.3.1 Header structure for MIME request without attachment

A SOAP 1.1 request message with no attachment needs only an HTTP POST header. Requests whose URL's don't contain "mtom" are considered to be MIME.

```
POST /a2a/mef/mime/GetSubmissions HTTP/1.1
Host: la1.www4.irs.gov
Content-Type: text/xml; charset=UTF-8
Content-Length: 4600
SOAPAction: "http://www.irs.gov/a2a/mef/mime/GetSubmissions"
```

1.3.2 Header structure for MTOM request without external attachment

MTOM request messages include both an HTTP POST header and a SOAP body part one. A Send request using MTOM with an inline attachment has a similar header to that of a request with no attachment.

```
POST /a2a/mef/mtom/GetAcks HTTP/1.1
Host: la1.www4.irs.gov
Content-Type: multipart/related; boundary=
MIMEurn_uuid_1FACEDB95C3509148F1570480098765; type="application/xop+xml";
start="<0.urn:uuid:1FACEDB95C3509148F1570480098766@w3.org>"; start-
info="text/xml"
Content-Length: 4631
SOAPAction: "http://www.irs.gov/a2a/mef/mtom/GetAcks"

--MIMEurn_uuid_1FACEDB95C3509148F1570480098765
Content-Type: application/xop+xml; charset=UTF-8; type="text/xml"
Content-Transfer-Encoding: binary
Content-ID: <0.urn:uuid:1FACEDB95C3509148F1570480098766@w3.org>
```

1.3.3 Header structure for MTOM request without external attachment

For a Send whose attachment is not inline, there are three headers, HTTP POST, SOAP body part, and attachment:

```
POST /a2a/mef/mtom/SendSubmissions HTTP/1.1
Host: la1.www4.irs.gov
Content-Type: multipart/related;
boundary=MIMEurn_uuid_1FACEDB95C3509148F1570480012345;
type="application/xop+xml";
start="<0.urn:uuid:1FACEDB95C3509148F1570480012346@w3.org>"; start-
info="text/xml"
Transfer-Encoding: chunked
SOAPAction: "http://www.irs.gov/a2a/mef/mtom/SendSubmissions"
Cookie: JSESSIONID=ABcd5fGH9jKnMnP3qrstu4wz.15
```



```
--MIMEurn_uuid_1FACEDB95C3509148F1570480012345
Content-Type: application/xop+xml; charset=UTF-8; type="text/xml"
Content-Transfer-Encoding: binary
Content-ID: <0.urn:uuid:1FACEDB95C3509148F1570480012346@w3.org>
```

```
--MIMEurn_uuid_1FACEDB95C3509148F1570480012345
Content-Type: application/zip
Content-Transfer-Encoding: binary
Content-ID: <1.urn:uuid:1FACEDB95C3509148F1570480012347@w3.org>
```

The Content-ID of the Attachment body part must match the href parameter of the Include statement in the SOAP Body, as shown in the following example:

```
<SubmissionsAttMTOM><xop:Include
href="cid:1.urn:uuid:1FACEDB95C3509148F1570480012347@w3.org"
xmlns:xop="http://www.w3.org/2004/08/xop/include" /></SubmissionsAttMTOM>
```

1.3.4 Header structure for MIME request with attachment

A MIME request message uses a content type of “text/xml” rather than MTOM’s “application/xop+xml”. There is no clear reference to the Content-ID of the attachment part earlier in the message, unlike MTOM.

```
POST /a2a/mef/mime/SendSubmissions HTTP/1.1
Host: la1.www4.irs.gov
Content-Type: multipart/related; type="text/xml";
start="<42FE52987D0939FAD3338B20708ABCDE>";
boundary="====_Part_42_30840851.1577968723456"
Content-Length: 90957
SOAPAction: "http://www.irs.gov/a2a/mef/mime/SendSubmissions"
Cookie: JSESSIONID=AB3d56GHJ_6h0e1FL1ABiRUo.50
```

```
====_Part_42_30840851.1577968723456
Content-Type: text/xml; charset=UTF-8
Content-Transfer-Encoding: binary
Content-Id: <42FE52987D0939FAD3338B20708ABCDE>
```

```
====_Part_42_30840851.1577968723456
Content-Type: application/octet-stream
Content-Transfer-Encoding: binary
Content-Id: 7EA3351CD4472607DE6EE59D2939FACE
```



1.4 SOAP Header

The SOAP Header specification is to be provided by the state or transmitter application. There are three SOAP Header element structures allowed in SOAP messages on the A2A channel:

- **WS-Addressing** – Provides mechanisms for endpoint-to-endpoint identification in SOAP messages
- **MeF Header** – Provides elements required for all SOAP messages on the A2A channel
- **WS-Security** – Provides mechanisms for user identification and assertions in SOAP messages

1.4.1 WS-Addressing

WS-Addressing provides transport-neutral mechanisms to address Web Services and messages. This specification defines XML [XML 1.0, XML Namespaces] elements to identify Web Service endpoints and to secure end-to-end endpoint identification in messages. Sample WS-Addressing elements are presented below.

NOTE: None of the MeF services use any WS-Addressing elements. The MeF Header elements contain the required WS-Addressing-like information. Use of WS-Addressing elements is allowed, but they will be ignored by the MeF services.

```
<!-- WS Addressing Elements -->
xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing ">
<e:Header>
<wsa:MessageID>uuid:1136bbc2-ee59-11d9-b767-1136bbc1b767</wsa:MessageID>
<wsa:To>http://www.irs.gov/a2a/mef/Login</wsa:To>
<wsa:Action>http://www.irs.gov/a2a/mef/Login</wsa:Action>
<wsa:From>http://www.mycompany.com/myapplication/Login</wsa:From>
```

1.4.2 MeF Header

Every SOAP message to and from an A2A Web Service must contain an MeF Header in the SOAP message header. The MeF Header is described in Section 1.6, MeFHeader Elements.

1.4.3 WS-Security for All A2A Web Service Users

Web Services Security (WS-Security) provides transport-neutral mechanisms to send security tokens as part of a message, to provide message integrity, and support message confidentiality. This specification defines a standard set of SOAP extensions used when building secure Web Services to implement message content integrity and confidentiality.

In the past, the initial authentication could be done by supplying a WS-Security password. This ability was removed in 2017 when Strong Authentication was made a requirement.



1.4.3.1 Initial vs Subsequent Requests

There are two closely related mechanisms used to authenticate to the MeF A2A Service. Both use different subsets of the XML elements contained within the WS-Security <wsse:Security> element:

- Initial authentication is done by supplying an appropriate Digital Certificate and Username, signed using the key for that certificate, as part of the request. This applies to the first interaction of a session and to single stand-alone requests. In most cases, the initial request is a Login, although most Get requests can also be authenticated in this way and can begin a session. Section 1.4.4 below details the initial WS-Security request.
- Subsequent interactions in a session are authenticated by including the Security Assertion Markup Language (SAML) Assertion that was returned by the initial interaction of a session. Some requests, such as all Send requests, can only be authenticated this way; they must be preceded by a request (typically a Login) that begins a session. Section 1.4.3.3 below details the subsequent WS-Security request.

1.4.3.2 Session Authentication Behavior

The multiple-request pattern – establishing a session – requires that the client application extract the <saml:Assertion> element returned with the reply of the initial interaction and supply it in the WS-Security element of all subsequent requests in the session. It also requires that the MeFHeader SessionKeyCd element specify “Y” in all except the last request of a session.

NOTE: The SAML Assertion must be inserted exactly as it appeared in the initial reply, because its entire contents are digitally signed. Any alteration to the Assertion may invalidate it. This is a change in behavior from MeF Releases before 10.8.

The SAML assertion will remain valid for subsequent interactions, until any of:

- A Logout request is made
- A request is made that has the MeF Header SessionKeyCd element set to ‘N’.
- The configured lifetime of the SAML assertion is exceeded. Currently, SAML assertions are valid for two hours after they are initially created.

A request should contain either a user’s Certificate or a SAML assertion, not both, although a SAML assertion usually contains a certificate belonging to one of the IRS systems. If a request contains a SAML assertion, it will be treated as a subsequent interaction and will be authenticated using the SAML assertion.

1.4.3.3 SAML Assertion

SAML assertions exchange security information across the Third Party Transmitter or State application, and the MeF A2A services. The initial request (typically a Login) authenticates a client request and sends back a SAML-based SOAP assertion, which affirms the relying party with the security information to be used for subsequent MeF A2A service requests.

A sample SAML assertion is presented below. Every incoming request with a SAML assertion should contain the entire WS Security that was returned from a successful MeF A2A Login request. The SAML assertion must provide the Username element in the WS Security UsernameToken to enable validation of the SAML assertion. There must be no field other than Username in the WS Security UsernameToken for a message with a SAML assertion. Also, the request must not contain a BinarySecurityToken and the only Signature present must be the one already included inside the SAML assertion. Every incoming request with a SAML assertion for a Strong Authentication user must also provide the AppSysID element in the MeFHeader containing the same value as the Username in the WS Security UsernameToken. The actual WS Security Password element should never be present in a request with a SAML assertion.



```

<o:Security
  xmlns:o="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd">
  <o:UsernameToken
    xmlns:wssu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd"
    wsu:Id="id-d2acc8bcd0eda58da9d312d515053ec15f55f87d">
    <o:Username>112320001</o:Username>
  </o:UsernameToken>
  <saml:Assertion
    xmlns:saml="urn:oasis:names:tc:SAML:1.0:assertion"
    AssertionID="id-5eb4861e4c447f1b358600dfaa605215e72cd6b2"
    IssueInstant="2024-09-23T22:33:11.683Z"
    Issuer="http://www.forumsys.com/sentry" MajorVersion="1"
    MinorVersion="1">
    <saml:Conditions
      NotBefore="2024-09-23T22:33:11.683Z"
      NotOnOrAfter="2024-09-24T00:33:11.683Z" />
    <saml:AuthenticationStatement
      AuthenticationInstant="2024-09-23T22:33:11.683Z"
      AuthenticationMethod="urn:oasis:names:tc:SAML:1.0:am:password">
      <saml:Subject>
        <saml:NameIdentifier
          Format="urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified">
UID=112320001+DNQ=300757, OU=A2A, O=US Government, C=US
        </saml:NameIdentifier>
        <saml:SubjectConfirmation>
          <saml:ConfirmationMethod>urn:oasis:names:tc:SAML:1.0:cm:bearer
        </saml:ConfirmationMethod>
        </saml:SubjectConfirmation>
      </saml:Subject>
    </saml:AuthenticationStatement>
    <saml:AttributeStatement>
      <saml:Subject>
        <saml:NameIdentifier
          Format="urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified">
UID=112320001+DNQ=300757, OU=A2A, O=US Government, C=US
        </saml:NameIdentifier>
        <saml:SubjectConfirmation>
          <saml:ConfirmationMethod>urn:oasis:names:tc:SAML:1.0:cm:bearer
        </saml:ConfirmationMethod>
        </saml:SubjectConfirmation>
      </saml:Subject>
      <saml:Attribute AttributeName="SMSESSION"
AttributeNamespace="http://www.forumsys.com/sentry">
        <saml:AttributeValue>jdh00oaGPo7W0bL5ryXBpmY1RXGS
        </saml:AttributeValue>
      </saml:Attribute>
    </saml:AttributeStatement>
    <ds:Signature
      xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
      <ds:SignedInfo>
        <ds:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-
exc-c14n#" />
        <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-
more#rsa-sha256" />
        <ds:Reference URI="#id-5eb4861e4c447f1b358600dfaa605215e72cd6b2">

```



```

    <ds:Transforms>
      <ds:Transform Algorithm="http://www.w3.org/2000/09/xmlldsig#enveloped-
signature" />
      <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
    </ds:Transforms>
    <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"
/>
    <ds:DigestValue>Z3XeR/fJzxSgbvRdec16qN0it391LgH9p7iR3zcnXs4=
</ds:DigestValue>
  </ds:Reference>
</ds:SignedInfo>
<ds:SignatureValue>
  SignatureValueInBase64
</ds:SignatureValue>
<ds:KeyInfo>
  <ds:X509Data>
    <ds:X509Certificate>
      MIIG+CertificateInBase64
    </ds:X509Certificate>
  </ds:X509Data>
</ds:KeyInfo>
</ds:Signature>
</saml:Assertion>
<u:Timestamp
  xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd">
  <u:Created>2024-09-23T22:33:05.114Z</u:Created>
  <u:Expires>2024-09-23T22:38:05.114Z</u:Expires>
</u:Timestamp>
</o:Security>

```

1.4.4 WS-Security for Strong Authentication Users

Beginning with MeF Release 4.0, WS-Security Strong Authentication is the type of authentication used by A2A Web Services for Strong Authentication users, and only Strong Authentication users have been permitted since January 2017. Strong Authentication users must have a valid X.509 digital security certificate obtained from an IRS-authorized certificate authority (CA) (such as WidePoint or Identrust) and have their certificates stored in the IRS directory by using the Automated Enrollment process. Only those users who have registered with the IRS as Strong Authentication users can use this approach to authenticate their identity when using A2A Web Services. Strong Authentication users are authenticated via their X.509 digital certificate in lieu of a password. They do not specify their Application System ID or password in the wsse:UsernameToken elements of the WS-Security header at Login. However, every A2A service request requires the Application System ID of the user. An AppSysID element (mandatory for all service requests from Strong Authentication users) was added to the MeF Header to satisfy this requirement. (See Section 1.6.11, element MeFHeaderType/AppSysID, for a description.). Login requests for Strong Authentication users must also be signed (For details, see Section 1.4.4.2, WS-Security SignedInfo, Reference, DigestValue, and DigestMethod Elements, and Section 1.4.4.3, WS-Security SignatureValue and SignatureMethod Elements).

NOTE: After a successful Login, Strong Authentication users can still use the SAML assertion security token for request authentication/validation. (For details, see Section 1.4.3.3, SAML Assertion.)

Table 1-1, Strong Authentication Request Message Elements, shows the WS-Security elements contained in an MeF A2A request message from Strong Authentication users, and includes a brief explanation of what



each element contains. The sections that follow illustrate the elements using an example ETINRetrieval request message.

Table 1.4-1: Strong Authentication Request Message Elements

Message Element	Contents/Notes
<wsse:BinarySecurityToken ...>	The X.509 Certificate in Base 64 Binary. It contains the Public Key (and Owner ID and Issuer ID)
<ds:SignedInfo> ... <ds:Reference URI="#...">	<ds:Reference URI=...> indicates the ID of one of the elements in the message which is being signed. For example: A <ds:Reference URI="#myHdr"> indicates that the message element with the ID ="myHdr" is being signed by this <ds:Reference URI=...>
<ds:SignedInfo> ... <ds:DigestValue>	<ds:DigestValue> is the Base 64 Binary output of the <ds:DigestMethod Algorithm=...> run on the entire element referenced by the <ds:Reference URI="#...."> (See example below)
<ds:Signature>	Contains information about what is being signed, the signature, the keys used to create the signature, and a place to store arbitrary information
<ds:SignatureValue>	Is the digital signature for the <ds:SignedInfo> element. Rivest, Sharmir and Adelman (RSA) and Elliptic Curve (ECDSA) signatures are allowed. RSA examples are used in this document. Note that a ECDSA certificate cannot be used with an RSA signature and vice versa



1.4.4.1 WS-Security BinarySecurityToken Element

An illustration of the `wsse:BinarySecurityToken` in an A2A service request for a Strong Authentication user is shown below. The `BinarySecurityToken` contains the X.509 Certificate in Base 64 Binary, and is referred to by the `ds:KeyInfo` structure in the `ds:Signature`.



```

<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/
XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<SOAP-ENV:Header>
<ns1:MeFHeader xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="id-1214941501">
<ns1:MessageID>12345202018200000456</ns1:MessageID>
<ns1:Action>Login</ns1:Action>
<ns1:MessageTs>2020-06-30T15:25:50.789Z</ns1:MessageTs>
<ns1:ETIN>12345</ns1:ETIN>
<ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
<ns1:TestCd>P</ns1:TestCd>
<ns1:AppSysID>65432190</ns1:AppSysID>
<ns1:WSDLVersionNum>10.3</ns1:WSDLVersionNum>
<ns1:ClientSoftwareTxt>SOATest</ns1:ClientSoftwareTxt>
</ns1:MeFHeader> <wss:Security xmlns:wss="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-
1.0.xsd">
<wss:BinarySecurityToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
EncodingType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0#Base64Binary"
ValueType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0#X509v3" wsu:Id="CertId-
1673181727">MIIHAbcdefghijklMNOPQRSTUVWXYZ0123459...Bwn71P</wss:BinarySecurityToken>
<ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
<ds:SignedInfo>
<ds:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"></ds:CanonicalizationMethod>
<ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256"></ds:SignatureMethod>
<ds:Reference URI="#id-1214941501">
<ds:Transforms>
<ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"></ds:Transform>
</ds:Transforms>
<ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#sha256"></ds:DigestMethod>
<ds:DigestValue>abcdefghijklmnopqrstuvwxy012345Em0o3VEOTck=</ds:DigestValue>
</ds:Reference>
<ds:Reference URI="#id-1871558655">
<ds:Transforms>
<ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"></ds:Transform>
</ds:Transforms>
<ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#sha256"></ds:DigestMethod>
<ds:DigestValue>abcdefghijklmnopqrstuvwxy012345Xm5ViA+royg=</ds:DigestValue>
</ds:Reference>
</ds:SignedInfo>
<ds:SignatureValue>
abcdefghijklmnopqrstuvwxy012345Vseb1EB/LtVBAoa8dq4GLC+ODTwaONh1uxtJfEq4ttUw
abcdefghijklmnopqrstuvwxy012345q/ZubZA4tKhf/7+GUO+AB0noCkdL4QuqJjE4zqCDsCYC
abcdefghijklmnopqrstuvwxy012345zOxoosMEIEO+fYxmrr3THAogr2qxSWq8J5PNdnlDEXHI
abcdefghijklmnopqrstuvwxy012345Hh/3TAOsRzcValb+m0cvayq/wApqzyfNilFWueHogOwr
abcdefghijklmnopqrstuvwxy012345oj5Nbw==
</ds:SignatureValue>
<ds:KeyInfo Id="KeyId-256137097">
<wss:SecurityTokenReference xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
wsu:Id="STRId-1862925355"><wss:Reference URI="#CertId-1673181727"
ValueType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0#X509v3">
</wss:Reference></wss:SecurityTokenReference>
</ds:KeyInfo>
</ds:Signature></wss:Security></SOAP-ENV:Header>
<SOAP-ENV:Body xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
wsu:Id="id-1871558655">
<LoginRequest xmlns="http://www.irs.gov/a2a/mef/MeFMSIServices.xsd"></LoginRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

wss:BinarySecurityToken contains the signing X.509 Certificate in Base64, without framing lines. The certificate contains the public key, issuer name, and its owner's name.

MeF-03-11-004



1.4.4.2 WS-Security Signed Info, Reference, DigestValue, and DigestMethod Elements

The `ds:SignedInfo`, `ds:Reference`, `ds:DigestValue`, and `ds:DigestMethod` elements in an A2A service request for a Strong Authentication user are illustrated below. The `SignedInfo` element contains the `Reference` and `DigestValue` elements. The `Reference` element references the ID attribute of one of the elements in the message which is being signed. In a signed A2A service request, *there must be at least two `Reference` elements present in each signature; one must point to the ID attribute of the MeF Header, and the other must point to the ID attribute of the SOAP Body. Other `Reference` elements may be added if the developer chooses, but they are not required.* The `DigestValue` element contains the Base 64 Binary output of the `DigestMethod` algorithm run on the entire element referenced by the `Reference` element after it has been processed by the algorithms given by the `Transforms` element, which for references to XML is typically the same as the `CanonicalizationMethod`.



```

<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/
XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<SOAP-ENV:Header>
<ns1:MeFHeader xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" wsu:Id="#id-Hdr">
<ns1:MessageID>12345202018200000234</ns1:MessageID>
<ns1:Action>Login</ns1:Action>
<ns1:MessageTs>2020-06-30T15:25:42.678Z</ns1:MessageTs>
<ns1:ETIN>12345</ns1:ETIN>
<ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
<ns1:TestCd>P</ns1:TestCd>
<ns1:AppSysID>65432190</ns1:AppSysID>
<ns1:WSDLVersionNum>10.3</ns1:WSDLVersionNum>
<ns1:ClientSoftwareTxt>SOATest</ns1:ClientSoftwareTxt>
</ns1:MeFHeader>
<wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
<wsse:BinarySecurityToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
EncodingType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0#Base64Binary"
ValueType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0#X509v3" wsu:Id="CertId-
1673181727">
MIIHAbcdefghijklMNOPQRSTUVWXYZ0123459w0BAQsF...
h81dABCDEFGF/jklmn5Bwn71P</wsse:BinarySecurityToken>
<ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
<ds:SignedInfo>
<ds:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"></ds:CanonicalizationMethod>
<ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256"></ds:SignatureMethod>
<ds:Reference URI="#id-Hdr">
<ds:Transforms>
<ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"></ds:Transform>
</ds:Transforms>
<ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#sha256"></ds:DigestMethod>
<ds:DigestValue>abcdefghijklMNOPQRSTUVWXYZ012345Em0o3VEOTck=</ds:DigestValue>
</ds:Reference>
<ds:Reference URI="#id-Body">
<ds:Transforms>
<ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"></ds:Transform>
</ds:Transforms>
<ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#sha256"></ds:DigestMethod>
<ds:DigestValue>abcdefghijklMNOPQRSTUVWXYZ012345Xm5ViA+royg=</ds:DigestValue>
</ds:Reference>
</ds:SignedInfo>
<ds:SignatureValue>
abcdefghijklMNOPQRSTUVWXYZ012345Vseb1EB/LtVBAoa8dq4GLC+ODTwaONh1uxtJfEq4ttUw
abcdefghijklMNOPQRSTUVWXYZ012345q/ZubZA4tKhf/7+GUO+AB0noCkdL4QuqJJE4zqCDsCYC
abcdefghijklMNOPQRSTUVWXYZ012345zOxoosMEIEO+fYxmrr3THAoqr2qxSWq8J5PNdnIDExHI
abcdefghijklMNOPQRSTUVWXYZ012345Hh/3TAOsRzcValb+m0cvayq/wApqzyfNilFWueHogOwr
abcdefghijklMNOPQRSTUVWXYZ012345oj5Nbw==
</ds:SignatureValue>
<ds:KeyInfo Id="KeyId-256137097">
<wsse:SecurityTokenReference xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
wsu:Id="STRId-1862925355"><wsse:Reference URI="#CertId-1673181727"
ValueType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0#X509v3">
</wsse:Reference></wsse:SecurityTokenReference>
</ds:KeyInfo>
</ds:Signature></wsse:Security>
</SOAP-ENV:Header>
<SOAP-ENV:Body xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
wsu:Id="#id-Body">
<LoginRequest xmlns="http://www.irs.gov/a2a/mef/MeFMSIServices.xsd"></LoginRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

ds:DigestValue is the output of the ds:DigestMethod Algorithm run on the full referenced element, converted to base 64

There must be at least two ds:Reference structures inside ds:SignedInfo, one pointing at the MeF Header and another at the SOAP body. Other ds:Reference's are optional.



1.4.4.3 WS-Security SignatureValue and SignatureMethod Elements

The `ds:SignatureValue` and `ds:SignatureMethod` elements in an A2A service request for a Strong Authentication user are explained below. The `SignatureValue` is the digital signature for the `<ds:SignedInfo>` element converted to base 64, after the `ds:SignedInfo` element has been processed by the algorithm given in `ds:CanonicalizationMethod`. RSA and ECDSA signatures are allowed. RSA examples are used in this document. The `SignatureMethod` specifies the Digest Algorithm (such as sha256 or sha384) used by the signature algorithm as well as the signature algorithm itself.



1.5 SOAP Body

The SOAP Body contains the request message and any attachment provided by the state or transmitter application. For responses, the SOAP body contains the response message and any attachment or a SOAP Fault returned by the service.

1.5.1 Message Attachments Zip File Format

Only a single zip file will be accepted as an attachment sent by a Transmitter or State application and only a single zip file will be returned as an attachment by a service. This zip file attachment is a container zip file compatible with the PKWARE Version 6.2.0 specification that contains one or more data items (submissions, acknowledgements, notifications, statuses, or receipts). For Transmitters, each data item must always be within its own inner zip file. The inner zip file(s) should be compressed; the outer container zip file must be an uncompressed zip file. For States, each data item is just a file within the outer uncompressed zip file. Beginning with Release 9.0, the name of a binary attachment zip file must be less than or equal to 64 characters.

1.5.2 Message Attachments Content Type

The attachments sent by Transmitters and States must have the attachment content type specified as type “application/octet-stream.” The Multimedia Internet Message Extensions (MIME) Web Services Description Language (WSDL) files all specify this in the `mime:content` specification for the attachments. However, the WSDL syntax for Message Transmission Optimization Mechanism (MTOM) does not allow for such a specification. Nevertheless, when sending in a request that includes an attachment, MTOM users must set the attachment content type to “application/octet-stream.” Beginning with MeF Release 8, the Direct Internet Message Encapsulation (DIME) transport mechanism is no longer supported.

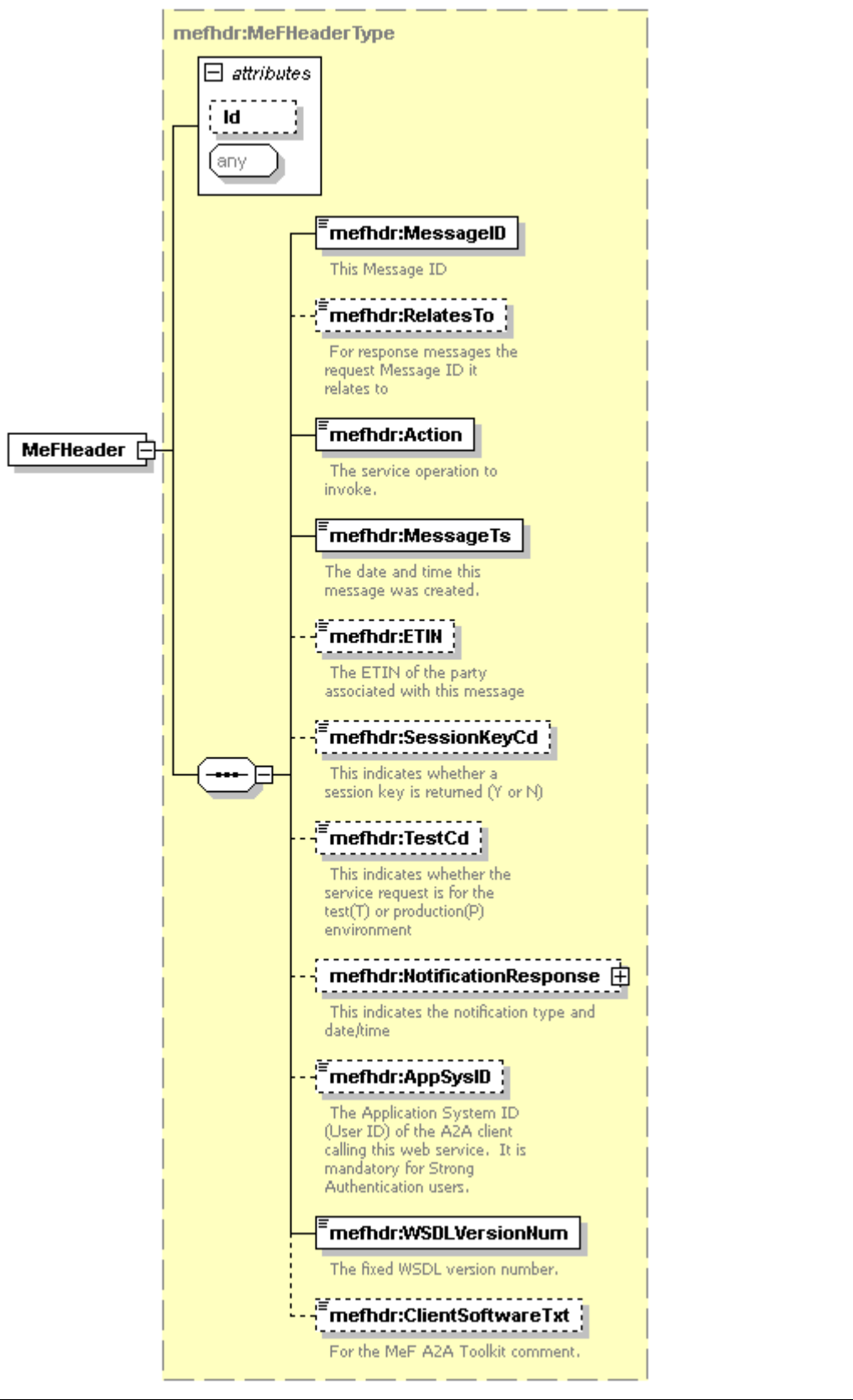


1.6 MeFHeader Elements

This section describes the elements in the MeFHeader message. Every SOAP message request and response for a Web Service from and to a State and Third-Party transmitter on the A2A channel must contain a MeFHeader message in the SOAP message header with the element tag "MeFHeader". The elements in the MeFHeader message are illustrated below. The individual elements are described in the sections that follow. Note that for Release 10.8 a mandatory "WSDLVersionNum" element has a fixed value of 10.8. Any request without this element and value will be rejected by the MeF A2A interface.



diagram



namespace	http://www.irs.gov/a2a/mef/MeFHeader.xsd					
type	mefhdr:MeFHeaderType					
properties	content complex					
children	mefhdr:MessageID mefhdr:RelatesTo mefhdr:Action mefhdr:MessageTs mefhdr:ETIN mefhdr:SessionKeyCd mefhdr:TestCd mefhdr:NotificationResponse mefhdr:AppSysID mefhdr:WSDLVersionNum					
attributes	Name	Type	Use	Default	Fixed	Annotation
	Id	xs:ID	optional			
source	<code><xs:element name="MeFHeader" type="mefhdr:MeFHeaderType"/></code>					

1.6.1 element MeFHeaderType/MessageID

The MessageID element is mandatory. All request and response messages must have a globally unique MessageID provided by the message source.

Table 1.6.6-1 Element MessageID Details

diagram	
type	mefhdr:HeaderMessageIDType
properties	isRef 0 content simple
facets	pattern [0-9]{12}[a-z0-9]{8}[E R D]?
annotation	documentation This Message ID
source	<pre><xs:element name="MessageID" type="mefhdr:HeaderMessageIDType"> <xs:annotation> <xs:documentation> This Message ID </xs:documentation> </xs:annotation> </xs:element> <xs:simpleType name="HeaderMessageIDType"> <xs:restriction base="xs:string"> <xs:pattern value="[0-9]{12}[a-z0-9]{8}[E R D]?"/> </xs:restriction> </xs:simpleType></pre>

Message ID

There are three types of messages to consider: request messages, response messages, and error messages. The message ID formats are specified as follows.



Request Message ID

To ensure the global uniqueness of a message ID, the following format is adopted for the request messages sent to the MeF system:

ETIN + ccyyddd + xxxxxxxx

For example:

Request message ID: 001302006073a2345any

The first five digits (00130) contain the Electronic Transmitter Identification Number (ETIN), the next four digits (2006) contain the year, the next three digits (073) contain the Julian date, and the last eight characters (a2345any) contain a lowercase alphanumeric sequence to uniquely identify messages sent within a day with the given ETIN. Please note that uppercase alphabetic characters are not allowed. The total number of characters of the request message ID is 20. Beginning with Release 9.0, the Request Message ID in the MeF Header will receive strict validation. The first five digits in the Message ID must match the ETIN provided with the request. The Message ID must be 20 positions in length consisting of 12 digits followed by 8 alphanumeric characters (only lower-case alphabetic characters are allowed).

Response Message ID

For response messages sent from the MeF system, the following format is adopted for the message ID indicating that the request was processed successfully:

Request Message ID + “R”

For example:

Response message ID: 001302006073a2345anyR

The total number of characters of the response message ID is 21.

Error Message ID

For error messages sent from the MeF system, the following format is adopted for the message ID:

Request Message ID + “E”

For example:

Response message ID: 001302006073a2345anyE

The total number of characters of the error message ID is 21.

1.6.2 element MeFHeaderType/RelatesTo

This element can be used to allow members in a Web Services engagement—either requesters or responders—to deliver a complex chain of messages between two, three, or more services by positively indicating that they are part of the same thread. The RelatesTo element is optional and is not applicable to the initial message of a thread. All other messages of a thread must contain the RelatesTo element with the MessageID of the initial message.



Table 1.6.6-2 Element RelatesTo Details

diagram	
type	mefhdr:HeaderMessageIdType
properties	isRef 0 content simple
facets	pattern [0-9]{12}[a-z0-9]{8}[E R D]?
annotation	documentation For response messages the request Message ID it relates to
Source	<pre> <xs:element name="RelatesTo" type="mefhdr:HeaderMessageIdType" minOccurs="0"> <xs:annotation> <xs:documentation> For response messages the request Message ID it relates to </xs:documentation> </xs:annotation> </xs:element> <xs:simpleType name="HeaderMessageIdType"> <xs:restriction base="xs:string"> <xs:pattern value="[0-9]{12}[a-z0-9]{8}[E R D]?"/> </xs:restriction> </xs:simpleType> </pre>

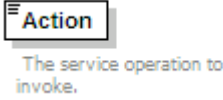
1.6.3 element MeFHeaderType/Action

The Action element is mandatory, and identifies the Web Service endpoints to be used for end-to-end endpoint identification in messages. For all service requests, this element must match the service at the URL invoked. The Action element can contain the actual URL invoked, the SOAP Action, or just the service name.

For example, if invoking the URL <https://la.www4.irs.gov/a2a/mef/Login>, then the Action element could be "https://la.www4.irs.gov/a2a/mef/Login", or "http://www.irs.gov/a2a/mef/Login", or "Login".




Table 1.6.6-3 Element Action Details

diagram	
type	xs:string
properties	isRef 0 content simple
annotation	documentation The service operation to invoke.
source	<pre> <xs:element name="Action" type="xs:string"> <xs:annotation> <xs:documentation> The service operation to invoke. </xs:documentation> </xs:annotation> </xs:element> </pre>

1.6.4 element MeFHeaderType/MessageTs

The MessageTs element is mandatory. The MessageTs element identifies the Web Service date and time of message creation.

Table 1.6.6-4 Element MessageTs Details

diagram	
type	xs:dateTime
properties	isRef 0 content simple
annotation	documentation The date and time this message was created.
source	<pre> <xs:element name="MessageTs" type="xs:dateTime"> <xs:annotation> <xs:documentation>The date and time this message was created.</xs:documentation> </xs:annotation> </xs:element> </pre>

- (1) For any request with a **BinarySecurityToken**, the timestamp value in the MessageTs element in the MeF Header must be within a specific time interval (presently plus or minus 9 minutes) of the MeF system time. Requests with a MessageTs outside the 9-minute range will be rejected with a SOAP fault message indicating “The Timestamp in the Request message Header is not within the allowable range



of current time.” And any SAML assertion present in the request will be invalidated as though you performed a Logout request.

- (2) For Strong Authentication users, the following services **require** a **BinarySecurityToken: Login** (always) and all **Single-Step** services. Single-Step services are those MSI and “Get” services that perform a combined login function. If an MSI or “Get” service is not performing a combined login function (i.e., it contains a SAML assertion), the **BinarySecurityToken should not** be present in the service request.
- (3) All “Send” type services **should never** contain a **BinarySecurityToken**: Please note that the MessageTs Interval check is NOT performed on any “Send” type requests.
- (4) The MessageTs format must conform to the xsd:dateTime data type defined in *XML Schema Part 2: Datatypes Second Edition* Section 3.2.7 (see <http://www.w3.org/TR/xmlschema-2/>). Examples of valid MessageTs values are shown below. Please note that all the time zones in the United States are represented by negative time offsets and that the colon (:) is required in the time offset syntax.

xsd:dateTime String	Time Represented
2009-01-20T11:30:18.045-05:00	January 20, 2009 11:30:18.045 am Eastern Standard Time
2009-01-20T08:30:18.045-08:00	January 20, 2009 08:30:18.045 am Pacific Standard Time
2009-01-20T16:30:18.045Z	January 20, 2009 04:30:18.045 pm Coordinated Universal Time (UTC) or GMT
2009-01-20T16:30:18.045	January 20, 2009 04:30:18.045 pm UTC or GMT (NOTE: No Z implies UTC)
	The four times above all represent the same moment on January 20, 2009
2009-07-04T12:35:00-04:00	July 4, 2009 12:35 pm Eastern Daylight Time
2009-07-04T09:35:00-07:00	July 4, 2009 09:35 am Pacific Daylight Time
2009-07-04T16:35:00Z	July 4, 2009 04:35 pm UTC or GMT
2009-07-04T16:35:00	July 4, 2009 04:35 pm UTC or GMT (NOTE: No Z implies UTC)
	The four times above all represent the same moment on July 4, 2009

1.6.5 element MeFHeaderType/ETIN

The ETIN element identifies the electronic transmitter identification number of the transmitter who sends the request message. The ETIN element is optional. It is not applicable to the Login, or Logout services. However, it is required for all other service requests.

Table 1.6.6-5 Element ETIN Details

diagram	
---------	---




type	xs:string
properties	isRef 0 content simple
annotation	documentation The ETIN of the party associated with this message
source	<pre><xs:element name="ETIN" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation> The ETIN of the party associated with this message </xs:documentation> </xs:annotation> </xs:element></pre>

1.6.6 element MeFHeaderType/SessionKeyCd

The SessionKeyCd element indicates whether the service request should return an open session (SAML assertion) for the request, or close the session at the end of the service (no SAML). The allowed values are “Y” (to indicate that an open persistent session should be established), or “N” (to indicate that the session should be closed [logged out] at the end of the service). The SessionKeyCd element is optional because it is not applicable to the Logout service. However, it is required for all other services.

Table 1.6.6-6 Element SessionKeyCd Details

Diagram	
Type	mefhdr: SessionKeyCdType
Properties	isRef 0 content simple
Annotation	documentation This indicates whether a session key is returned (“Y” or “N”)
Source	<pre><xs:element name="SessionKeyCd" type="mefhdr:SessionKeyCdType" minOccurs="0"> <xs:annotation> <xs:documentation> This indicates whether a session key is returned (“Y” or “N”) </xs:documentation> </xs:annotation> </xs:element></pre>



	<pre> <xs:simpleType name="SessionKeyCdType"> <xs:restriction base="xs:string"> <xs:enumeration value="Y"/> <xs:enumeration value="N"/> </xs:restriction> </xs:simpleType> </pre>
--	---

1.6.7 element MeFHeaderType/TestCd

The TestCd element identifies if the request is a production request or an ATS (test) request. The allowed values are “P” (to indicate that this is a production request) or “T” (to indicate that this is an ATS (test) request). The TestCd element is optional because it is not applicable to the Login, or Logout services. However, it is required for all other service requests.

Table 1.6.6-7 Element TestCd Details

Diagram	
Type	mefhdr:TestCdType
Properties	isRef 0 content simple
Annotation	documentation This indicates whether the service request is for the test(T) or production(P) environment
Source	<pre> <xs:element name="TestCd" type="mefhdr:TestCdType" minOccurs="0"> <xs:annotation> <xs:documentation> This indicates whether the service request is for the test(T) or production(P) environment </xs:documentation> </xs:annotation> </xs:element> <xs:simpleType name="TestCdType"> <xs:restriction base="xs:string"> <xs:enumeration value="T"/> <xs:enumeration value="P"/> </xs:restriction> </xs:simpleType> </pre>



1.6.8 element MeFHeaderType/NotificationResponse

The optional NotificationResponse element is used to communicate secondary information back to the Third Party or State applications. It is intended for use on response messages only, not on input requests. This information is not necessarily directly related to the request. For example, it is used to notify the application that the X509 Certificate will expire after some number of days.

Table 1.6.6-8 Element NotificationResponse Details

Diagram	
Type	mefhdr:MeFNotificationResponse
Properties	isRef 0 content complex
Children	NotificationDesc NotificationDt
Annotation	documentation This indicates the notification type and date/time
Source	<pre><xs:element name="NotificationResponse" type="mefhdr:MeFNotificationResponse" minOccurs="0"> <xs:annotation> <xs:documentation> This indicates the notification type and date/time </xs:documentation> </xs:annotation> </xs:element></pre>

1.6.9 element MeFHeaderType/NotificationDesc

The NotificationDesc element is used to communicate secondary information back to the Third Party or State applications (e.g., "X.509 Certificate Expires in 7 days").

Table 1.6.6-9 Element NotificationDesc Details

Diagram	
---------	--




Type	xs:string
Properties	isRef 0 content simple
Annotation	documentation The item to which this notification applies (e.g. X.509 Certificate Expires in 7 days)
Source	<pre><xs:element name="NotificationDesc" type="xs:string"> <xs:annotation> <xs:documentation> The item to which this notification applies (e.g. X.509 Certificate Expires in 7 days) </xs:documentation> </xs:annotation> </xs:element></pre>

1.6.10 element MeFHeaderType/NotificationDesc

The NotificationDt element is used to communicate the date and time of the notification back to the Third Party or State applications. It is always the current system date and time.

Table 1.6.6-10 Element NotificationDt Details


Diagram	 <pre>mefhdr:NotificationDt</pre> <p>The current system date and time of this notification.</p>
Type	xs:dateTime
Properties	isRef 0 content simple
Annotation	documentation The current system date and time of this notification.
Source	<pre><xs:element name="NotificationDt" type="xs:dateTime"> <xs:annotation> <xs:documentation> The current system date and time of this notification. </xs:documentation> </xs:annotation> </xs:element></pre>

1.6.11 element MeFHeaderType/AppSysID

The AppSysID element identifies the Application System ID (username) of the A2A client calling this Web Service. The AppSysID element is optional. It is mandatory for all requests from Strong Authentication users.



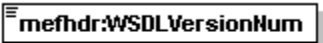
Table 1.6.6-11 Element AppSysID Details

Diagram	 <p>The Application System ID (User ID) of the A2A client calling this web service. It is mandatory for Strong Authentication users.</p>
Namespace	http://www.irs.gov/a2a/mef/MeFHeader.xsd
Type	xs:string
Properties	isRef 0 content simple
Annotation	documentation The Application System ID (User ID) of the A2A client calling this Web service. It is mandatory for Strong Authentication users.
Source	<pre><xs:element name="AppSysID" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation> The Application System ID (User ID) of the A2A client calling this Web service. It is mandatory for Strong Authentication users. </xs:documentation> </xs:annotation> </xs:element></pre>

1.6.12 element MeFHeaderType/WSDLVersionNum

The WSDLVersionNum element identifies the fixed WSDL version number for all MeF A2A messages from clients calling Web Services. Any request without this element and value will be rejected by the MeF A2A interface.

Table 1.6.6-12 Element WSDLVersionNum Details

diagram	 <p>The fixed WSDL version number.</p>
namespace	http://www.irs.gov/a2a/mef/MeFHeader.xsd
type	xs:string
properties	isRef 0 content simple fixed 10.8
annotation	documentation The fixed WSDL version number.




source	<pre> <xs:element name=" WSDLVersionNum" type="xs:string" fixed="10.8"> <xs:annotation> <xs:documentation> The fixed WSDL version number. </xs:documentation> </xs:annotation> </xs:element> </pre>
--------	---

1.6.13 element MeFHeaderType/ClientSoftwareTxt

The ClientSoftwareTxt element is for use by the MeF A2A Toolkit. It is optional and should not set outside of the Toolkit software.

Table 1.6.6-13 Element ClientSoftwareTxt Details

Diagram	
Namespace	http://www.irs.gov/a2a/mef/MeFHeader.xsd
Type	xs:string
Properties	isRef 0 content simple
Source	<pre> <xs:element name="ClientSoftwareTxt" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation> For the MeF A2A Toolkit comment. </xs:documentation> </xs:annotation> </xs:element> </pre>

1.6.14 attribute Id

The Id attribute is used with Strong Authentication to identify the MeF Header referenced in a SignedInfo/Reference used to generate a XML digital signature DigestValue. It is optional and only required in signed requests from Strong Authentication users. It is optional in requests authenticated by SAML token.

1.7 MeF A2A ErrorExceptionDetail Elements

In MeF Release 10.0 the MeFException element is replaced with the ErrorExceptionDetail element with completely renamed child elements. Any faults returned will contain this new structure in the SOAP Fault detail of the SOAP body.



element **ErrorExceptionDetail**

Table 1.7.7-1 Element *ErrorExceptionDetail* Details

Diagram	
Namespace	http://www.irs.gov/a2a/mef/MeFHeader.xsd
Type	mefhdr:ErrorExceptionDetailType
Properties	content Complex
Children	mefhdr:ErrorMessageTxt mefhdr:ErrorClassificationCd mefhdr:ErrorMessageCd mefhdr:HostNm

Table 1.7.7-2 Element *ErrorMessageTxt* Details

Diagram	
Namespace	http://www.irs.gov/a2a/mef/MeFHeader.xsd
Type	xs:string
Diagram	
Namespace	http://www.irs.gov/a2a/mef/MeFHeader.xsd
Type	mefhdr:ErrorClassificationCdType
Facets	enumeration SYSTEM_ERROR enumeration RESOURCE_UNAVAILABLE



	enumeration MESSAGE_FORMAT_ERROR
	enumeration SECURITY_ERROR
	enumeration REQUEST_ERROR

Table 1.7.7-3 Element ErrorMessageCd Details



Diagram	
Namespace	http://www.irs.gov/a2a/mef/MeFHeader.xsd
Type	xs:string

Table 1.7.7-4 Element HostNm Details

Diagram	
Namespace	http://www.irs.gov/a2a/mef/MeFHeader.xsd
Type	xs:string

1.8 MeF A2A Web Services Summary

Each MeF A2A service request must contain a SOAP header, which must contain both an MeF header and a WS-Security header. The first request of a session must be a signed request with a BinarySecurityToken and Signature in the WS-Security header, and it may not contain an attachment (and thus may not be a “Send” operation). It is expected that subsequent requests of a session will contain a SAML assertion and a UsernameToken with a Username and no other elements in the WS-Security header. The SAML assertion must be an exact copy of the SAML assertion returned in the response to a prior request. The value of Username must be the same as the AppSysID in the MeF Header and the request must not contain either a Signature or a BinarySecurityToken. If a SAML assertion is used in a request with a SessionIndicator of “N”, it will be invalidated as if the Logout service was called and may never be used again.

The MeF A2A services are divided into three classifications:

- **MSI Services**
 - Login, Logout, ETINRetrieval, EtinStatus, GetStateParticipantsList
- **State Services MIME, and MTOM**
 - GetNewSubmissions, GetSubmissionsByMsgID, GetSubmissions, GetSubmission, SendSubmissionReceipts, SendAcks, GetNewAckNotifications, GetAckNotifications, GetAckNotification, GetSubmissionReconciliationList
- **Transmitter Services MIME, and MTOM**
 - SendSubmissions, GetNewAcks, GetAcksByMsgID, GetAcks, GetAck, GetNewSubmissionsStatus, GetSubmissionsStatus, GetSubmissionStatus

The URL addresses for accessing the services from the client applications on the A2A channel are in Table 1-2, MeF A2A Web Services URLs. Beginning with MeF Release 6.2 States will be allowed to access the



Get Submissions Status family of Web services (GetSubmissionsStatus and GetSubmissionStatus). To accomplish this the States will need to use the appropriate URLs previously reserved for use by Third Party transmitters.

Table 1.8.8-1: MeF A2A Web Services URLs

Table 1-2. MeF A2A Services URLs
<p>MSI Services</p> <p>https://la.www4.irs.gov/a2a/mef/EtinRetrieval https://la.www4.irs.gov/a2a/mef/EtinStatus https://la.www4.irs.gov/a2a/mef/GetStateParticipantsList https://la.www4.irs.gov/a2a/mef/Logout https://la.www4.irs.gov/a2a/mef/Login</p>
<p>State Services MIME</p> <p>https://la.www4.irs.gov/a2a/mef/mime/GetNewSubmissions https://la.www4.irs.gov/a2a/mef/mime/GetSubmissionsByMsgID https://la.www4.irs.gov/a2a/mef/mime/GetSubmissions https://la.www4.irs.gov/a2a/mef/mime/GetSubmission https://la.www4.irs.gov/a2a/mef/mime/SendSubmissionReceipts https://la.www4.irs.gov/a2a/mef/mime/SendAcks https://la.www4.irs.gov/a2a/mef/mime/GetNewAckNotifications https://la.www4.irs.gov/a2a/mef/mime/GetAckNotifications https://la.www4.irs.gov/a2a/mef/mime/GetAckNotification https://la.www4.irs.gov/a2a/mef/mime/GetSubmissionReconciliationList</p>
<p>State Services MTOM</p> <p>https://la.www4.irs.gov/a2a/mef/mtom/GetNewSubmissions https://la.www4.irs.gov/a2a/mef/mtom/GetSubmissionsByMsgID https://la.www4.irs.gov/a2a/mef/mtom/GetSubmissions https://la.www4.irs.gov/a2a/mef/mtom/GetSubmission https://la.www4.irs.gov/a2a/mef/mtom/SendSubmissionReceipts https://la.www4.irs.gov/a2a/mef/mtom/SendAcks https://la.www4.irs.gov/a2a/mef/mtom/GetNewAckNotifications https://la.www4.irs.gov/a2a/mef/mtom/GetAckNotifications https://la.www4.irs.gov/a2a/mef/mtom/GetAckNotification https://la.www4.irs.gov/a2a/mef/mtom/GetSubmissionReconciliationList</p>
<p>Transmitter Services MIME</p> <p>https://la.www4.irs.gov/a2a/mef/mime/SendSubmissions https://la.www4.irs.gov/a2a/mef/mime/GetNewAcks https://la.www4.irs.gov/a2a/mef/mime/GetAcksByMsgID https://la.www4.irs.gov/a2a/mef/mime/GetAcks https://la.www4.irs.gov/a2a/mef/mime/GetAck</p>



Table 1-2. MeF A2A Services URLs

<p> https://la.www4.irs.gov/a2a/mef/mime/GetNewSubmissionsStatus https://la.www4.irs.gov/a2a/mef/mime/GetSubmissionsStatus https://la.www4.irs.gov/a2a/mef/mime/GetSubmissionStatus https://la.www4.irs.gov/a2a/mef/mime/GetNew2290Schedule1s https://la.www4.irs.gov/a2a/mef/mime/Get2290Schedule1sByMsgID https://la.www4.irs.gov/a2a/mef/mime/Get2290Schedule1s https://la.www4.irs.gov/a2a/mef/mime/Get2290Schedule1 </p> <p>Transmitter Services MTOM</p> <p> https://la.www4.irs.gov/a2a/mef/mtom/SendSubmissions https://la.www4.irs.gov/a2a/mef/mtom/GetNewAcks https://la.www4.irs.gov/a2a/mef/mtom/GetAcksByMsgID https://la.www4.irs.gov/a2a/mef/mtom/GetAcks https://la.www4.irs.gov/a2a/mef/mtom/GetAck https://la.www4.irs.gov/a2a/mef/mtom/GetNewSubmissionsStatus https://la.www4.irs.gov/a2a/mef/mtom/GetSubmissionsStatus https://la.www4.irs.gov/a2a/mef/mtom/GetSubmissionStatus https://la.www4.irs.gov/a2a/mef/mtom/GetNew2290Schedule1s https://la.www4.irs.gov/a2a/mef/mtom/Get2290Schedule1sByMsgID https://la.www4.irs.gov/a2a/mef/mtom/Get2290Schedule1s https://la.www4.irs.gov/a2a/mef/mtom/Get2290Schedule1 </p>
--

The following table lists interface messages for each service.

<i>MeF Service Request and Response Messages for A2A</i>	
MSI Services Messages	
EtinRetrievalRequestMessage	
EtinRetrievalResponseMessage	
EtinStatusRequestMessage	
EtinStatusResponseMessage	
GetStateParticipantsListRequestMessage	
GetStateParticipantsListResponseMessage	
LoginRequestMessage	
LoginResponseMessage	
LogoutRequestMessage	
LogoutResponseMessage	
MeFFaultMessage	
State Services Messages	Transmitter Services Messages
GetAckNotificationRequest	GetAckRequest
GetAckNotificationResponse	GetAckResponse
GetAckNotificationsRequest	GetAcksRequest



<i>MeF Service Request and Response Messages for A2A</i>	
GetAckNotificationsResponse	GetAcksResponse
GetNewAckNotificationsRequest	GetAcksByMsgIDRequest
GetNewAckNotificationsResponse	GetAcksByMsgIDResponse
GetNewSubmissionsRequest	GetNewAcksRequest
GetNewSubmissionsResponse	GetNewAcksResponse
GetSubmissionRequest	GetNewSubmissionsStatusRequest
GetSubmissionResponse	GetNewSubmissionsStatusResponse
GetSubmissionsRequest	GetSubmissionsStatusRequest
GetSubmissionsResponse	GetSubmissionsStatusResponse
GetSubmissionsByMsgIDRequest	GetSubmissionStatusRequest
GetSubmissionsByMsgIDResponse	SendSubmissionsRequest
SendAcksRequest	SendSubmissionsResponse
SendAcksResponse	
SendSubmissionReceiptsRequest	
SendSubmissionReceiptsResponse	
GetSubmissionReconciliationListRequest	
GetSubmissionReconciliationListResponse	

The style used in the following sections to describe the A2A services warrants explanation. Each service section begins with a service description indicating the service intent, the request message and response message elements, and what conditions will result in an error being returned. The subsections that follow contain diagrams illustrating the message elements and sub-elements for the requests, responses, and exceptions (errors). For example:

- Section 1.8.1.2, Login, describes the Login service, the request and response message elements, and the error response
- Subsection 1.8.1.2.1, element LoginRequest, illustrates the Login Request message elements
- Subsection 1.8.1.2.2, element LoginResponse, illustrates the Login Response message elements

1.8.1 Modernized System Infrastructure (MSI) Services

The following table contains the summary of the services, bindings, ports, and SOAP messages for all the A2A Web Services provided directly by the MSI.

Table 1.8.8-2 Modernized System Infrastructure (MSI) Services

services	bindings	prototypes	messages
EtinRetrieval	EtinRetrieval	EtinRetrieval	EtinRetrievalRequestMessage
EtinStatus	EtinStatus	EtinStatus	EtinRetrievalResponseMessage
GetStateParticipantsList	GetStateParticipantsList	GetStateParticipantsList	EtinStatusRequestMessage
Login	Login	Login	EtinStatusResponseMessage
Logout	Logout	Logout	GetStateParticipantsListRequest Message



			GetStateParticipantsListResponse Message LoginRequestMessage LoginResponseMessage LogoutRequestMessage LogoutResponseMessage
--	--	--	---

1.8.1.1 Login

Both States and the Third Party transmitters will need to login to the system in order to invoke any other service requests. This will be accomplished using the infrastructure login Web Service. Strong Authentication users must specify their Application System ID in the AppSysID element of the MeF Header and their **BinarySecurityToken, and Signature** (containing the **SignatureValue**) in the wsse:Security portion of the SOAP header along with an empty LoginRequest message tag in the SOAP body. If the authentication is successful, a LoginResponse message is returned in the SOAP body of the response message. If the authentication is successfully completed and the SessionKeyCd is set to “Y” in the MeFHeader message within the SOAP header, a SAML session token is created and returned in the wsse:Security portion of the SOAP header of the response message along with a site affinity “dc=” cookie in the HTTP header. The wsse:Security SAML session token and the site affinity “dc=” cookie must be used together on all subsequent web services using that SAML session token. If authentication fails, a SOAP error message will be returned to the requesting system.

Table 1.8.8-3 Login Message Details

type	tns:Login
Extensibility	<code><soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/></code>
operations	Login extensibility <code><soap:operation soapAction="http://www.irs.gov/a2a/mef/Login"/></code> input <code><soap:body parts="Login" use="literal"/><soap:header message="tns:LoginRequestMessage" part="MeFHeader" use="literal"/></code> output <code><soap:body parts="LoginResponse" use="literal"/><soap:header message="tns:LoginResponseMessage" part="MeFHeader" use="literal"/></code>
used by	Service Login in Port Login
source	<code><wsdl:binding name="Login" type="tns:Login"></code> <code><soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/></code> <code><wsdl:operation name="Login"></code> <code><soap:operation soapAction="http://www.irs.gov/a2a/mef/Login"/></code> <code><wsdl:input></code> <code><soap:body parts="Login" use="literal"/></code> <code><soap:header message="tns:LoginRequestMessage" part="MeFHeader" use="literal"/></code> <code></wsdl:input></code>



```

<wsdl:output>
  <soap:body parts="LoginResponse" use="literal"/>
  <soap:header message="tns:LoginResponseMessage" part="MeFHeader" use="literal"/>
</wsdl:output>
<wsdl:fault name="fault">
  <soap:fault name="fault" use="literal"/>
</wsdl:fault>
</wsdl:operation>
</wsdl:binding>

```

1.8.1.1.1 element LoginRequest

Table 1.8.8-4 Element LoginRequest Details

Diagram	
Namespace	http://www.irs.gov/a2a/mef/MeFMSIServices.xsd
type	mef:LoginRequestType
properties	content complex
source	<xs:element name="LoginRequest" type="mef:LoginRequestType"/>

1.8.1.1.2 element LoginResponse

Table 1.8.8-5 Element LoginResponse Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFMSIServices.xsd
type	mef:LoginResponseType
properties	content complex



children	mef:StatusTxt
source	<code><xs:element name="LoginResponse" type="mef:LoginResponseType"/></code>

1.8.1.2 Logout

Both States and Third-Party transmitters will have authorization to programmatically request a logout Web Service to end a persistent session using an infrastructure Web Service designed to perform this task. Users must specify their Application System ID in the AppSysID element of the MeF Header, the SAML session key in the wsse:Security portion of the SOAP header and an empty LogoutRequest message tag in the SOAP body (note that Signed Logout requests are not supported). Every Logout request must include the same site affinity “dc=” cookie in the HTTP Header of the Logout request that was returned with the successful Login response wsse:Security SAML session token. If the requesting session is successfully terminated, a successful LogoutResponse message will be returned to the requestor in the SOAP body indicating that the session was terminated. Otherwise, a SOAP error message will be returned to the requesting system.

Table 1.8.8-6 Logout Message Details

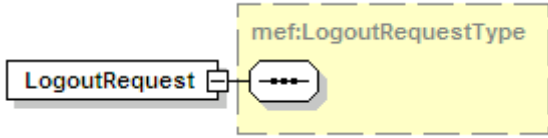
type	tns:Logout
extensibility	<code><soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/></code>
operations	<p>Logout</p> <p>Extensibility <code><soap:operation soapAction="http://www.irs.gov/a2a/mef/Logout"/></code></p> <p>Input <code><soap:body parts="Logout" use="literal"/><soap:header message="tns:LogoutRequestMessage" part="MeFHeader" use="literal"/></code></p> <p>output <code><soap:body parts="LogoutResponse" use="literal"/><soap:header message="tns:LogoutResponseMessage" part="MeFHeader" use="literal"/></code></p>
used by	Service Logout in Port Logout
source	<pre> <wsdl:binding name="Logout" type="tns:Logout"> <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/> <wsdl:operation name="Logout"> <soap:operation soapAction="http://www.irs.gov/a2a/mef/Logout"/> <wsdl:input> <soap:body parts="Logout" use="literal"/> <soap:header message="tns:LogoutRequestMessage" part="MeFHeader" use="literal"/> </wsdl:input> <wsdl:output> <soap:body parts="LogoutResponse" use="literal"/> <soap:header message="tns:LogoutResponseMessage" part="MeFHeader" use="literal"/> </wsdl:output> <wsdl:fault name="fault"> <soap:fault name="fault" use="literal"/> </wsdl:fault> </wsdl:operation> </wsdl:binding> </pre>



	<pre></wsdl:operation></pre> <pre></wsdl:binding></pre>
--	---

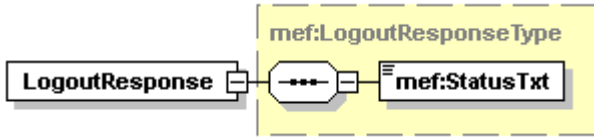
1.8.1.2.1 element LogoutRequest

Table 1.8.8-7 Element LogoutRequest Details

diagram	 <p>The diagram shows a box labeled 'LogoutRequest' connected to a dashed-line box labeled 'mef:LogoutRequestType'. Inside the dashed box is a smaller box containing a dashed-line box labeled 'mef:LogoutRequestType' and a smaller box labeled 'mef:StatusTxt'.</p>
namespace	http://www.irs.gov/a2a/mef/MeFMSIServices.xsd
type	mef:LogoutRequestType
properties	content complex
source	<code><xs:element name="LogoutRequest" type="mef:LogoutRequestType"/></code>

1.8.1.2.2 element LogoutResponse

Table 1.8.8-8 Element LogoutResponse Details

diagram	 <p>The diagram shows a box labeled 'LogoutResponse' connected to a dashed-line box labeled 'mef:LogoutResponseType'. Inside the dashed box is a smaller box containing a dashed-line box labeled 'mef:LogoutResponseType' and a smaller box labeled 'mef:StatusTxt'.</p>
namespace	http://www.irs.gov/a2a/mef/MeFMSIServices.xsd
type	mef:LogoutResponseType
properties	content complex
children	mef:StatusTxt
source	<code><xs:element name="LogoutResponse" type="mef:LogoutResponseType"/></code>

1.8.1.3 ETIN Retrieval

Both States and Third-Party transmitters will have authorization to programmatically request a Web Service to retrieve an array of ETINs for the originating Application System ID (specified in the AppSysID element of the MeF Header in the SOAP header by Strong Authentication users). The requesting system will also provide an ETINRetrievalRequest message in the SOAP body with a Test Indicator (“T” or “P”) indicating whether the list of ETINs should be for the Test or Production environment. If successful, the service returns an ETINRetrievalResponse message in the SOAP body containing an array of ETINs, statuses and forms



for the Application System ID, or null if no ETINs are found. Otherwise, a SOAP error message will be returned to the requesting system. If the request is for Test ETINs, then ETINs where the ETIN status or Form status indicate Test are returned. If the request is for Production ETINs, only ETINs where both the ETIN status and Form status indicate Production are returned.

Table 1.8.8-9 ETIN Retrieval Message Details

type	tns:EtinRetrieval
extensibility	<soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
operations	EtinRetrieval Extensibility <soap:operation soapAction="http://www.irs.gov/a2a/mef/EtinRetrieval"/> Input <soap:body parts="EtinRetrieval" use="literal"/><soap:header message="tns:EtinRetrievalRequestMessage" part="MeFHeader" use="literal"/> output <soap:body parts="EtinRetrievalResponse" use="literal"/><soap:header message="tns:EtinRetrievalResponseMessage" part="MeFHeader" use="literal"/>
used by	Service EtinRetrieval in Port EtinRetrieval
source	<pre> <wsdl:binding name="EtinRetrieval" type="tns:EtinRetrieval"> <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/> <wsdl:operation name="EtinRetrieval"> <soap:operation soapAction="http://www.irs.gov/a2a/mef/EtinRetrieval"/> <wsdl:input> <soap:body parts="EtinRetrieval" use="literal"/> <soap:header message="tns:EtinRetrievalRequestMessage" part="MeFHeader" use="literal"/> </wsdl:input> <wsdl:output> <soap:body parts="EtinRetrievalResponse" use="literal"/> <soap:header message="tns:EtinRetrievalResponseMessage" part="MeFHeader" use="literal"/> </wsdl:output> <wsdl:fault name="fault"> <soap:fault name="fault" use="literal"/> </wsdl:fault> </wsdl:operation> </wsdl:binding> </pre>

1.8.1.3.1 element EtinRetrievalRequest



Table 1.8.8-10 Element ETINRetrievalRequest Details

namespace	http://www.irs.gov/a2a/mef/MeFMSIServices.xsd
type	mef:ETINRetrievalRequestType
properties	content complex
children	mef:TestCd
source	<code><xs:element name="ETINRetrievalRequest" type="mef:ETINRetrievalRequestType"/></code>

Table 1.8.8-11 Element TestCdType Details

namespace	http://www.irs.gov/a2a/mef/MeFMSIServices.xsd
type	restriction of xs:string
used by	element ETINRetrievalRequestType/TestCd
facets	enumeration T enumeration P
source	<pre> <xs:simpleType name="TestCdType"> <xs:restriction base="xs:string"> <xs:enumeration value="T"/> <xs:enumeration value="P"/> </xs:restriction> </xs:simpleType> </pre>

1.8.1.3.2 element ETINRetrievalResponse

Table 1.8.8-12 Element ETINRetrievalResponse Details

namespace	http://www.irs.gov/a2a/mef/MeFMSIServices.xsd



type	mef:ETINRetrievalResponseType
properties	content complex
children	mef:Cnt mef:ETINStatusGrp
source	<code><xs:element name="ETINRetrievalResponse" type="mef:ETINRetrievalResponseType"/></code>

Table 1.8.8-13 Element ETINStatusGrp Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFMSIServices.xsd
type	mef:ETINStatusGrpType
properties	isRef 0 content complex
children	mef:ETIN mef:StatusTxt mef:FormStatusGrp
source	<code><xs:element name="ETINStatusGrp" type="mef:ETINStatusGrpType" minOccurs="0" maxOccurs="unbounded"/></code>

Table 1.8.8-14 Element FormStatusGrp Details

diagram	
type	mef:FormStatusGrpType
properties	isRef 0 content complex
children	mef:FormNum mef:FormStatusTxt
source	<code><xs:element name="FormStatusGrp" type="mef:FormStatusGrpType" minOccurs="0" maxOccurs="unbounded"/></code>



1.8.1.4 ETIN Status

Both States and Third-Party transmitters will have authorization to programmatically request a Web Service to retrieve the status of an ETIN for an originating Application System ID. The requesting system will provide the Application System ID (specified in the AppSysID element of the MeF Header in the SOAP header by Strong Authentication users) and an EtinStatusRequest message in the SOAP body containing the ETIN. If successful, the service returns an EtinStatusResponse message in the SOAP body containing the ETIN, its status and a list of forms for the requested ETIN, or null if the ETIN is not found. Otherwise, a SOAP error message will be returned to the requesting system.

Table 1.8.8-15 ETinStatus Message Details

type	tns:EtinStatus
extensibility	<soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
operations	<p>EtinStatus</p> <p>Extensibility <soap:operation soapAction="http://www.irs.gov/a2a/mef/EtinStatus"/></p> <p>Input <soap:body parts="EtinStatus" use="literal"/><soap:header message="tns:EtinStatusRequestMessage" part="MeFHeader" use="literal"/></p> <p>output <soap:body parts="EtinStatusResponse" use="literal"/><soap:header message="tns:EtinStatusResponseMessage" part="MeFHeader" use="literal"/></p>
used by	Service EtinStatus in Port EtinStatus
source	<pre> <wsdl:binding name="EtinStatus" type="tns:EtinStatus"> <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/> <wsdl:operation name="EtinStatus"> <soap:operation soapAction="http://www.irs.gov/a2a/mef/EtinStatus"/> <wsdl:input> <soap:body parts="EtinStatus" use="literal"/> <soap:header message="tns:EtinStatusRequestMessage" part="MeFHeader" use="literal"/> </wsdl:input> <wsdl:output> <soap:body parts="EtinStatusResponse" use="literal"/> <soap:header message="tns:EtinStatusResponseMessage" part="MeFHeader" use="literal"/> </wsdl:output> <wsdl:fault name="fault"> <soap:fault name="fault" use="literal"/> </wsdl:fault> </wsdl:operation> </wsdl:binding> </pre>



1.8.1.4.1 element ETINStatusRequest

Table 1.8.8-16 Element ETINStatusRequest Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFMSIServices.xsd
type	mef:ETINStatusRequestType
properties	content complex
children	ETIN
source	<code><xs:element name="ETINStatusRequest" type="mef:ETINStatusRequestType"/></code>

1.8.1.4.2 element ETINStatusResponse

Table 1.8.8-17 Element ETINStatusResponse Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFMSIServices.xsd
type	mef:ETINStatusResponseType
properties	content complex
children	mef:ETIN mef:StatusTxt mef:FormStatusGrp
source	<code><xs:element name="ETINStatusResponse" type="mef:ETINStatusResponseType"/></code>



ETINStatusResponseType/FormStatusGrp

Table 1.8.8-18 Element FormStatusGrp Details

diagram	
type	mef:FormStatusGrpType
properties	isRef 0 content complex
children	mef:FormNum mef:FormStatusTxt
source	<code><xs:element name="FormStatusGrp" type="mef:FormStatusGrpType" minOccurs="0" maxOccurs="unbounded"/></code>

1.8.1.5 Get State Participants List

Both States and Third-Party transmitters will have authorization to programmatically request a Web Service to retrieve an array of State participants. Even though there are no formal input arguments to this service request, the requestor must provide an empty GetStateParticipantsListRequest message tag in the SOAP body. If successful, the service returns a GetStateParticipantsListResponse message in the SOAP body containing an array of State names and participant programs (Doing Business As indicators) or null if no State participants are found. Otherwise, a SOAP error message will be returned to the requesting system.

Table 1.8.8-19 GetStateParticipantsList Message Details

type	tns:GetStateParticipantsList
extensibility	<code><soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/></code>
operations	GetStateParticipantsList Extensibility <code><soap:operation soapAction="http://www.irs.gov/a2a/mef/GetStateParticipantsList"/></code> Input <code><soap:body parts="GetStateParticipantsList" use="literal"/><soap:header message="tns:GetStateParticipantsListRequestMessage" part="MeFHeader" use="literal"/></code> output <code><soap:body parts="GetStateParticipantsListResponse" use="literal"/><soap:header message="tns:GetStateParticipantsListResponseMessage" part="MeFHeader" use="literal"/></code>
used by	Service GetStateParticipantsList in Port GetStateParticipantsList
source	<code><wsdl:binding name="GetStateParticipantsList" type="tns:GetStateParticipantsList"></code> <code><soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/></code> <code><wsdl:operation name="GetStateParticipantsList"></code> <code><soap:operation soapAction="http://www.irs.gov/a2a/mef/GetStateParticipantsList"/></code> <code><wsdl:input></code>



```

<soap:body parts="GetStateParticipantsList" use="literal"/>
<soap:header message="tns:GetStateParticipantsListRequestMessage" part="MeFHeader" use="literal"/>
</wsdl:input>
<wsdl:output>
<soap:body parts="GetStateParticipantsListResponse" use="literal"/>
<soap:header message="tns:GetStateParticipantsListResponseMessage" part="MeFHeader" use="literal"/>
</wsdl:output>
<wsdl:fault name="fault">
<soap:fault name="fault" use="literal"/>
</wsdl:fault>
</wsdl:operation>
</wsdl:binding>

```

1.8.1.5.1 element GetStateParticipantsListRequest

Table 1.8.8-20 Element GetStateParticipantsListRequest Details

diagram	<p>The diagram shows a rectangular box labeled 'GetStateParticipantsListRequest' connected by a line to a hexagonal box with a dashed border labeled 'mef:GetStateParticipantsListRequestType'.</p>
namespace	http://www.irs.gov/a2a/mef/MeFMSIServices.xsd
type	mef:GetStateParticipantsListRequestType
properties	content complex
source	<xs:element name="GetStateParticipantsListRequest" type="mef:GetStateParticipantsListRequestType"/>

1.8.1.5.2 element GetStateParticipantsListResponse



Table 1.8.8-21 Element GetStateParticipantsListResponse Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFMSIServices.xsd
type	mef:GetStateParticipantsListResponseType
properties	content complex
children	mef:Cnt mef:StateParticipantGrp
source	<code><xs:element name="GetStateParticipantsListResponse" type="mef:GetStateParticipantsListResponseType"/></code>

GetStateParticipantsListResponseType/StateParticipantGrp

Table 1.8.8-22 Element StateParticipantGrp Details

diagram	
type	mef:StateParticipantGrpType
properties	isRef 0 content complex
children	mef:StateCd mef:ParticipantProgramsTxt state name and Doing Business As indicator
source	<code><xs:element name="StateParticipantGrp" type="mef:StateParticipantGrpType" minOccurs="0" maxOccurs="unbounded"/></code>

1.8.2 Third Party Transmitter Web Services

The Web Services pertinent to the Third Party transmitters are described in the following sections. The Web Service descriptions and SOAP message contents for the MIME (J2EE), and MTOM (.NET) clients are identical except for the attachment transport mechanism. Therefore, a single set of service and message descriptions is applicable to MIME, and MTOM clients. Beginning with MeF Release 6.2 States will be allowed to access the Get Submissions Status family of Web services (GetSubmissionsStatus and GetSubmissionStatus). To accomplish this the States will need to use the appropriate WSDL and XML schema files previously reserved for use by Third Party transmitters. These details for the States are outlined in Section 0.



Table 1.8.8-23 Third Party Transmitter Web Services

services	bindings	porttypes	messages
GetAck	GetAck	GetAck	GetAckRequest
GetAcks	GetAcks	GetAcks	GetAckResponse
GetAcksByMsgID	GetAcksByMsgID	GetAcksByMsgID	GetAcksByMsgIDRequest
GetNewAcks	GetNewAcks	GetNewAcks	GetAcksByMsgIDResponse
GetNewSubmissionsStatus	GetNewSubmissionsStatus	GetNewSubmissionsStatus	GetAcksRequest
GetSubmissionsStatus	GetSubmissionsStatus	GetSubmissionsStatus	GetAcksResponse
GetSubmissionStatus	GetSubmissionStatus	GetSubmissionStatus	GetNewAcksRequest
SendSubmissions	SendSubmissions	SendSubmissions	GetNewAcksResponse
Get2290Schedule1	Get2290Schedule1	Get2290Schedule1	GetNewSubmissionsStatus Request
Get2290Schedule1s	Get2290Schedule1s	Get2290Schedule1s	GetNewSubmissionsStatus Response
Get2290Schedule1sByMsgID	Get2290Schedule1sByMsgID	Get2290Schedule1sByMsgID	GetSubmissionsStatus Request
GetNew2290Schedule1s	GetNew2290Schedule1s	GetNew2290Schedule1s	GetSubmissionsStatus Response
			GetSubmissionStatus Request
			GetSubmissionStatusResponse
			SendSubmissionsRequest
			SendSubmissionsResponse
			Get2290Schedule1Request
			Get2290Schedule1Response
			Get2290Schedule1sByMsgID Request
			Get2290Schedule1sByMsgID Response
			Get2290Schedule1sRequest
			Get2290Schedule1sResponse
			GetNew2290Schedule1s Request
			GetNew2290Schedule1s Response



1.8.2.1 Get Ack

Third Party Transmitters will have authorization to programmatically request a Web Service to retrieve an acknowledgement for a specified submission ID. The requesting system will provide a GetAckRequest message in the SOAP body containing the submission ID. If successful, the service returns an empty GetAckResponse message tag in the SOAP body and a single zip file attachment containing the acknowledgement. If the submission ID is not found or any other error occurs, a SOAP fault message will be returned to the requesting system.

1.8.2.1.1 element GetAckRequest

Table 1.8.8-24 Element GetAckRequest Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFTtransmitterService.xsd
type	GetAckRequestType
properties	content complex
children	SubmissionId
source	<code><xsd:element name="GetAckRequest" type="GetAckRequestType"/></code>

1.8.2.1.2 element GetAckResponse

Table 1.8.8-25 Element GetAckResponse Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFTtransmitterService.xsd
type	GetAckResponseType
properties	content complex
source	<code><xsd:element name="GetAckResponse" type="GetAckResponseType"/></code>

1.8.2.2 Get Acks

Third Party Transmitters will have authorization to programmatically request a Web Service to retrieve acknowledgements for a specified list of submission IDs. The requesting system will provide a GetAcksRequest message in the SOAP body containing a list of submission IDs. If successful, the service



returns a GetAcksResponse message in the SOAP body containing a list of submission errors for any submission IDs from the submission ID list that were not found, and a single zip file attachment containing the acknowledgements. Otherwise, a SOAP error message will be returned to the requesting system.

1.8.2.2.1 element GetAcksRequest

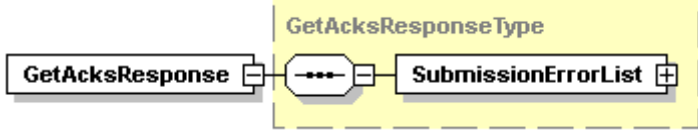
Table 1.8.8-25 Element GetAcksRequest Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd
type	GetAcksRequestType
properties	content complex
children	SubmissionIdList
source	<pre> <xsd:element name="GetAcksRequest" type="GetAcksRequestType"/> <xsd:complexType name="GetAcksRequestType"> <xsd:sequence> <xsd:element name="SubmissionIdList" type="SubmissionIdListType"/> </xsd:sequence> </xsd:complexType> <xsd:complexType name="SubmissionIdListType"> <xsd:sequence> <xsd:element name="SubmissionId" type="SubmissionIdType" maxOccurs="unbounded"/> </xsd:sequence> </xsd:complexType> <xsd:simpleType name="SubmissionIdType"> <xsd:restriction base="xsd:string"> <xsd:pattern value="[0-9]{13}[a-z0-9]{7}"/> </xsd:restriction> </xsd:simpleType> </pre>



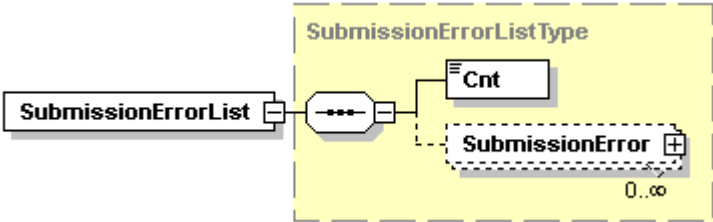
1.8.2.2.2 element GetAcksResponse

Table 1.8.8-27 Element GetAcksResponse Details

diagram	 <p>The diagram shows a box labeled 'GetAcksResponseType' containing a sequence of two elements: 'GetAcksResponse' followed by 'SubmissionErrorList'. A dashed line with three dots connects the two elements, indicating a sequence. The 'SubmissionErrorList' element has a plus sign in a box to its right, indicating it is a complex type.</p>
namespace	http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd
type	GetAcksResponseType
properties	content complex
children	ErrorList
source	<xsd:element name="GetAcksResponse" type="GetAcksResponseType"/>

GetAcksResponseType/SubmissionErrorList

Table 1.8.8-28 Element SubmissionErrorList Details

diagram	 <p>The diagram shows a box labeled 'SubmissionErrorListType' containing a sequence of two elements: 'SubmissionErrorList' followed by 'Cnt'. A dashed line with three dots connects the two elements, indicating a sequence. The 'Cnt' element has a plus sign in a box to its right, indicating it is a complex type. Below the 'Cnt' element, there is a dashed box labeled 'SubmissionError' with a plus sign in a box to its right, and the cardinality '0..∞' is shown below it, indicating that the 'Cnt' element contains a list of 'SubmissionError' elements.</p>
namespace	http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd
type	SubmissionErrorListType
properties	isRef 0 content complex
children	Cnt SubmissionError
source	<xsd:element name="SubmissionErrorList" type="SubmissionErrorListType"/>



GetAcksResponseType/SubmissionErrorListType/SubmissionError

Table 1.8.8-29 Element SubmissionError Details

diagram	<p>The diagram shows the XSD structure for SubmissionErrorType. It is a complex type containing a sequence of elements: SubmissionId and ErrorClassificationDetail. ErrorClassificationDetail is a complex type containing a sequence of ErrorMessageTxt, ErrorClassificationCd, and ErrorMessageCd.</p>
namespace	http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd
children	SubmissionId Error
used by	element SubmissionErrorListType/SubmissionError
source	<pre> <xsd:complexType name="SubmissionErrorType"> <xsd:sequence> <xsd:element name="SubmissionId" type="SubmissionIdType"/> <xsd:element name="ErrorClassificationDetail" type="ErrorClassificationDetailType"/> </xsd:sequence> </xsd:complexType> </pre>

1.8.2.3 Get New Acks

Third-Party Transmitters will have authorization to programmatically request a Web Service to retrieve new acknowledgements for a specified category and/or agency. The requesting system will provide a GetNewAcksRequest message in the SOAP body containing the maximum number of results that should be returned (up to 500) and optionally the category and/or agency and the MeFHeader message in the SOAP header containing the ETIN. The requesting system can supply just the MaxResultCnt, the MaxResultCnt and ExtndAcknowledgementCategoryCd, the MaxResultCnt and GovernmentAgencyTypeCd, or the MaxResultCnt and ExtndAcknowledgementCategoryCd and GovernmentAgencyTypeCd. If successful, the service returns a GetNewAcksResponse message in the SOAP body containing a Boolean flag indicating if more acknowledgements with the specified criteria are available and a single zip file attachment containing the acknowledgements. Otherwise, a SOAP fault error message will be returned to the requesting system. The valid submission ExtndAcknowledgementCategoryCd values are:

- CORP (Corporations)
- EO (Tax Exempt Organizations)
- PART (Partnerships)
- EXCISE (Excise Tax Form Submissions)
- IND (Individuals)
- UNKN (Unknown)



- ESTRST (Estate & Trust)
- EMPL (Employee)
- all (CORP and/or PART and/or EXCISE and/or EO and/or IND and/or UNKN and/or ESTRST and/or EMPL)

NOTE: Any particular ETIN requesting “all” will retrieve acknowledgements for any combination of CORP, PART, EXCISE, EO, IND, UNKN, ESTRST and EMPL depending on the access allowed for that ETIN in the IRS database and the acknowledgements available at the time.

The valid agency types are Federal, State, or both.

1.8.2.3.1 element GetNewAcksRequest

Table 1.8.8-30 Element GetNewAcksRequest Details

diagram	<p>The diagram illustrates the structure of the GetNewAcksRequestType element. It is a sequence of three elements: MaxResultCnt, ExtndAcknowledgementCatego..., and GovernmentAgencyTypeCd. The GetNewAcksRequest element is shown as a container for this sequence.</p>
namespace	http://www.irs.gov/a2a/mef/MeFTtransmitterService.xsd
children	MaxResultCnt ExtndAcknowledgementCategoryCd GovernmentAgencyTypeCd
used by	element GetNewAcksRequest
Source	<pre> <xsd:complexType name="GetNewAcksRequestType"> <xsd:sequence> <xsd:element name="MaxResultCnt" type="xsd:positiveInteger"/> <xsd:element name="ExtndAcknowledgementCategoryCd" type="ExtndAcknowledgementCategoryCdType" minOccurs="0"/> <xsd:element name="GovernmentAgencyTypeCd" type="GovernmentAgencyTypeCdType" minOccurs="0"/> </xsd:sequence> </xsd:complexType> </pre>

GetNewAcksRequestType/MaxResultCnt

Table 1.8.8-31 Element MaxResultCnt Details


diagram	<p>The diagram shows the MaxResultCnt element as a simple integer value.</p>
---------	---



namespace	http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd
type	xsd:positiveInteger
properties	isRef 0 content simple
source	<xsd:element name="MaxResultCnt" type="xsd:positiveInteger"/>


GetNewAcksRequestType/ExtndAcknowledgementCategoryCd

Table 1.8.8-32 Element ExtndAcknowledgementCategoryCd Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd
type	ExtndAcknowledgementCategoryCdType
properties	isRef 0 content simple
facets	enumeration CORP enumeration EO enumeration PART enumeration EXCISE enumeration IND enumeration UNKN enumeration ESTRST enumeration EMPL enumeration all
source	<xsd:element name="Category" type="CategoryType" minOccurs="0"/>

GetNewAcksRequestType/GovernmentAgencyTypeCd

Table 1.8.8-33 Element GovernmentAgencyTypeCd Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd
type	GovernmentAgencyTypeCdType
properties	isRef 0



	content simple
facets	enumeration Federal enumeration State enumeration both
source	<code><xsd:element name="Agency" type="AgencyType" minOccurs="0"/></code>

1.8.2.3.2 element GetNewAcksResponse

Table 1.8.8-34 Element GetNewAcksResponse Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd
type	GetNewAcksResponseType
properties	content complex
children	MoreAvailableInd
source	<code><xsd:element name="GetNewAcksResponse" type="GetNewAcksResponseType"/></code>

GetNewAcksResponseType/MoreAvailableInd

Table 1.8.8-35 Element MoreAvailableInd Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd
type	xsd:boolean
properties	isRef 0 content simple
source	<code><xsd:element name="MoreAvailableInd" type="xsd:boolean"/></code>

1.8.2.4 Get Acks by Message ID

Third-Party Transmitters will have authorization to programmatically request a Web Service to retrieve acknowledgements previously retrieved by a Get New Acks request. The requesting system will provide a



GetAcksByMsgIDRequest message in the SOAP body containing the MessageID used in the previous Get New Acks request. If successful, the service returns a GetAcksByMsgIDResponse message in the SOAP body containing a Boolean flag indicating if more acknowledgements were available at the time the previous Get New Acknowledgements request was processed and a single zip file attachment containing the acknowledgements. Otherwise, a SOAP error message will be returned to the requesting system.

1.8.2.4.1 element GetAcksByMsgIDRequest

Table 1.8.8-36 Element GetAcksByMsgIDRequest Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFTtransmitterService.xsd
type	GetAcksByMsgIDRequestType
properties	content complex
children	MessageID
source	<pre> <xsd:element name="GetAcksByMsgIDRequest" type="GetAcksByMsgIDRequestType"/> <xsd:complexType name="GetAcksByMsgIDRequestType"> <xsd:sequence> <xsd:element name="MessageID" type="MessageIDType"/> </xsd:sequence> </xsd:complexType> </pre>

1.8.2.4.2 element GetAcksByMsgIDResponse

Table 1.8.8-37 Element GetAcksByMsgIDResponse Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFTtransmitterService.xsd
type	GetAcksByMsgIDResponseType
properties	content complex
children	MoreAvailableInd



source	<pre> <xsd:element name="GetAcksByMsgIDResponse" type="GetAcksByMsgIDResponseType"/> <xsd:complexType name="GetAcksByMsgIDResponseType"> <xsd:sequence> <xsd:element name="MoreAvailableInd" type="xsd:boolean"/> </xsd:sequence> </xsd:complexType> </pre>
--------	---

1.8.2.5 Get New Submissions Status

Third Party Transmitters will have authorization to programmatically request a Web Service to retrieve new submissions statuses for a particular ETIN. The requesting system will provide a `GetNewSubmissionsStatusRequest` message in the SOAP body containing the maximum number of results that should be returned (up to 100) and the `MeFHeader` message in the SOAP header containing the ETIN. If successful, the service returns a `GetNewSubmissionsStatusResponse` message in the SOAP body containing a Boolean flag indicating if more submissions statuses are available, and a single zip file attachment containing the submission statuses. Otherwise, a SOAP error message will be returned to the requesting system.

1.8.2.5.1 element `GetNewSubmissionsStatusRequest`

Table 1.8.8-38 Element `GetNewSubmissionsStatusRequest` Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd
type	GetNewSubmissionsStatusRequestType
properties	content complex
children	MaxResultsCnt
source	<pre><xsd:element name="GetNewSubmissionsStatusRequest" type="GetNewSubmissionsStatusRequestType"/></pre>



1.8.2.5.2 element GetNewSubmissionsStatusResponse

Table 1.8.8-39 element GetNewSubmissionsStatusResponse Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFTTransmitterService.xsd
type	GetNewSubmissionsStatusResponseType
properties	content complex
children	MoreAvailableInd
source	<pre><xsd:element name="GetNewSubmissionsStatusResponse" type="GetNewSubmissionsStatusResponseType"/> <xsd:element name="MoreAvailableInd" type="xsd:boolean"/></pre>

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFTTransmitterService.xsd
type	ErrorClassificationDetailType
properties	isRef 0 content complex
children	ErrorMessageTxt ErrorClassificationCd ErrorMessageCd
source	<pre><xsd:element name="ErrorClassificationDetail" type="ErrorClassificationDetailType"/></pre>

1.8.2.6 Get Submissions Status

Both States and Third Party Transmitters will have authorization to programmatically request a Web Service to retrieve submissions statuses for a specific set of submissions. The requesting system will provide a GetSubmissionsStatusRequest message in the SOAP body containing a list of submission IDs. If successful, the service returns a GetSubmissionsStatusResponse message in the SOAP body containing a list of submission errors for any submission IDs from the submission ID list that were not found and a single zip file attachment containing the submission statuses. Otherwise, a SOAP error message will be returned to the requesting system.



1.8.2.6.1 element GetSubmissionsStatusRequest

Table 1.8.8-40 Element GetSubmissionsStatusRequest Details

diagram	<p>The diagram illustrates the structure of the <code>GetSubmissionsStatusRequestType</code>. It shows a sequence of elements: <code>GetSubmissionsStatusRequest</code>, followed by <code>SubmissionIdList</code>. The <code>SubmissionIdList</code> element is contained within a dashed box labeled <code>SubmissionIdListType</code>. Inside this box, there is a sequence of <code>SubmissionId</code> elements, with a cardinality of <code>1..∞</code> indicated below the <code>SubmissionId</code> element.</p>
namespace	http://www.irs.gov/a2a/mef/MeFTTransmitterService.xsd
type	GetSubmissionsStatusRequestType
properties	content complex
children	SubmissionIdList
source	<pre> <xsd:element name="GetSubmissionsStatusRequest" type="GetSubmissionsStatusRequestType"/> <xsd:complexType name="GetSubmissionsStatusRequestType"> <xsd:sequence> <xsd:element name="SubmissionIdList" type="SubmissionIdListType"/> </xsd:sequence> </xsd:complexType> <xsd:complexType name="SubmissionIdListType"> <xsd:sequence> <xsd:element name="SubmissionId" type="SubmissionIdType" maxOccurs="unbounded"/> </xsd:sequence> </xsd:complexType> <xsd:simpleType name="SubmissionIdType"> <xsd:restriction base="xsd:string"> <xsd:pattern value="[0-9]{13}[a-z0-9]{7}"/> </xsd:restriction> </xsd:simpleType> </pre>



1.8.2.6.2 element GetSubmissionsStatusResponse

Table 1.8.8-41 Element GetSubmissionsStatusResponse Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFTtransmitterService.xsd
type	GetSubmissionsStatusResponseType
properties	content complex
children	SubmissionErrorList
source	<code><xsd:element name="GetSubmissionsStatusResponse" type="GetSubmissionsStatusResponseType"/></code>

GetSubmissionsStatusResponseType/SubmissionErrorList

Table 1.8.8-42 Element SubmissionErrorList Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFTtransmitterService.xsd
type	ListOfSubmissionErrorType
properties	isRef 0 content complex
children	Cnt SubmissionError
source	<code><xsd:element name="ErrorList" type="ListOfSubmissionErrorType"/></code>



ListOfSubmissionErrorType/SubmissionError

Table 1.8.8-43 Element SubmissionError Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFTtransmitterService.xsd
type	SubmissionErrorType
properties	isRef 0 content complex
children	SubmissionId ErrorClassificationDetail
source	<code><xsd:element name="SubmissionError" type="SubmissionErrorType" maxOccurs="unbounded"/></code>

SubmissionErrorType/ErrorClassificationDetail

Table 1.8.8-44 Element ErrorClassificationDetail Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFTtransmitterService.xsd
type	ErrorClassificationDetailType
properties	isRef 0 content complex
children	ErrorMessageTxt ErrorClassificationCd ErrorMessageCd
source	<code><xsd:element name="ErrorClassificationDetail" type="ErrorClassificationDetailType"/></code>



1.8.2.7 Get Submission Status

Both States and Third Party Transmitters will have authorization to programmatically request a Web Service to retrieve a submission status for a specific submission. The requesting system will provide a GetSubmissionStatusRequest message in the SOAP body containing a submission ID. If successful, the service returns an empty GetSubmissionStatusResponse tag in the SOAP body and a single zip file attachment containing the submission status. Otherwise, a SOAP error message will be returned to the requesting system.

1.8.2.7.1 element GetSubmissionStatusRequest

Table 1.8.8-45 Element GetSubmissionStatusRequest Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd
type	GetSubmissionStatusRequestType
properties	content complex
children	SubmissionId
source	<code><xsd:element name="GetSubmissionStatusRequest" type="GetSubmissionStatusRequestType"/></code>

1.8.2.7.2 element GetSubmissionStatusResponse

Table 1.8.8-46 Element GetSubmissionStatusResponse Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd
type	GetSubmissionStatusResponseType
properties	content complex
source	<code><xsd:element name="GetSubmissionStatusResponse" type="GetSubmissionStatusResponseType"/></code>



1.8.2.8 Send Submissions

Third Party Transmitters will have authorization to programmatically request a Web Service to send submissions for a specific set of returns. The requesting system will provide a SendSubmissionsRequest message in the SOAP body containing a list of submission IDs and a single zip file attachment containing the submissions. The requesting system must also already be logged in, have a valid session, and provide the session key in the wsse:Security portion of the SOAP header. If successful, the service returns an empty SendSubmissionsResponse tag in the SOAP body, and a single zip file attachment containing a list of submission receipts. Otherwise, a SOAP error message will be returned to the requesting system.

1.8.2.8.1 element SendSubmissionsRequest

Table 1.8.8-47 Element SendSubmissionsRequest Details

diagram	<p>The diagram shows a class named SendSubmissionsRequestType (highlighted in a yellow dashed box) which contains two elements: SendSubmissionsRequest and SubmissionDataList. The SubmissionDataList element is shown with a plus sign in its top-right corner, indicating it is a complex type containing multiple instances of SubmissionData.</p>
namespace	http://www.irs.gov/a2a/mef/MeFTtransmitterService.xsd
type	SendSubmissionsRequestType
properties	content complex
children	SubmissionDataList
source	<pre> <xsd:element name="SendSubmissionsRequest" type="SendSubmissionsRequestType"/> <xsd:complexType name="SendSubmissionsRequestType"> <xsd:sequence> <xsd:element name="SubmissionDataList" type="SubmissionDataListType"/> </xsd:sequence> </xsd:complexType> <xsd:complexType name="SubmissionDataListType"> <xsd:sequence> <xsd:element name="SubmissionData" type="SubmissionDataType" maxOccurs="unbounded"/> </xsd:sequence> </xsd:complexType> </pre>



SendSubmissionsRequestType/SubmissionDataList

Table 1.8.8-48 Element SubmissionDataList Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd
type	SubmissionDataListType
properties	isRef 0 content complex
children	SubmissionData
source	<pre> <xsd:element name="SubmissionDataList" type="SubmissionDataListType"/> <xsd:complexType name="SubmissionDataListType"> <xsd:sequence> <xsd:element name="SubmissionData" type="SubmissionDataType" maxOccurs="unbounded"/> </xsd:sequence> </xsd:complexType> </pre>

SubmissionDataListType/SubmissionData

Table 1.8.8-49 Element SubmissionData Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd
type	SubmissionDataType



properties	isRef 0 content complex
children	SubmissionId ElectronicPostmark
source	<pre> <xsd:element name="SubmissionData" type="SubmissionDataType" maxOccurs="unbounded"/> <xsd:complexType name="SubmissionDataType"> <xsd:sequence> <xsd:element name="SubmissionId" type="SubmissionIdType"/> <xsd:element name="ElectronicPostmarkTs" type="xsd:dateTime" minOccurs="0"/> </xsd:sequence> </xsd:complexType> </pre>

1.8.2.8.2 element SendSubmissionsResponse

Table 1.8.8-50 Element SendSubmissionsResponse Details

Diagram	
Namespace	http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd
Type	SendSubmissionsResponseType
Properties	content complex
Source	<pre> <xsd:element name="SendSubmissionsResponse" type="SendSubmissionsResponseType"/> <xsd:complexType name="SendSubmissionsResponseType"> <xsd:sequence> <xsd:element name="DepositId" type="DepositIdType"/> </xsd:sequence> </xsd:complexType> <xsd:simpleType name="DepositIdType"> <xsd:restriction base="xsd:string"> <xsd:pattern value="[A-Za-z0-9]*/> <xsd:maxLength value="80"/> </xsd:restriction> </xsd:simpleType> </pre>



1.8.2.9 Get 2290Schedule1

Third Party Transmitters will have authorization to programmatically request a Web Service to retrieve a form 2290 Schedule1 for a specified submission ID. The requesting system will provide a Get2290Schedule1Request message in the SOAP body containing the submission ID. If successful, the service returns an empty Get2290Schedule1Response message tag in the SOAP body and a single zip file attachment containing the stamped 2290 Schedule1 form. If the submission ID is not found or any other error occurs, a SOAP fault message will be returned to the requesting system.

1.8.2.9.1 element Get2290Schedule1Request

Table 1.8.8-51 Element Get2290Schedule1Request Details

diagram	
namespace	http://www.irs.gov/a2a/mef/ETECTransmitterService.xsd
type	Get2290Schedule1RequestType
properties	content complex
children	SubmissionId
source	<pre> <xsd:element name="Get2290Schedule1Request" type="Get2290Schedule1RequestType"/> <xsd:complexType name="Get2290Schedule1RequestType"> <xsd:sequence> <xsd:element name="SubmissionId" type="SubmissionIdType"/> </xsd:sequence> </xsd:complexType> <xsd:simpleType name="SubmissionIdType"> <xsd:restriction base="xsd:string"> <xsd:pattern value="[0-9]{13}[a-z0-9]{7}"/> </xsd:restriction> </xsd:simpleType> </pre>



1.8.2.9.2 element Get2290Schedule1Response

Table 1.8.8-52 Element Get2290Schedule1Response Details

diagram	<p>The diagram shows a rectangular box labeled 'Get2290Schedule1Response' connected by a line to a hexagonal box with four dots inside. This hexagonal box is enclosed within a larger dashed-line box labeled 'Get2290Schedule1ResponseType'.</p>
namespace	http://www.irs.gov/a2a/mef/ETECTransmitterService.xsd
type	Get2290Schedule1ResponseType
properties	content complex
source	<code><xsd:element name="Get2290Schedule1Response" type="Get2290Schedule1ResponseType"/></code>

1.8.2.10 Get 2290Schedule1s

Third Party Transmitters will have authorization to programmatically request a Web Service to retrieve form 2290 Schedule1s for a specified list of submission IDs. The requesting system will provide a Get2290Schedule1sRequest message in the SOAP body containing a list of submission IDs. If successful, the service returns a Get2290Schedule1sResponse message in the SOAP body containing the count of the number of Schedule 1s included in the response, a list of submission errors for any submission IDs from the submission ID list that were not found and a single zip file attachment containing the stamped 2290 Schedule1 forms. Otherwise, a SOAP error message will be returned to the requesting system.

1.8.2.10.1 element Get2290Schedule1sRequest

Table 1.8.8-53 Element Get2290Schedule1sRequest Details

diagram	<p>The diagram shows a rectangular box labeled 'Get2290Schedule1sRequest' connected by a line to a hexagonal box with four dots inside. This hexagonal box is connected to a rectangular box labeled 'SubmissionIdList'. 'SubmissionIdList' is connected to another hexagonal box with four dots inside, which is connected to a rectangular box labeled 'SubmissionId'. The 'SubmissionId' box has a '1..∞' cardinality indicator below it. The entire structure from the hexagonal box to 'SubmissionId' is enclosed within a dashed-line box labeled 'Get2290Schedule1sRequestType'.</p>
namespace	http://www.irs.gov/a2a/mef/ETECTransmitterService.xsd
type	Get2290Schedule1sRequestType
properties	content complex
children	SubmissionIdList
source	<code><xsd:element name="Get2290Schedule1sRequest" type="Get2290Schedule1sRequestType"/></code>



```

<xsd:complexType name="Get2290Schedule1sRequestType">
  <xsd:sequence>
    <xsd:element name="SubmissionIdList" type="SubmissionIdListType"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SubmissionIdListType">
  <xsd:sequence>
    <xsd:element name="SubmissionId" type="SubmissionIdType" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:simpleType name="SubmissionIdType">
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="[0-9]{13}[a-z0-9]{7}"/>
  </xsd:restriction>
</xsd:simpleType>

```

1.8.2.10.2 element Get2290Schedule1sResponse

Table 1.8.8-54 Element Get2290Schedule1sResponse Details

diagram	
namespace	http://www.irs.gov/a2a/mef/ETECTransmitterService.xsd
type	Get2290Schedule1sResponseType
properties	content complex
children	Cnt ETECSubmissionErrorList
source	<code><xsd:element name="Get2290Schedule1sResponse" type="Get2290Schedule1sResponseType"/></code>



```

<xsd:complexType name="Get2290Schedule1sResponseType">
  <xsd:sequence>
    <xsd:element name="Cnt" type="xsd:nonNegativeInteger"/>
    <xsd:element name="ETECSubmissionErrorList" type="ETECSubmissionErrorListType"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ETECSubmissionErrorListType">
  <xsd:sequence>
    <xsd:element name="SubmissionError" type="SubmissionErrorType" minOccurs="0"
maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SubmissionErrorType">
  <xsd:sequence>
    <xsd:element name="SubmissionId" type="SubmissionIdType"/>
    <xsd:element name="ErrorClassificationDetail" type="ErrorClassificationDetailType"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:simpleType name="SubmissionIdType">
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="[0-9]{13}[a-z0-9]{7}"/>
  </xsd:restriction>
</xsd:simpleType>
<xsd:complexType name="ErrorClassificationDetailType">
  <xsd:sequence>
    <xsd:element name="ErrorMessageTxt" type="xsd:string"/>
    <xsd:element name="ErrorClassificationCd" type="ErrorClassificationCdType"/>
    <xsd:element name="ErrorMessageCd" type="xsd:string" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:simpleType name="ErrorClassificationCdType">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="SYSTEM_ERROR"/>
    <xsd:enumeration value="RESOURCE_UNAVAILABLE"/>
    <xsd:enumeration value="MESSAGE_FORMAT_ERROR"/>
    <xsd:enumeration value="SECURITY_ERROR"/>
  </xsd:restriction>
</xsd:simpleType>

```



	<pre><xsd:enumeration value="REQUEST_ERROR"/> </xsd:restriction></pre>
--	--

1.8.2.11 Get New 2290Schedule1s

Third-Party Transmitters will have authorization to programmatically request a Web Service to retrieve new Form 2290 Schedule1s for a specified ETIN. The requesting system will provide a GetNew2290Schedule1sRequest message in the SOAP body containing the maximum number of results that should be returned (up to 100) and the MeFHeader message in the SOAP header containing the ETIN. If successful, the service returns a GetNew2290Schedule1sResponse message in the SOAP body containing the count of the number of Schedule 1s included in the response, a Boolean flag indicating if more Schedule 1s are available, and a single zip file attachment containing the stamped 2290 Schedule1 forms. Otherwise, a SOAP error message will be returned to the requesting system.

1.8.2.11.1 element GetNew2290Schedule1sRequest

Table 1.8.8-55 Element GetNew2290Schedule1sRequest Details

diagram	
namespace	http://www.irs.gov/a2a/mef/ETECTransmitterService.xsd
type	GetNew2290Schedule1sRequestType
properties	content complex
children	MaxResultCnt
source	<code><xsd:element name="GetNew2290Schedule1sRequest" type="GetNew2290Schedule1sRequestType"/></code>

1.8.2.11.2 element GetNew2290Schedule1sResponse

Table 1.8.8-56 Element GetNew2290Schedule1sResponse Details

diagram	
namespace	http://www.irs.gov/a2a/mef/ETECTransmitterService.xsd



type	GetNew2290Schedule1sResponseType
properties	content complex
children	Cnt MoreAvailableInd
source	<pre> <xsd:element name="GetNew2290Schedule1sResponse" type="GetNew2290Schedule1sResponseType"/> <xsd:complexType name="GetNew2290Schedule1sResponseType"> <xsd:sequence> <xsd:element name="Cnt" type="xsd:nonNegativeInteger"/> <xsd:element name="MoreAvailableInd" type="xsd:boolean"/> </xsd:sequence> </xsd:complexType> </pre>

1.8.2.12 Get 2290Schedule1s by Message ID

Third-Party Transmitters will have authorization to programmatically request a Web Service to retrieve form 2290 Schedule1s previously retrieved by a Get New 2290Schedule1s request. The requesting system will provide a Get2290Schedule1sByMsgIDRequest message in the SOAP body containing the MessageID used in the previous Get New 2290Schedule1s request. If successful, the service returns a Get2290Schedule1sByMsgIDResponse message in the SOAP body containing the count of the number of Schedule 1s included in the response, a Boolean flag indicating if more 2290Schedule1s were available at the time the previous Get New 2290Schedule1s request was processed and a single zip file attachment containing the stamped 2290 Schedule1 forms. Otherwise, a SOAP error message will be returned to the requesting system.

1.8.2.12.1 element GetNew2290Schedule1sResponse

Table 1.8.8-57 Element Get2290Schedule1sByMsgIDRequest Details

diagram	
namespace	http://www.irs.gov/a2a/mef/ETECTransmitterService.xsd
type	Get2290Schedule1sByMsgIDRequestType
properties	content complex
children	MessageId
source	<pre> <xsd:element name="Get2290Schedule1sByMsgIDRequest" type="Get2290Schedule1sByMsgIDRequestType"/> </pre>



	<pre> <xsd:complexType name="Get2290Schedule1sByMsgIDRequestType"> <xsd:sequence> <xsd:element name="MsgId" type="MsgIdType"/> </xsd:sequence> </xsd:complexType> </pre>
--	--

1.8.2.12.2 element Get2290Schedule1sByMsgIDResponse

Table 1.8.8-58 Element Get2290Schedule1sByMsgIDResponse Details

diagram	
namespace	http://www.irs.gov/a2a/mef/ETECTransmitterService.xsd
type	Get2290Schedule1sByMsgIDResponseType
properties	content complex
children	Cnt MoreAvailableInd
source	<pre> <xsd:element name="Get2290Schedule1sByMsgIDResponse" type="Get2290Schedule1sByMsgIDResponseType"/> <xsd:complexType name="Get2290Schedule1sByMsgIDResponseType"> <xsd:sequence> <xsd:element name="Cnt" type="xsd:nonNegativeInteger"/> <xsd:element name="MoreAvailableInd" type="xsd:boolean"/> </xsd:sequence> </xsd:complexType> </pre>



1.8.3 State Web Services

The Web Services pertinent to the states are described in the following sections. The Web Service descriptions and SOAP message contents for the MIME (J2EE), and MTOM (.NET) clients are identical except for the attachment transport mechanism. Therefore, a single set of service and message descriptions is applicable to MIME, and MTOM clients. Beginning with MeF Release 6.2 States will be allowed to access the Get Submissions Status family of Web services (GetSubmissionsStatus and GetSubmissionStatus). The details for these web services are outlined in Sections 1.8.2.6 and 1.8.2.7. To accomplish this the States will need to use the appropriate MeF Transmitter Service WSDL and XML schema files previously reserved for use by Third Party transmitters (see Table 2-1).

Table 1.8.8-59 State Web Services

services	bindings	porttypes	messages
GetAckNotification	GetAckNotification	GetAckNotification	GetAckNotificationRequest
GetAckNotifications	GetAckNotifications	GetAckNotifications	GetAckNotificationResponse
GetNewAckNotifications	GetNewAckNotifications	GetNewAckNotifications	GetAckNotificationsRequest
GetNewSubmissions	GetNewSubmissions	GetNewSubmissions	GetAckNotificationsResponse
GetSubmission	GetSubmission	GetSubmission	GetNewAckNotificationsRequest
GetSubmissions	GetSubmissions	GetSubmissions	GetNewAckNotificationsResponse
GetSubmissionsByMsgID	GetSubmissionsByMsgID	GetSubmissionsByMsgID	GetNewSubmissionsRequest
SendAcks	SendAcks	SendAcks	GetNewSubmissionsResponse
SendSubmissionReceipts	SendSubmissionReceipts	SendSubmissionReceipts	GetSubmissionRequest
GetSubmissionReconciliationList	GetSubmissionReconciliationList	GetSubmissionReconciliationList	GetSubmissionResponse
			GetSubmissionsByMsgIDRequest
			GetSubmissionsByMsgIDResponse
			GetSubmissionsRequest
			GetSubmissionsResponse
			SendAcksRequest
			SendAcksResponse
			SendSubmissionReceiptsRequest
			SendSubmissionReceiptsResponse
			GetSubmissionReconciliationList Request
			GetSubmissionReconciliationList Response

Table 1.8.8-60 Transmitter Services Accessible by States

services	bindings	porttypes	messages
----------	----------	-----------	----------



GetSubmissionsStatus	GetSubmissionsStatus	GetSubmissionsStatus	GetSubmissionsStatusRequest
GetSubmissionStatus	GetSubmissionStatus	GetSubmissionStatus	GetSubmissionsStatusResponse
			GetSubmissionStatusRequest
			GetSubmissionStatusResponse

1.8.3.1 Get Ack Notification

States will have authorization to programmatically request a Web Service to retrieve information as to whether Third-Party transmitters have retrieved the acknowledgements submitted by the states for a specified state return. The requesting system will provide a `GetAckNotificationRequestArgs` message in the SOAP body containing the submission ID. If successful, the service returns an empty `GetAckNotificationResponse` message tag in the SOAP body and a single zip file attachment containing the acknowledgement retrieval notification. Otherwise, a SOAP error message will be returned to the requesting system.

1.8.3.1.1 element `GetAckNotificationRequestArgs`

Table 1.8.8-61 Element `GetAckNotificationRequestArgs` Details

Diagram	
Namespace	<code>http://www.irs.gov/a2a/mef/MeFStateService.xsd</code>
Type	GetAckNotificationRequestType
Properties	content complex
Children	SubmissionId
Source	<pre> <xsd:element name="GetAckNotificationRequestArgs" type="msx:GetAckNotificationRequestType"/> <xsd:complexType name="GetAckNotificationRequestType"> <xsd:sequence> <xsd:element name="SubmissionId" type="msx:SubmissionIdType"/> </xsd:sequence> </xsd:complexType> </pre>



1.8.3.1.2 element GetAckNotificationResponse

Table 1.8.8-62 Element GetAckNotificationResponse Details

diagram	<p>The diagram shows a rectangular box labeled 'GetAckNotificationResponse' connected by a line to a hexagonal box containing four dots. This hexagonal box is enclosed within a larger, dashed yellow rectangular box labeled 'GetAckNotificationResponseType'.</p>
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	GetAckNotificationResponseType
properties	content complex
source	<xsd:element name="GetAckNotificationResponse" type="msx:GetAckNotificationResponseType"/>

1.8.3.2 Get Ack Notifications

States will have authorization to programmatically request a Web Service to retrieve information as to whether Third-Party transmitters have retrieved the acknowledgements submitted by the states for a specified set of state returns. The requesting system will provide a GetAckNotificationsRequestArgs message in the SOAP body containing a list of submission IDs. If successful, the service returns a GetAckNotificationsResponse message in the SOAP body containing a list of submission errors for any submission IDs from the submission ID list that were not found and a single zip file attachment containing the acknowledgement retrieval notification. Otherwise, a SOAP error message will be returned to the requesting system.

1.8.3.2.1 element GetAckNotificationsRequestArgs

Table 1.8.8-63 Element GetAckNotificationsRequestArgs Details

diagram	<p>The diagram shows a rectangular box labeled 'GetAckNotificationsRequestArgs' connected by a line to a hexagonal box containing four dots. This hexagonal box is connected to a rectangular box labeled 'SubmissionIdList'. The 'SubmissionIdList' box is connected to another hexagonal box containing four dots, which is then connected to a rectangular box labeled 'SubmissionId'. The 'SubmissionId' box has a '1..∞' cardinality indicator below it. The 'SubmissionIdList' box and the second hexagonal box are enclosed within a larger, dashed yellow rectangular box labeled 'GetAckNotificationsRequestType'.</p>
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	GetAckNotificationsRequestType
properties	content complex



children	SubmissionIdList
source	<pre> <xsd:element name="GetAckNotificationsRequestArgs" type="msx:GetAckNotificationsRequestType"/> <xsd:complexType name="GetAckNotificationsRequestType"> <xsd:sequence> <xsd:element name="SubmissionIdList" type="msx:SubmissionIdListType"/> </xsd:sequence> </xsd:complexType> </pre>

1.8.3.2.2 element GetAckNotificationsResponse

Table 1.8.8-64 Element GetAckNotificationsResponse Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	GetAckNotificationsResponseType
properties	content complex
children	ErrorList
source	<code><xsd:element name="GetAckNotificationsResponse" type="msx:GetAckNotificationsResponseType"/></code>

GetAckNotificationsResponseType/SubmissionErrorList

Table 1.8.8-65 Element SubmissionErrorList Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	SubmissionErrorListType
properties	isRef 0



	content complex
children	Cnt SubmissionError
source	<code><xsd:element name="SubmissionErrorList" type="msx:SubmissionErrorListType"/></code>

SubmissionErrorListType/SubmissionError

Table 1.8.8-66 Element SubmissionError Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	SubmissionErrorType
properties	isRef 0 content complex
children	SubmissionId Error
source	<code><xsd:element name="SubmissionError" type="SubmissionErrorType" minOccurs="0" maxOccurs="unbounded"/></code>

SubmissionErrorType/ErrorClassificationDetail

Table 1.8.8-67 Element ErrorClassificationDetail Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	MeFErrorType
properties	isRef 0 content complex
children	ErrorMessage ErrorClassification ErrorCode
source	<code><xsd:element name="ErrorClassificationDetail" type="ErrorClassificationDetailType"/></code>



1.8.3.3 Get New Ack Notifications

States will have authorization to programmatically request a Web Service to retrieve information as to whether Third-Party transmitters have retrieved the acknowledgements submitted by the states for a particular ETIN. The requesting system will provide a GetNewAckNotificationsRequestArgs message in the SOAP body containing the maximum number of results that should be returned (up to 100) and the MeFHeader message in the SOAP header containing the ETIN. If successful, the service returns a GetNewAckNotificationsResponse message in the SOAP body containing a Boolean flag indicating if more acknowledgement retrieval notifications are available, and a single zip file attachment containing the acknowledgement retrieval notifications. Otherwise, a SOAP error message will be returned to the requesting system.

1.8.3.3.1 element GetNewAckNotificationsRequestArgs

Table 1.8.8-68 Element GetNewAckNotificationsRequestArgsDetails

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	GetNewAckNotificationsRequestType
properties	content complex
children	MaxResultsCnt
source	<pre> <xsd:element name="GetNewAckNotificationsRequestArgs" type="msx:GetNewAckNotificationsRequestType"/> <xsd:complexType name="GetNewAckNotificationsRequestType"> <xsd:sequence> <xsd:element name="MaxResultCnt" type="xsd:positiveInteger"/> </xsd:sequence> </xsd:complexType> </pre>



1.8.3.3.2 element GetNewAckNotificationsResponse

Table 1.8.8-69 Element GetNewAckNotificationsResponse Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	GetNewAckNotificationsResponseType
properties	content complex
children	MoreAvailableInd
source	<pre> <xsd:element name="GetNewAckNotificationsResponse" type="msx:GetNewAckNotificationsResponseType"/> <xsd:complexType name="GetNewAckNotificationsResponseType"> <xsd:sequence> <xsd:element name="MoreAvailableInd" type="xsd:boolean"/> </xsd:sequence> </xsd:complexType> </pre>

1.8.3.4 Get New Submissions

The State applications will have authorization to programmatically request a Web Service to retrieve new submissions for a specified category of state submissions. The requesting system will provide a GetNewSubmissionsRequestArgs message in the SOAP body containing an optional submission category and a MeFHeader message in the SOAP header containing the ETIN. Beginning with MeF Release 6.1, the response message attachment zip file is built by the MeF application in advance of any Get New Submissions Web service calls. If successful, the service returns a GetNewSubmissionsResponse message in the SOAP body and a single pre-built zip file attachment containing the submissions. Otherwise, a SOAP fault error message will be returned to the requesting system. The GetNewSubmissionsResponse message contains a Boolean flag indicating if more submissions with the specified criteria are available and a list of IRS Data structures defined in Section 1.8.3.4.2. The valid submission categories are:

- CORP (Corporations)
- EO (Tax Exempt Organizations)
- PART (Partnerships)
- IND (Individuals)
- ESTRST (Estate & Trust)
- all (CORP or PART or EO or IND or ESTRST)

NOTE: Any particular ETIN requesting “all” will retrieve submissions for any one of CORP, PART, EO, IND or ESTRST depending on the access allowed for that ETIN in the IRS database and the submissions available at the time.



1.8.3.4.1 element GetNewSubmissionsRequestArgs

Table 1.8.8-70 Element GetNewSubmissionsRequestArgs Details

diagram	<p>The diagram shows a class hierarchy. A box labeled 'GetNewSubmissionsRequestAr...' is connected by a solid line to a larger box labeled 'GetNewSubmissionsRequestType'. Inside the 'GetNewSubmissionsRequestType' box, there is a dashed-line box containing 'SubsetStSubmissionCategoryCd'.</p>
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	GetNewSubmissionsRequestType
properties	content complex
children	SubsetStSubmissionCategoryCd
source	<pre> <xsd:element name="GetNewSubmissionsRequestArgs" type="msx:GetNewSubmissionsRequestType"/> <xsd:complexType name="GetNewSubmissionsRequestType"> <xsd:sequence> <xsd:element name="SubsetStSubmissionCategoryCd" type="SubsetStSubmissionCategoryCdType" minOccurs="0"/> </xsd:sequence> </xsd:complexType> </pre>

GetNewSubmissionsRequestType/SubsetStSubmissionCategoryCd

Table 1.8.8-71 Element SubsetStSubmissionCategoryCd Details

Diagram	<p>The diagram shows a box labeled 'StateSubmissionCategoryCd' with a dashed line connecting it to a larger box labeled 'SubsetStSubmissionCategoryCd'.</p>
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	SubsetStSubmissionCategoryCdType
properties	isRef 0 content simple
facets	enumeration CORP enumeration EO enumeration PART enumeration IND enumeration ESTRST enumeration all



source	<code><xsd:element name="SubsetStSubmissionCategoryCd" type="SubsetStSubmissionCategoryCdType" minOccurs="0"/></code>
--------	---

1.8.3.4.2 element GetNewSubmissionsResponse

Table 1.8.8-72 Element GetNewSubmissionsResponse Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	GetNewSubmissionsResponseType
properties	content complex
children	MoreAvailableInd IRSDataForStateSubmissionList
source	<code><xsd:element name="GetNewSubmissionsResponse" type="msx:GetNewSubmissionsResponseType"/></code>

GetNewSubmissionsResponseType/IRSDataForStateSubmissionList

Table 1.8.8-73 Element IRSDataForStateSubmissionList Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	IRSDataForStateSubmissionListType
properties	isRef 0 content complex
children	Cnt IRSDataForStateSubmission
source	<code><xsd:element name="IRSDataForStateSubmissionList" type="msx:IRSDataForStateSubmissionListType"/></code>



IRSDataForStateSubmissionListType/IRSDataForStateSubmission

Table 1.8.8-74 Element IRSDataForStateSubmission Details

<p>diagram</p>	<p>The diagram illustrates the structure of the IRSDataForStateSubmissionType element. It is a complex type containing the following elements in sequence:</p> <ul style="list-style-type: none"> SubmissionId (required) ETIN (required) ReceivedAtIRSTs (required) A choice between FederalEINStatusCd and SSNValidationStatus (required) ElectronicPostmarkTs (optional, shown with a dashed box) SourceIRSSubmissionId (optional, shown with a dashed box) ITINMismatchCd (optional, shown with a dashed box) ImperfectReturnInd (optional, shown with a dashed box) TransmitterIPAddressTxt (optional, shown with a dashed box) TransmitterTs (optional, shown with a dashed box)
<p>namespace</p>	<p>http://www.irs.gov/a2a/mef/MeFStateService.xsd</p>
<p>type</p>	<p>IRSDataForStateSubmissionType</p>
<p>properties</p>	<p>isRef 0 content complex</p>
<p>children</p>	<p>SubmissionId ETIN ReceivedAtIRSTs FederalEINStatusCd SSNValidationStatus ElectronicPostmarkTs SourceIRSSubmissionId ITINMismatchCd ImperfectReturnInd TransmitterIPAddressTxt TransmitterTs</p>
<p>source</p>	<pre><xsd:element name="IRSDataForStateSubmission" type="msx:IRSDataForStateSubmissionType" minOccurs="0" maxOccurs="unbounded"/></pre>



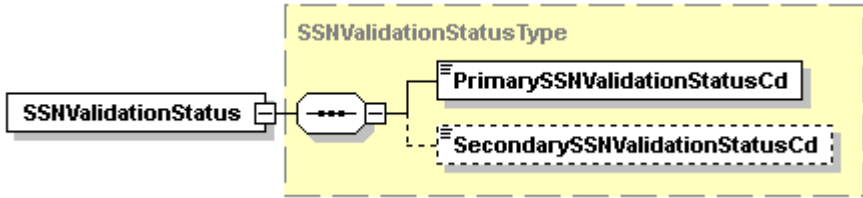

```

<xsd:complexType name="IRSDataForStateSubmissionType">
  <xsd:sequence>
    <xsd:element name="SubmissionId" type="msx:SubmissionIdType"/>
    <xsd:element name="ETIN" type="xsd:string"/>
    <xsd:element name="ReceivedAtIRSTs" type="xsd:dateTime"/>
    <xsd:choice minOccurs="0">
      <xsd:element name="FederalEINStatusCd">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:enumeration value="Found Match"/>
            <xsd:enumeration value="Found No Match"/>
            <xsd:enumeration value="Not Found"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
      <xsd:element name="SSNValidationStatus" type="SSNValidationStatusType"/>
    </xsd:choice>
    <xsd:element name="ElectronicPostmarkTs" type="xsd:dateTime" minOccurs="0"/>
    <xsd:element name="SourceIRSSubmissionId" type="msx:SubmissionIdType" minOccurs="0"/>
    <xsd:element name="ITINMismatchCd" type="xsd:string" minOccurs="0"/>
    <xsd:element name="ImperfectReturnInd" type="xsd:string" minOccurs="0"/>
    <xsd:element name="TransmitterIPAddressTxt" type="xsd:string" minOccurs="0"/>
    <xsd:element name="TransmitterTs" type="xsd:dateTime" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType> </xsd:complexType>

```

IRSDataForStateSubmissionType/SSNValidationStatus

Table 1.8.8-75 Element SSNValidationStatus Details

diagram	 <p>The diagram illustrates the structure of the SSNValidationStatusType. It shows a class SSNValidationStatusType (represented by a dashed yellow box) containing an element SSNValidationStatus. This element is a choice between two options: PrimarySSNValidationStatusCd (represented by a solid box) and SecondarySSNValidationStatusCd (represented by a dashed box).</p>
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	SSNValidationStatusType



properties	isRef 0 content complex
children	PrimarySSNValidationStatusCd SecondarySSNValidationStatusCd
source	<pre> <xsd:element name="SSNValidationStatus" type="SSNValidationStatusType" minOccurs="0"/> <xsd:complexType name="SSNValidationStatusType"> <xsd:sequence> <xsd:element name="PrimarySSNValidationStatusCd"> <xsd:simpleType> <xsd:restriction base="xsd:string"> <xsd:enumeration value="Found Match"/> <xsd:enumeration value="Found No Match"/> <xsd:enumeration value="Not Found"/> <xsd:enumeration value="No Access"/> </xsd:restriction> </xsd:simpleType> </xsd:element> <xsd:element name="SecondarySSNValidationStatusCd" minOccurs="0"> <xsd:simpleType> <xsd:restriction base="xsd:string"> <xsd:enumeration value="Found Match"/> <xsd:enumeration value="Found No Match"/> <xsd:enumeration value="Not Found"/> <xsd:enumeration value="No Access"/> </xsd:restriction> </xsd:simpleType> </xsd:element> </xsd:sequence> </xsd:complexType> </pre>

1.8.3.5 Get Submissions by Message ID

The State applications will have authorization to programmatically request a Web Service to retrieve submissions previously retrieved by a Get New Submissions request. The requesting system will provide a GetSubmissionsByMsgIDRequest message in the SOAP body containing the Message ID used in the previous Get New Submissions request. If successful, the service returns a GetSubmissionsByMsgIDResponse message in the SOAP body and a single zip file attachment containing the submissions. Otherwise, a SOAP error message will be returned to the requesting system.



The GetSubmissionsByMsgIDResponse message contains a Boolean flag indicating if more submissions were available at the time the previous Get New Submissions request was processed and a list of IRS Data structures defined in Section.

1.8.3.5.1 element GetSubmissionsByMsgIDRequestArgs

Table 1.8.8-76 Element GetSubmissionsByMsgIDRequestArgs Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	GetSubmissionsByMsgIDRequestType
properties	content complex
children	Msgageld
source	<pre> <xsd:element name="GetSubmissionsByMsgIDRequestArgs" type="msx:GetSubmissionsByMsgIDRequestType"/> <xsd:complexType name="GetSubmissionsByMsgIDRequestType"> <xsd:sequence> <xsd:element name="Msgageld" type="msx:MsgageldType"/> </xsd:sequence> </xsd:complexType> </pre>

1.8.3.5.2 element GetSubmissionsByMsgIDResponse

Table 1.8.8-77 Element GetSubmissionsByMsgIDResponse Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	GetSubmissionsByMsgIDResponseType
properties	content complex



children	MoreAvailableInd IRSDataForStateSubmissionList
source	<code><xsd:element type="msx:GetSubmissionsByMsgIDResponseType" name="GetSubmissionsByMsgIDResponseType"/></code>

GetSubmissionsByMsgIDResponseType/IRSDataForStateSubmissionList

Table 1.8.8-78 Element IRSDataForStateSubmissionList Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	IRSDataForStateSubmissionListType
properties	isRef 0 content complex
children	Cnt IRSDataForStateSubmission
source	<code><xsd:element name="IRSDataList" type="msx:IRSDataForStateSubmissionListType"/></code>

IRSDataForStateSubmissionListType/IRSDataForStateSubmission



Table 1.8.8-79 Element IRSDataForStateSubmission Details

<p>diagram</p>	
<p>namespace</p>	<p>http://www.irs.gov/a2a/mef/MeFStateService.xsd</p>
<p>type</p>	<p>IRSDataForStateSubmissionType</p>
<p>properties</p>	<p>isRef 0 content complex</p>
<p>children</p>	<p>SubmissionId ETIN ReceivedAtIRSTs FederalEINStatusCd SSNValidationStatus ElectronicPostmarkTs SourceIRSSubmissionId ITINMismatchCd ImperfectReturnInd TransmitterIPAddressTxt TransmitterTs</p>
<p>source</p>	<pre><xsd:element name="IRSData" type="msx:IRSDataForStateSubmissionType" minOccurs="0" maxOccurs="unbounded"/></pre>



```

<xsd:complexType name="IRSDataForStateSubmissionType">
  <xsd:sequence>
    <xsd:element name="SubmissionId" type="msx:SubmissionIdType"/>
    <xsd:element name="ETIN" type="xsd:string"/>
    <xsd:element name="ReceivedAtIRSTs" type="xsd:dateTime"/>
    <xsd:choice minOccurs="0">
      <xsd:element name="FederalEINStatusCd">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:enumeration value="Found Match"/>
            <xsd:enumeration value="Found No Match"/>
            <xsd:enumeration value="Not Found"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
      <xsd:element name="SSNValidationStatus" type="SSNValidationStatusType"/>
    </xsd:choice>
    <xsd:element name="ElectronicPostmarkTs" type="xsd:dateTime" minOccurs="0"/>
    <xsd:element name="SourceIRSSubmissionId" type="msx:SubmissionIdType" minOccurs="0"/>
    <xsd:element name="ITINMismatchCd" type="xsd:string" minOccurs="0"/>
    <xsd:element name="ImperfectReturnInd" type="xsd:string" minOccurs="0"/>
    <xsd:element name="TransmitterIPAddressTxt" type="xsd:string" minOccurs="0"/>
    <xsd:element name="TransmitterTs" type="xsd:dateTime" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

```

IRSData/SSNValidationStatus

Table 1.8.8-80 Element SSNValidationStatus Details

diagram	<p>The diagram illustrates the structure of the SSNValidationStatus element. It shows a class SSNValidationStatus on the left, which has a composition relationship (indicated by a solid line with an open arrowhead) with a class SSNValidationStatusType. The SSNValidationStatusType class is highlighted in a yellow box and contains two subclasses: PrimarySSNValidationStatusCd and SecondarySSNValidationStatusCd. The relationship between the superclass and its subclasses is shown with solid lines and open arrowheads, indicating inheritance.</p>
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd



type	SSNValidationStatusType
properties	isRef 0 content complex
children	PrimarySSNValidationStatusCd SecondarySSNValidationStatusCd
source	<pre> <xsd:element name="SSNValidationStatus" type="SSNValidationStatusType" minOccurs="0"/> <xsd:complexType name="SSNValidationStatusType"> <xsd:sequence> <xsd:element name="PrimarySSNValidationStatusCd"> <xsd:simpleType> <xsd:restriction base="xsd:string"> <xsd:e value="Found Match"/> <xsd:enumeration value="Found No Match"/> <xsd:enumeration value="Not Found"/> <xsd:enumeration value="No Access"/> </xsd:restriction> </xsd:simpleType> </xsd:element> <xsd:element name="SecondarySSNValidationStatusCd" minOccurs="0"> <xsd:simpleType> <xsd:restriction base="xsd:string"> <xsd:enumeration value="Found Match"/> <xsd:enumeration value="Found No Match"/> <xsd:enumeration value="Not Found"/> <xsd:enumeration value="No Access"/> </xsd:restriction> </xsd:simpleType> </xsd:element> </xsd:sequence> </xsd:complexType> </pre>

1.8.3.6 Get Submission

States will have authorization to programmatically request a Web Service to retrieve a submission for a specified submission ID. The requesting system will provide a GetSubmissionRequestArgs message in the SOAP body containing the submission ID. If successful, the service returns a GetSubmissionResponse



message in the SOAP body and a single zip file attachment containing the submission. Otherwise, a SOAP error message will be returned to the requesting system.

The GetSubmissionResponse message contains the IRS Data structure defined in Section .

1.8.3.6.1 element GetSubmissionRequestArgs

Table 1.8.8-81 Element GetSubmissionRequestArgs Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	GetSubmissionRequestType
properties	content complex
children	SubmissionId
source	<code><xsd:element name="GetSubmissionRequestArgs" type="msx:GetSubmissionRequestType"/></code>

1.8.3.6.2 element GetSubmissionResponse

Table 1.8.8-82 Element GetSubmissionResponse Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	GetSubmissionResponseType
properties	content complex
children	IRSDataForStateSubmission
source	<code><xsd:element name="GetSubmissionResponse" type="msx:GetSubmissionResponseType"/></code>



GetSubmissionResponseType/IRSDataForStateSubmission

Table 1.8.8-83 Element IRSDataForStateSubmission Details

<p>diagram</p>	<p>The diagram shows the structure of the <code>IRSDataForStateSubmissionType</code> element. It is a complex type containing the following elements:</p> <ul style="list-style-type: none"> <code>SubmissionId</code> <code>ETIN</code> <code>IRSReceivedTs</code> A choice between <code>FederalEINStatusCd</code> and <code>SSNValidationStatus</code>. <code>ElectronicPostmarkTs</code> <code>IRSSubmissionId</code> <code>ITINMismatchCd</code> <code>ImperfectReturnInd</code> <code>TransmitterIPAddressTxt</code> <code>TransmitterTs</code>
<p>namespace</p>	<p>http://www.irs.gov/a2a/mef/MeFStateService.xsd</p>
<p>type</p>	<p>IRSDataForStateSubmissionType</p>
<p>properties</p>	<p>isRef 0 content complex</p>
<p>children</p>	<p>SubmissionId ETIN ReceivedAtIRSTs FederalEINStatusCd SSNValidationStatus ElectronicPostmarkTs SourceIRSSubmissionId ITINMismatchCd ImperfectReturnInd TransmitterIPAddressTxt TransmitterTs</p>
<p>source</p>	<p><code><xsd:element name="IRSData" type="msx:IRSDataForStateSubmissionType" maxOccurs="unbounded"/></code></p>



```

<xsd:complexType name="IRSDataForStateSubmissionType">
  <xsd:sequence>
    <xsd:element name="SubmissionId" type="msx:SubmissionIdType"/>
    <xsd:element name="ETIN" type="xsd:string"/>
    <xsd:element name="ReceivedAtIRSTs" type="xsd:dateTime"/>
    <xsd:choice minOccurs="0">
      <xsd:element name="FederalEINStatusCd">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:enumeration value="Found Match"/>
            <xsd:enumeration value="Found No Match"/>
            <xsd:enumeration value="Not Found"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
      <xsd:element name="SSNValidationStatus" type="SSNValidationStatusType"/>
    </xsd:choice>
    <xsd:element name="ElectronicPostmarkTs" type="xsd:dateTime" minOccurs="0"/>
    <xsd:element name="SourceIRSSubmissionId" type="msx:SubmissionIdType" minOccurs="0"/>
    <xsd:element name="ITINMismatchCd" type="xsd:string" minOccurs="0"/>
    <xsd:element name="ImperfectReturnInd" type="xsd:string" minOccurs="0"/>
    <xsd:element name="TransmitterIPAddressTxt" type="xsd:string" minOccurs="0"/>
    <xsd:element name="TransmitterTs" type="xsd:dateTime" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

```

IRSData/SSNValidationStatus

Table 1.8.8-84 Element SSNValidationStatus Details

diagram	<p>The diagram shows a class SSNValidationStatus connected to a class SSNValidationStatusType. Inside SSNValidationStatusType, there are two subclasses: PrimarySSNValidationStatusCd (represented by a solid box) and SecondarySSNValidationStatusCd (represented by a dashed box). A dashed line indicates a relationship between the SSNValidationStatus class and the SecondarySSNValidationStatusCd class.</p>
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	SSNValidationStatusType



properties	isRef 0 content complex
children	PrimarySSNValidationStatusCd SecondarySSNValidationStatusCd
source	<pre> <xsd:element name="SSNValidationStatus" type="SSNValidationStatusType" minOccurs="0"/> <xsd:complexType name="SSNValidationStatusType"> <xsd:sequence> <xsd:element name="PrimarySSNValidationStatusCd"> <xsd:simpleType> <xsd:restriction base="xsd:string"> <xsd:enumeration value="Found Match"/> <xsd:enumeration value="Found No Match"/> <xsd:enumeration value="Not Found"/> <xsd:enumeration value="No Access"/> </xsd:restriction> </xsd:simpleType> </xsd:element> <xsd:element name="SecondarySSNValidationStatusCd" minOccurs="0"> <xsd:simpleType> <xsd:restriction base="xsd:string"> <xsd:enumeration value="Found Match"/> <xsd:enumeration value="Found No Match"/> <xsd:enumeration value="Not Found"/> <xsd:enumeration value="No Access"/> </xsd:restriction> </xsd:simpleType> </xsd:element> </xsd:sequence> </xsd:complexType> </pre>

1.8.3.7 Get Submissions

States will have authorization to programmatically request a Web Service to retrieve submissions for a specified list of submission IDs. The requesting system will provide a GetSubmissionsRequestArgs message in the SOAP body containing a list of submission IDs. If successful, the service returns a GetSubmissionsResponse message in the SOAP body and a single zip file attachment containing the submissions. Otherwise, a SOAP error message will be returned to the requesting system.



The GetSubmissionsResponse message contains a list of IRSDataForStateSubmission structures defined in Section 1.8.3.6.2 and a list of submission errors for any submission IDs from the submission ID list that were not found.

1.8.3.7.1 element GetSubmissionResponse

Table 1.8.8-85 Element GetSubmissionsRequestArgs Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	GetSubmissionsRequestType
properties	content complex
children	SubmissionIdList
source	<pre> <xsd:element name="GetSubmissionsRequestArgs" type="msx:GetSubmissionsRequestType"/> <xsd:complexType name="GetSubmissionsRequestType"> <xsd:sequence> <xsd:element name="SubmissionIdList" type="msx:SubmissionIdListType"/> </xsd:sequence> </xsd:complexType> <xsd:complexType name="SubmissionIdListType"> <xsd:sequence> <xsd:element name="SubmissionId" type="SubmissionIdType" maxOccurs="unbounded"/> </xsd:sequence> </xsd:complexType> </pre>



1.8.3.7.2 element GetSubmissionsResponse

Table 1.8.8-86 Element GetSubmissionsResponse Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	GetSubmissionsResponseType
properties	content complex
children	IRSDataForStateSubmissionList SubmissionErrorList
source	<code><xsd:element name="GetSubmissionsResponse" type="msx:GetSubmissionsResponseType"/></code>

GetSubmissionsResponseType/IRSDataForStateSubmissionList

Table 1.8.8-87 Element IRSDataForStateSubmissionList Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	IRSDataForStateSubmissionListType
properties	isRef 0 content complex
children	Cnt IRSDataForStateSubmission
source	<code><xsd:element name="IRSDataList" type="msx:IRSDataForStateSubmissionListType"/></code>



IRSDataForStateSubmissionListType/IRSDataForStateSubmission

Table 1.8.8-88 Element IRSDataForStateSubmission Details

<p>diagram</p>	
<p>namespace</p>	<p>http://www.irs.gov/a2a/mef/MeFStateService.xsd</p>
<p>type</p>	<p>IRSDataForStateSubmissionType</p>
<p>properties</p>	<p>isRef 0 content complex</p>
<p>children</p>	<p>SubmissionId ETIN ReceivedAtIRSTs FederalEINStatusCd SSNValidationStatus ElectronicPostmarkTs SourceIRSSubmissionId ITINMismatchCd ImperfectReturnInd TransmitterIPAddressTxt TransmitterTs</p>
<p>source</p>	<pre><xsd:element name="IRSData" type="msx:IRSDataForStateSubmissionType" minOccurs="0" maxOccurs="unbounded"/></pre>



```

<xsd:complexType name="IRSDataForStateSubmissionType">
  <xsd:sequence>
    <xsd:element name="SubmissionId" type="msx:SubmissionIdType"/>
    <xsd:element name="ETIN" type="xsd:string"/>
    <xsd:element name="ReceivedAtIRSTs" type="xsd:dateTime"/>
    <xsd:choice minOccurs="0">
      <xsd:element name="FederalEINStatusCd">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:enumeration value="Found Match"/>
            <xsd:enumeration value="Found No Match"/>
            <xsd:enumeration value="Not Found"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
      <xsd:element name="SSNValidationStatus" type="SSNValidationStatusType"/>
    </xsd:choice>
    <xsd:element name="ElectronicPostmarkTs" type="xsd:dateTime" minOccurs="0"/>
    <xsd:element name="SourceIRSSubmissionId" type="msx:SubmissionIdType" minOccurs="0"/>
    <xsd:element name="ITINMismatchCd" type="xsd:string" minOccurs="0"/>
    <xsd:element name="ImperfectReturnInd" type="xsd:string" minOccurs="0"/>
    <xsd:element name="TransmitterIPAddressTxt" type="xsd:string" minOccurs="0"/>
    <xsd:element name="TransmitterTs" type="xsd:dateTime" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

```

IRSDataForStateSubmission/SSNValidationStatus

Table 1.8.8-89 Element SSNValidationStatus Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	SSNValidationStatusType



properties	isRef 0 content complex
children	PrimarySSNValidationStatusCd SecondarySSNValidationStatusCd
source	<pre> <xsd:element name="SSNValidationStatus" type="SSNValidationStatusType" minOccurs="0"/> <xsd:complexType name="SSNValidationStatusType"> <xsd:sequence> <xsd:element name="PrimarySSNValidationStatusCd"> <xsd:simpleType> <xsd:restriction base="xsd:string"> <xsd:enumeration value="Found Match"/> <xsd:enumeration value="Found No Match"/> <xsd:enumeration value="Not Found"/> <xsd:enumeration value="No Access"/> </xsd:restriction> </xsd:simpleType> </xsd:element> <xsd:element name="SecondarySSNValidationStatusCd" minOccurs="0"> <xsd:simpleType> <xsd:restriction base="xsd:string"> <xsd:enumeration value="Found Match"/> <xsd:enumeration value="Found No Match"/> <xsd:enumeration value="Not Found"/> <xsd:enumeration value="No Access"/> </xsd:restriction> </xsd:simpleType> </xsd:element> </xsd:sequence> </xsd:complexType> </pre>

1.8.3.8 Send Acks Request

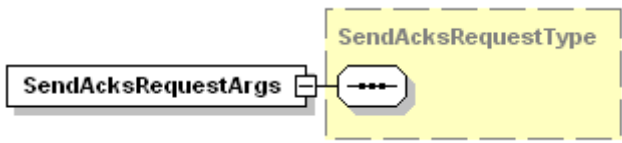
States will have authorization to programmatically request a Web Service to acknowledge the processing of state returns. The requesting system will provide an empty SendAcksRequestArgs tag in the SOAP body and a single zip file attachment containing the acknowledgements. The requesting system must also already be logged in, have a valid session, and provide the session key in the wsse:Security portion of the SOAP header. If successful, the service returns a SendAcksResponse message in the SOAP body containing a list of acknowledgements that are duplicates or have failed validation. If all the



acknowledgements are accepted, then the response will contain an empty list with a count of zero. If any other error occurs, a SOAP fault message will be returned to the requesting system.

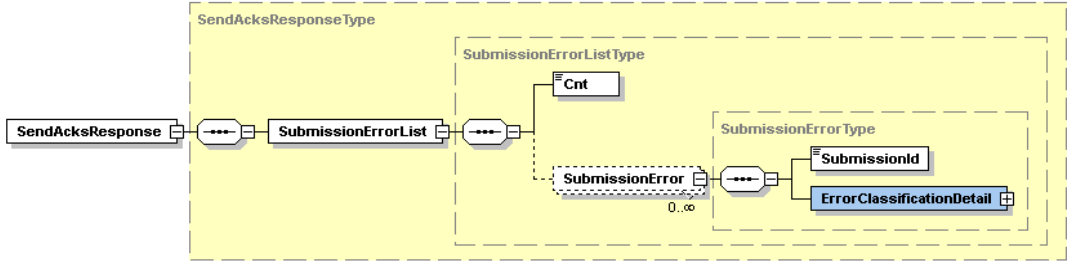
1.8.3.8.1 element SendAcksRequestArgs

Table 1.8.8-90 Element SendAcksRequestArgs Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	SendAcksRequestType
properties	content complex
source	<code><xsd:element name="SendAcksRequestArgs" type="msx:SendAcksRequestType"/></code>

1.8.3.8.2 element SendAcksResponse

Table 1.8.8-91 Element SendAcksResponse Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	SendAcksResponseType
properties	content complex
children	SubmissionErrorList
source	<code><xsd:element name="SendAcksResponse" type="msx:SendAcksResponseType"/></code>



```

<xsd:complexType name="SendAcksResponseType">
  <xsd:sequence>
    <xsd:element name="SubmissionErrorList" type="msx:SubmissionErrorListType"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SubmissionErrorListType">
  <xsd:sequence>
    <xsd:element name="Cnt" type="xsd:nonNegativeInteger"/>
    <xsd:element name="SubmissionError" type="SubmissionErrorType" minOccurs="0"
maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SubmissionErrorType">
  <xsd:sequence>
    <xsd:element name="SubmissionId" type="SubmissionIdType"/>
    <xsd:element name="ErrorClassificationDetail" type="ErrorClassificationDetailType"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ErrorClassificationDetailType">
  <xsd:sequence>
    <xsd:element name="ErrorMessageTxt" type="xsd:string"/>
    <xsd:element name="ErrorClassificationCd" type="ErrorClassificationCdType"/>
    <xsd:element name="ErrorMessageCd" type="xsd:string" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

```

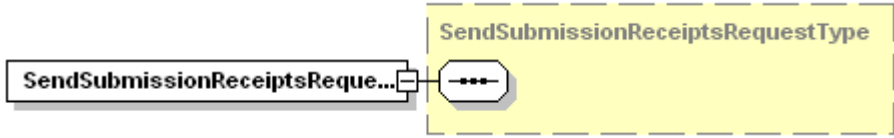
1.8.3.9 Send Submission Receipts

States will have authorization to programmatically request a Web Service to indicate that the specified state returns have been successfully received by the state's application. The requesting system will provide an empty `SendSubmissionReceiptsRequestArgs` tag in the SOAP body and a single zip file attachment containing the receipts for the state returns. The requesting system must also already be logged in, have a valid session, and provide the session key in the `wsse:Security` portion of the SOAP header. If successful, the service returns a `SendSubmissionReceiptsResponse` message in the SOAP body indicating the number of receipts accepted and a `ListOfSubmissionErrors` for any submissions for which receipts have already been received or any receipts with and invalid timestamp. Also, if any other error occurs, a SOAP fault message will be returned to the requesting system.



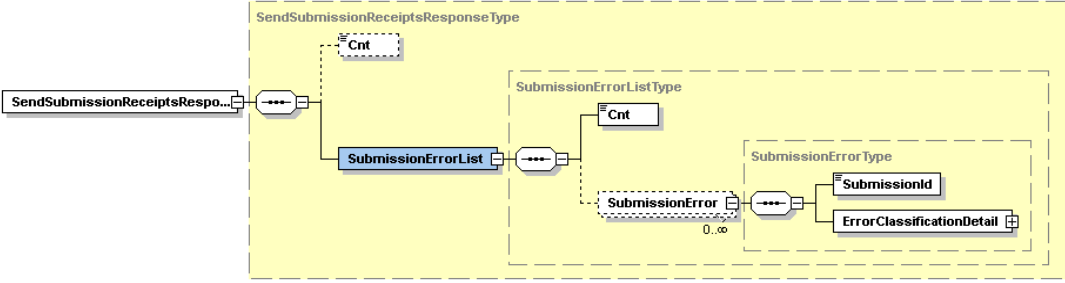
1.8.3.9.1 element SendSubmissionReceiptsRequestArgs

Table 1.8.8-92 Element SendSubmissionReceiptsRequestArgs Details

diagram	 <p>The diagram shows a box labeled 'SendSubmissionReceiptsReque...' connected to a dashed-line box labeled 'SendSubmissionReceiptsRequestType'. Inside the dashed box is a small box with four dashes '----'.</p>
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	SendSubmissionReceiptsRequestType
properties	content complex
source	<code><xsd:element name="SendSubmissionReceiptsRequestArgs" type="msx:SendSubmissionReceiptsRequestType"/></code>

1.8.3.9.2 element SendSubmissionReceiptsResponse

Table 1.8.8-93 Element SendSubmissionReceiptsResponse Details

diagram	 <p>The diagram shows a box labeled 'SendSubmissionReceiptsRespo...' connected to a dashed-line box labeled 'SendSubmissionReceiptsResponseType'. Inside this dashed box is a box labeled 'Cnt' connected to a box labeled 'SubmissionErrorList'. 'SubmissionErrorList' is connected to another dashed-line box labeled 'SubmissionErrorListType'. Inside 'SubmissionErrorListType' is a box labeled 'Cnt' connected to a box labeled 'SubmissionError'. 'SubmissionError' is connected to a dashed-line box labeled 'SubmissionErrorType'. Inside 'SubmissionErrorType' are boxes labeled 'SubmissionId' and 'ErrorClassificationDetail'. The 'SubmissionError' box has a cardinality of '0..∞'.</p>
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	SendSubmissionReceiptsResponseType
properties	content complex
children	Cnt SubmissionErrorList
source	<code><xsd:element name="SendSubmissionReceiptsResponse" type="msx:SendSubmissionReceiptsResponseType"/></code>



	<pre> <xsd:complexType name="SendSubmissionReceiptsResponseType"> <xsd:sequence> <xsd:element name="Cnt" type="xsd:nonNegativeInteger" minOccurs="0"/> <xsd:element name="SubmissionErrorList" type="msx:SubmissionErrorListType"/> </xsd:sequence> </xsd:complexType> </pre>
--	---

1.8.3.9 Get Submission Reconciliation List

The State applications will have authorization to programmatically request a Web Service to retrieve a list of state return submission IDs for submissions that MeF has received and are destined to be retrieved and processed by the state. The requesting system will provide a GetSubmissionReconciliationListRequest message in the SOAP body containing the maximum number of submission IDs to be returned (up to 500) and optionally the submission category and/or start date and end date and/or submission status and a MeFHeader message in the SOAP header containing the ETIN. The requesting system can supply just the MaxResultCnt, or the MaxResultCnt and any combination of SubsetStSubmissionCategoryCd, StartDt and EndDt pair and MeFSubmissionStatusCd. Note that if a StartDt element is present, then a EndDt element must also be present and vice versa. If successful, the service returns a GetSubmissionReconciliationListResponse message in the SOAP body. Otherwise, a SOAP fault error message will be returned to the requesting system. The Get Submission Reconciliation List Response message contains a Boolean flag indicating if more submission IDs are available and a list of submission IDs matching the requested criteria.

The valid submission Categories are:

- **CORP (Corporations)**
- **EO (Tax Exempt Organizations)**
- **PART (Partnerships)**
- **IND (Individuals)**
- **ESTRST (Estate & Trust)**
- **all (CORP and/or PART and/or EO and/or IND and/or ESTRST)**

NOTE: Any particular ETIN requesting “all” will retrieve submission IDs for any combination of CORP, PART, EO, IND and ESTRST depending on the access allowed for that ETIN in the IRS database and the submission IDs available at the time.

The valid Submission Status values are:

- **NOTRETRIEVED (Submissions not yet been retrieved by the state)**
- **RECEIPTNOACK (Submissions retrieved and receipted by the state but not yet acknowledged).**
- **NORECEIPTNOACK (Submissions retrieved by the state but not yet receipted or acknowledged)
(This is the default if the optional Submission Status element is not specified)**

If the Submission Status value is NOTRETRIEVED, then the submission IDs in the returned list will represent those that have not yet been retrieved by the state. If the Submission Status value is RECEIPTNOACK, then the submission IDs in the returned list will represent those receipted, but for which an acknowledgement has not been received. Please note that if the submission is acknowledged, then the acknowledgement implies the submission receipt. If the Submission Status value is NORECEIPTNOACK, then the submission IDs in the returned list will represent those that have been retrieved, but for which no



receipt and, therefore, no acknowledgement has been received. In all cases, theTotalNumber value in the response represents the total number of submissions that have not yet been retrieved.

1.8.3.9.1 element GetSubmissionReconciliationListRequest

Table 1.8.8-94 Element GetSubmissionReconciliationListRequest Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	GetSubmissionReconciliationListRequestType
properties	content complex
children	MaxResultCnt SubsetStSubmissionCategoryCd StartDt EndDt MeFSubmissionStatusCd
source	<pre> <xsd:complexType name="GetSubmissionReconciliationListRequestType"> <xsd:sequence> <xsd:element name="MaxResultCnt" type="xsd:positiveInteger"/> <xsd:element name="SubsetStSubmissionCategoryCd" type="SubsetStSubmissionCategoryCdType" minOccurs="0"/> <xsd:element name="StartDt" type="xsd:dateTime" minOccurs="0"/> <xsd:element name="EndDt" type="xsd:dateTime" minOccurs="0"/> <xsd:element name="MeFSubmissionStatusCd" type="MeFSubmissionStatusCdType" minOccurs="0"/> </xsd:sequence> </xsd:complexType> </pre>

GetSubmissionReconciliationListRequestType/MaxResultCnt

Table 1.8.8-95 Element MaxResultCnt Details


diagram	
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd



type	xsd:positiveInteger
properties	isRef 0 content simple
source	<code><xsd:element name="MaxResultCnt" type="xsd:positiveInteger"/></code>


GetSubmissionReconciliationListRequestType/SubsetStSubmissionCategoryCd

Table 1.8.8-96 Element SubsetStSubmissionCategoryCd Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	SubsetStSubmissionCategoryCdType
properties	isRef 0 content simple
facets	enumeration CORP enumeration EO enumeration PART enumeration IND enumeration ESTRST enumeration all
source	<code><xsd:element name="SubsetStSubmissionCategoryCd" type="SubsetStSubmissionCategoryCdType" minOccurs="0"/></code>

GetSubmissionReconciliationListRequestType/MeFSubmissionStatusCd


Table 1.8.8-97 Element MeFSubmissionStatusCd Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	MeFSubmissionStatusCdType
properties	isRef 0 content simple
facets	enumeration NOTRETRIEVED enumeration RECEIPTNOACK enumeration NORECEIPTNOACK
source	<code><xsd:element name="MeFSubmissionStatusCd" type="MeFSubmissionStatusCdType" minOccurs="0"/></code>




GetSubmissionReconciliationListRequestType/StartDt

Table 1.8.8-98 Element StartDt Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	xsd:dateTime
properties	isRef 0 content simple
source	<code><xsd:element name="StartDt" type="xsd:dateTime" minOccurs="0"/></code>

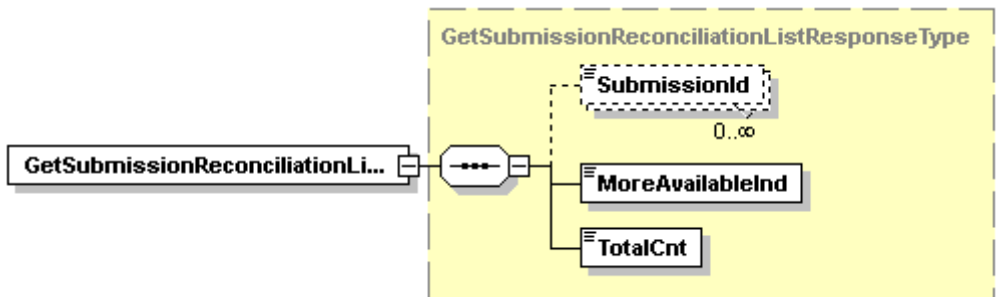
GetSubmissionReconciliationListRequestType/EndDt

Table 1.8.8-99 Element EndDt Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	xsd:dateTime
properties	isRef 0 content simple
source	<code><xsd:element name="EndDt" type="xsd:dateTime" minOccurs="0"/></code>

1.8.3.10.2 element GetSubmissionReconciliationListResponse

Table 1.8.8-100 Element GetSubmissionReconciliationListResponse Details


diagram	 <p>The diagram shows the structure of the GetSubmissionReconciliationListResponse Type. It is a complex type containing a sequence of elements: GetSubmissionReconciliationLi..., SubmissionId (with a cardinality of 0..∞), MoreAvailableInd, and TotalCnt.</p>
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	GetSubmissionReconciliationListResponseType
properties	content complex



children	SubmissionId MoreAvailableInd TotalCnt
source	<pre> <xsd:element name="GetSubmissionReconciliationListResponse" type="msx:GetSubmissionReconciliationListResponseType"/> <xsd:complexType name="GetSubmissionReconciliationListResponseType"> <xsd:sequence> <xsd:element name="SubmissionId" type="SubmissionIdType" minOccurs="0" maxOccurs="unbounded"/> <xsd:element name="MoreAvailableInd" type="xsd:boolean"/> <xsd:element name="TotalCnt" type="xsd:nonNegativeInteger"/> </xsd:sequence> </xsd:complexType> </pre>

element GetSubmissionReconciliationListResponseType/SubmissionId

Table 1.8.8-101 Element SubmissionId Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFStateService.xsd
type	SubmissionIdType
properties	isRef 0 content simple
facets	pattern [0-9]{13}[a-z0-9]{7}
source	<pre> <xsd:element name="SubmissionId" type="SubmissionIdType" minOccurs="0" maxOccurs="unbounded"/> <xsd:simpleType name="SubmissionIdType"> <xsd:restriction base="xsd:string"> <xsd:pattern value="[0-9]{13}[a-z0-9]{7}"/> </xsd:restriction> </xsd:simpleType> </pre>

1.8.4 Exceptions

When errors occur processing a service request a SOAP fault response is returned containing a `ErrorExceptionDetail` message within the SOAP fault detail tag.

1.8.4.1 element ErrorExceptionDetail



Table 1.8.8-102 Element *ErrorExceptionDetail* Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFHeader.xsd
type	mefhdr:ErrorExceptionDetailType
properties	content complex
children	mefhdr:ErrorMessageTxt mefhdr:ErrorClassificationCd mefhdr:ErrorMessageCd mefhdr:HostNm
source	<code><xs:element name="ErrorExceptionDetail" type="mefhdr:ErrorExceptionDetailType"/></code>

1.9 Attachment Files

The A2A and IFA all use “attachment” files as the mechanism to exchange submission and response information with the MeF. Beginning with MeF Release 6.2, every XML message (federal and state submission manifests and submissions) must begin with an XML declaration that looks like `<?xml version="1.0" encoding="UTF-8"?>`. The declaration must always be the first XML statement on the first line of the file beginning in column 1. There may be other optional elements present, but the above format, including the elements `version="1.0"` and `encoding="UTF-8"` must be present and contain exactly the values `1.0` and `UTF-8`. Beginning with Release 9.0, the name of a binary attachment zip file must be less than or equal to 64 characters. Attachment types are as follows:

- **IRS submission**
- **State submission**
- **Acknowledgements**
- **Receipts**
- **Acknowledgement notifications**
- **Status records**
- **Stamped Form 2290 Schedule 1s**

1.9.1 Submission ID

A submission identifier (ID) uniquely identifies a submission, and is present in all attachment files, and many request and response messages. To ensure the global uniqueness of a submission ID, the following format is adopted:

EFIN + ccyydd + xxxxxxx



For example:

Submission ID: 00349720060731234567

The first six digits (003497) contain the Electronic Filer Identification Number (EFIN), the next four digits (2006) contain the year, the next three digits (073) contain the Julian date, and the last seven digits (1234567) contain a sequence number to uniquely identify messages sent within a day with the given EFIN. The total number of characters of the submission ID is 20.

1.9.2 IRS Submission

An IRS submission is sent from a transmitter to the IRS. The structure of an IRS submission is shown in Figure 1-4 below.

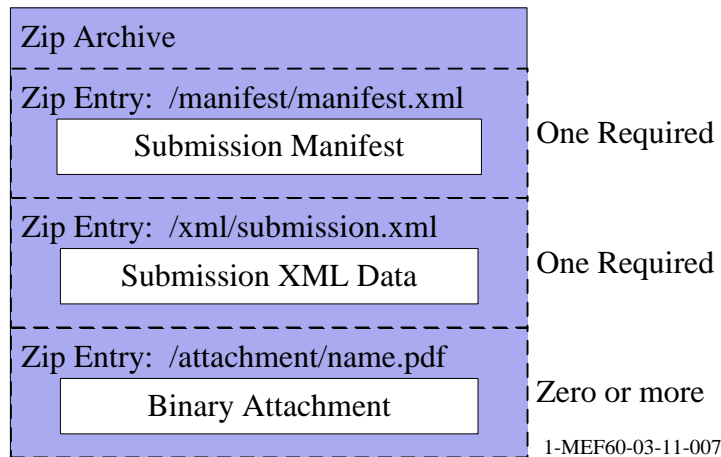


Figure 1-4: IRS Submission Message Structure

- **Submission Manifest** – This is an XML document containing metadata about the submission. The data elements are defined as IRSSubmissionManifest in the XML schema (common/eFileAttachments.xsd) available from the IRS elsewhere.
- **Submission XML Data** – This is XML data conforming to an IRS XML schema for the indicated submission type. The XML schema depends on the submission type and is available from the IRS elsewhere, as discussed in Publication 5446, *MeF Submission Composition Guide*.
- **Binary Attachment** – A PDF file containing a taxpayer business document associated with the submission. Information describing the binary file is included in the submission XML data, according to IRS XML schema definitions.



1.9.3 State Submission

A state submission is created by an originator, and is then sent from a transmitter to the IRS. The structure of a state submission is shown in Figure 1–5.

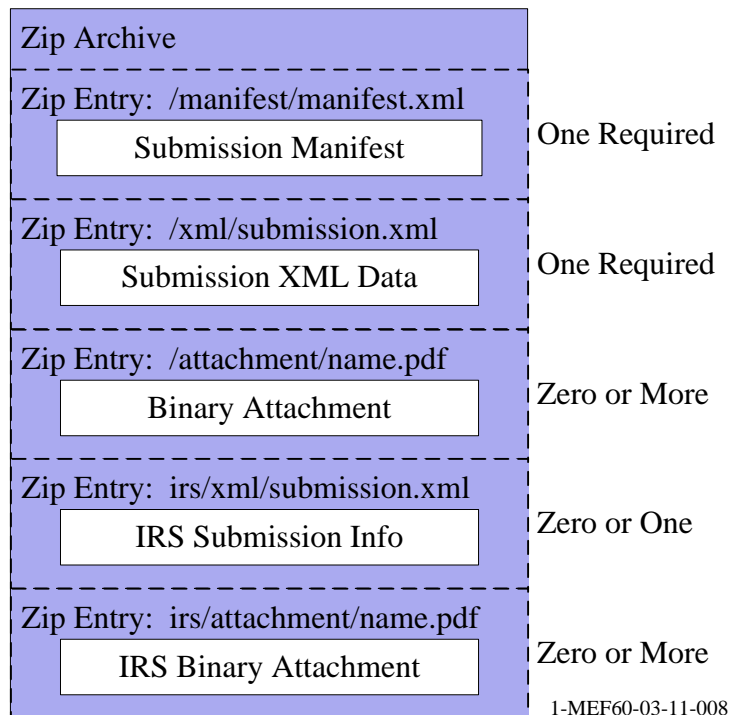


Figure 1-5: State Submission Message Structure

- **Submission Manifest** – This is an XML document containing metadata about the submission. The data elements are defined as IRSSubmissionManifest in the XML schema (common/eFileAttachments.xsd) available from the IRS elsewhere.
- **Submission XML Data** – This is XML data conforming to the state-provided XML schema associated with the indicated submission type.
- **Binary Attachment** – A PDF file containing business content as allowed by rules defined by the receiving state agency.
- **IRS Submission Info** – This is an XML document provided by the taxpayer containing information derived from an IRS submission, conforming to rules specified by the receiving state agency.
- **IRS Binary Attachment** – A PDF document associated with the IRS submission info. A standalone state submission will not have this element.

State Submission Response Structure

A state may receive more than one state submissions in response to a request for submissions. State submissions are combined into one ZIP archive that is returned in the response. The data structure for one or more state submission responses is shown in Figure 1-6.



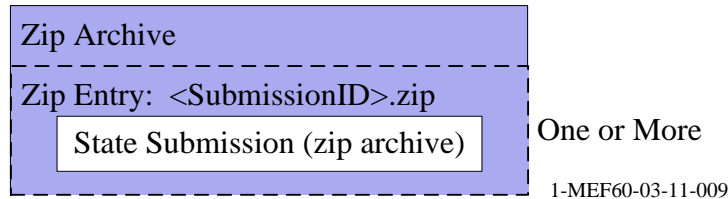


Figure 1-6: State Submission Response Message Structure

- **State Submission – Each outgoing message attachment (zip file) contains one or more submission archives. Each submission IRS archive represents a single state submission.**

1.9.4 IRS Data for State Submission (IRS Submission Info)

When a state submission is sent to the state agency, the IRS also provides additional data derived from its own processing of the state submission. This IRS data is associated with one state submission. The content of the IRS data is shown in Table 1-3, IRS Data for State Submission Elements.

Table 1.9-1: IRS Data for State Submission Elements

Element	Definition
Submission ID	The submission ID of the received submission.
ETIN	The identifier of the transmitter that sent this submission to the IRS.
Received Timestamp	The date and time when the submission was received by the IRS.
Federal EIN Status or SSN Validation Status	(Optional)The results of the Federal EIN and name control lookup. (Valid Codes are: "Found Match", "Found No Match" or "Not Found")or The results of the SSN and name control lookup. (Valid Codes are: "Found Match", "Found No Match" or "Not Found" or "No Access").
Electronic Postmark	(Optional) The time stamp indicating when the transmitter received the originated submission before sending it on to the IRS.
Source IRS Submission ID	(Optional) The submission ID of an IRS submission from which the state copy was made (only for EO, currently).
ITIN Mismatch Code	(Optional) An indication if the linked federal Individual Primary TIN is an ITIN Mismatch (Codes TBD.)
Imperfect Return Indicator	(Optional) An indication if the linked federal Individual taxpayer return was accepted with an attribute of exception.



1.9.5 Acknowledgements

State tax agencies send one state acknowledgement for each state submission it has processed to the IRS. Transmitters can then retrieve the acknowledgements from the IRS. If the state sends more than one acknowledgement to the IRS in one request, the acknowledgements are combined into one container XML document that is then put into to a ZIP archive. When acknowledgements are received by transmitters, a submission ID which was sent to the IRS multiple times may have multiple acknowledgements, one for each of the receipt ID's returned on separate sends. Only one of those multiple acknowledgements will represent a submission's being thoroughly processed by the application, and all the others must be rejects. The data structure for one or more acknowledgements is shown in Figure 1-7, Acknowledgements Structure.

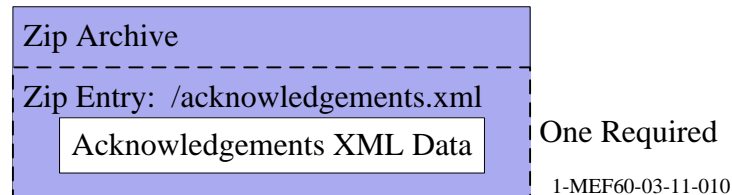


Figure 1-7: Acknowledgements Structure

- **Acknowledgements XML Data – This is an XML document containing acknowledgements. The A2A Web Services and IFA channels both receive submission acknowledgements. The data elements are defined as Acknowledgement in the XML schema (common/eFileAttachments.xsd) available from the IRS elsewhere.**

1.9.6 Receipts

Receipts are sent from the IRS to a transmitter informing that a list of submissions has been received. Receipts are also sent from a state tax agency to the IRS, informing that a list of submissions has been successfully retrieved. They share the same data structure. The structure for receipts data is shown in Figure 1-8.

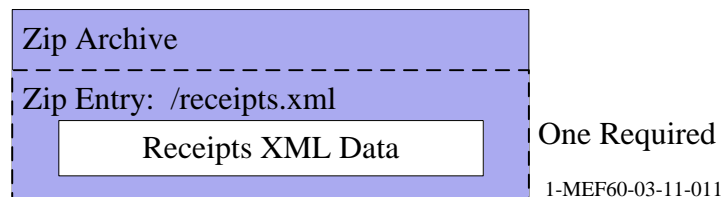


Figure 1-8: Receipts Structure

- **Receipts XML Data – This is an XML document containing one or more submission receipts. The data elements for each receipt are defined as SubmissionReceiptGrp in the XML schema (common/eFileAttachments.xsd) available from the IRS elsewhere. The individual elements are SubmissionId, SubmissionReceivedTs, and ReceiptId (an alphanumeric up to 80 characters long).**



1.9.7 Acknowledgement Notifications

Acknowledgement notifications are retrieved from the IRS by a state tax agency. Each notification informs the state that the appropriate transmitter has received a state acknowledgement. The data structure for one or more acknowledgement notifications is shown in Figure 1-9.

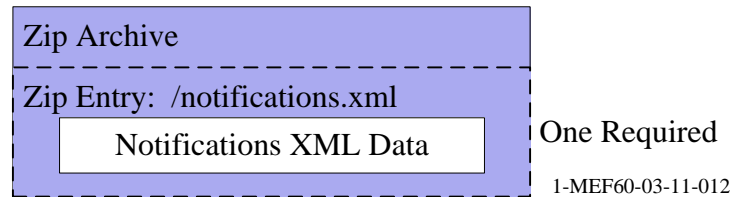


Figure 1-9: Acknowledgement Notifications Structure

- Notifications XML Data** – This is an XML document containing one or more notifications. The data elements for each notification are defined as AckNotification in the XML schema (common/eFileAttachments.xsd) available from the IRS elsewhere.

1.9.8 Status Records

Status Records are sent from the IRS to a transmitter. The structure for sending one or more status records is shown in Figure 1-10.

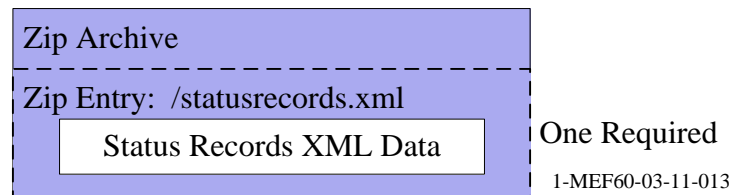


Figure 1-10: Status Records Structure

- Status Records XML Data** – This is an XML document containing status records. The data elements for each status record are defined as StatusRecordGrp in the XML schema (common/eFileAttachments.xsd) available from the IRS elsewhere. The individual elements are SubmissionId, SubmissionStatusTxt, SubmsnStatusAcknowledgementDt, and DisclaimerTxt. The options for the submission status descriptions are shown in Table 1-4, Status Description Options.

Table 1.9.9-2: Status Description Options

Submission Status Description
<p>The status of a submission with respect to events based on external systems activities. The possible values include:</p> <p>ACK_A_CREATED – A status encapsulation for "Accepted Acknowledgment Created" indicates an acknowledgement for an accepted submission has been created by the MeF System for the submission (IRS).</p> <p>ACK_R_CREATED – A status encapsulation for "Rejected Acknowledgment Created" indicates an acknowledgement for a rejected submission has been created by the MeF System for the submission (IRS).</p> <p>ACK_A_RETRIEVED – A status encapsulation for "Accepted Acknowledgment Retrieved" indicates an accepted acknowledgement has been retrieved by the transmitter (for IRS).</p> <p>ACK_R_RETRIEVED – A status encapsulation for "Rejected Acknowledgment Retrieved" indicates a rejected acknowledgement has been retrieved by the transmitter (for IRS).</p>



Submission Status Description
<p>ACKD_BYSTATE – A status encapsulation for “Acknowledged By State” indicates the state has sent to the MeF System an acknowledgment for the State submission.</p> <p>ACK_RETRIEVED – A status encapsulation for "Acknowledgment Retrieved" indicates an acknowledgement issued by State has been retrieved by the transmitter (for State sub).</p> <p>DENIED – A status encapsulation for “Denied by IRS” indicates the state submission has been denied by the IRS.</p> <p>NOTIFIED – A status encapsulation for "Notified" indicates that the IRS has notified the state that the state-generated acknowledgement has been retrieved by the transmitter.</p> <p>READY – A status encapsulation for "Ready for Pickup" indicates that the MeF System has made the State submission available to the state for retrieval.</p> <p>RECEIVED – A status that encapsulation for "Received" indicates the IRS submission has been received by the MeF System (IRS or State).</p> <p>STATE_RECVD – A status encapsulation for "Received by State" indicates that the state has sent to the MeF System a receipt that the State submission has been received.</p> <p>STATE_RETR – A status encapsulation for "Sent to State" indicates the state has tried to retrieve the State submission.</p>
Retrieval Status Description
<p>The current retrieval status. The possible values include:</p> <p>G = Status corresponding to the initial generation of the retrievable entity.</p> <p>R = Status corresponding to the fact that the retrievable entity has been retrieved by a requesting party.</p>

1.9.9 Stamped Form 2290 Schedule 1s

The Stamped Form 2290 Schedule 1s are contained in a zip file attachment within the response message. The ZIP file includes one or more PDF files. Each PDF represents an IRS “stamped” version of a Schedule 1 from an accepted form 2290 submission. The structure of a Stamped Form 2290 Schedule 1s attachment is shown in Figure 1-11.

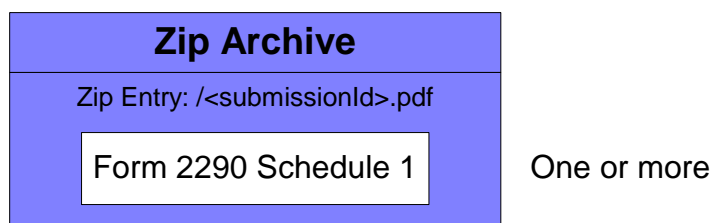


Figure 1-11: Stamped Form 2290 Schedule 1s Attachment Structure

1.9.9 Errors

When a state or third party transmitter provides a list of submission identifiers for data retrieval but some of those submissions are not found in the MeF Application system, a list of errors will be returned, together with the list of data items that are found. The data elements for each error record are defined as SubmissionError in Table 1.8-29 and other tables above.

1.10 IFA Service Messages

For file transmissions through the IFA channels, only the MIME Multipart structure (as illustrated in Figure 1-12) is supported by the MeF system.



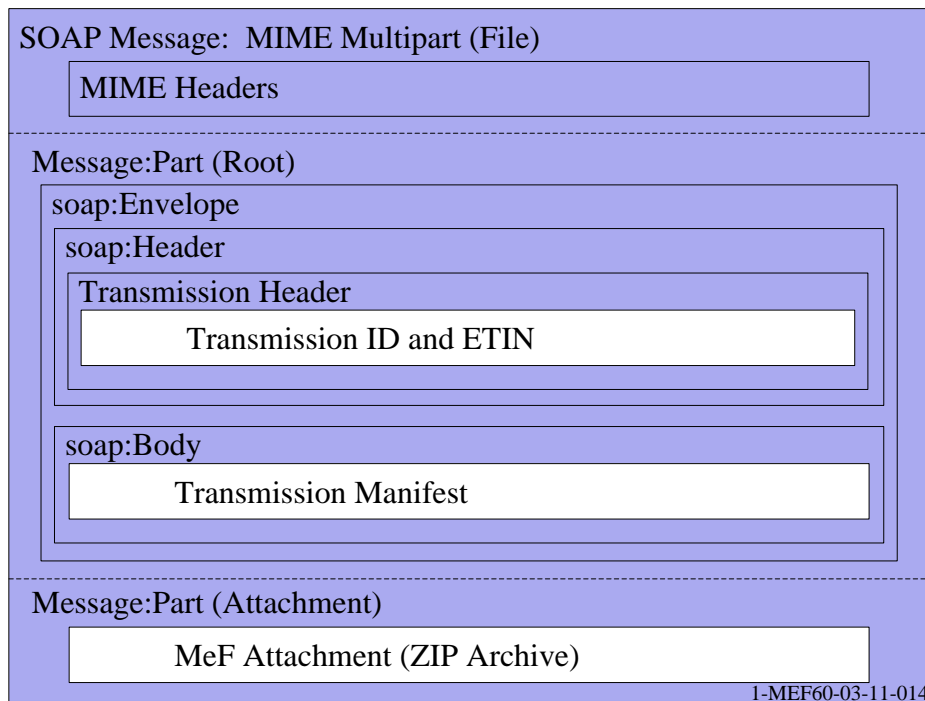


Figure 1-12: IFA Transmission File Structure

1.11 A2A Web Services Error Handling and Exception Reporting

Countermeasures to prevent internal implementation details from being revealed to the Third Party Transmitters or State applications are:

- **Use exception handling throughout application's code**
- **Handle and log exceptions that are allowed to propagate to the application boundary**
- **Return generic, harmless error messages to the external client**

1.11.1 XML Gateway Exceptions

When the XML Gateway detects an error in an A2A service request, the client process is notified via a SOAP fault response. The XML Gateway fault responses are documented in Section 5.1

An example of a SOAP fault for a Login request with incorrect authentication, such as an invalid or improperly registered certificate, is shown below.

```
<soap:Body>
  <soap:Fault>
    <faultcode>soap:Server</faultcode>
    <faultstring>Unauthorized request</faultstring>
    <detail>
      <fs:Error
```




```

xmlns:fs="http://www.forumsystems.com/2004/04/error">
<fs:Message>Unauthorized request</fs:Message>
<fs:User></fs:User>
<fs:Policy>ALT-MeF-MSI</fs:Policy>
<fs:TaskList>MeF-Ping_256</fs:TaskList>
<fs:Task>User Identity & Access Control</fs:Task>
</fs:Error>
</detail>
</soap:Fault></soap:Body>

```

An example of a SOAP fault for a Login request with incorrect authorization, such as an attempt to use a service not allowed for the given userid, is shown below.

```

<soap:Body>
<soap:Fault>
<faultcode>soap:Client</faultcode>
<faultstring xml:lang="en-US">Authorization Failure</faultstring>
<detail>
<fs:Detail
xmlns:fs="http://www.forumsystems.com/2004/04/soap-fault-detail">
<fs:Message>IDP Rule 'ALT-MeF Process Error IDP Rule' aborted processing.
Task 'Control' received an error from the remote server</fs:Message>
<fs:User />
<fs:Policy>ALT-MeF-MSI</fs:Policy>
<fs:TaskList>Ping-SAML-Enriched</fs:TaskList>
<fs:Task>Control</fs:Task>
</fs:Detail>
</detail>
</soap:Fault>
</soap:Body>

```

1.11.2 Faults Response

Fault messages are correlated as replies using the [relationship] property. The [action] property below designates WS-Addressing fault messages:

<http://schemas.xmlsoap.org/ws/2004/08/addressing/fault>

The WS-Addressing definitions of faults use the following properties:

[Subcode] The fault subcode.

[Reason] The English language reason element.

1.11.2.1 MeF Fault

When faults (or errors) occur while processing a MSI service request, the client process is notified via a SOAP fault response containing a `ErrorExceptionDetail` message within the SOAP fault detail tag. This interface is defined as a `MeFFault` message with the only element being the complex type `ErrorExceptionDetail` as defined below.



MeFaultMessage

Table 1.11.11-1 MeFaultMessage

parts	MeFault element mefhdr:ErrorExceptionDetail
used by	PortType EtinRetrieval in Operation EtinRetrieval PortType EtinStatus in Operation EtinStatus PortType GetStateParticipantsList in Operation GetStateParticipantsList PortType Logout in Operation Logout PortType Login in Operation Login
source	<pre><wsdl:message name="MeFaultMessage"> <wsdl:part name="MeFault" element="mefhdr:ErrorExceptionDertail"/> </wsdl:message></pre>

1.11.2.1.1 element ErrorExceptionDetail

Table 1.11.11-2 Element ErrorExceptionDetail Details

diagram	
namespace	http://www.irs.gov/a2a/mef/MeFHeader.xsd
type	mefhdr:ErrorExceptionDetailType
properties	content complex
children	mefhdr:ErrorMessageTxt mefhdr:ErrorClassificationCd mefhdr:ErrorMessageCd mefhdr:HostNm
source	<pre><xs:element name="ErrorExceptionDetail" type="mefhdr:ErrorExceptionDetailType"/></pre>

1.11.2.2 MeF Exception

When faults (or errors) occur while processing a Third Party transmitter or State service request, the client process is notified via a SOAP fault response containing a ErrorExceptionDetail message within the SOAP fault detail tag. This interface is defined as a Fault with the only element being the complex type ErrorExceptionDetail as defined below.



Table 1.11.11-3 *ErrorExceptionDetail* Details

parts	ErrorExceptionDetail element mefhdr:ErrorExceptionDetail
used by	PortType SendSubmissions in Operation SendSubmissions PortType GetNewAcks in Operation GetNewAcks PortType GetAcks in Operation GetAcks PortType GetAck in Operation GetAck PortType GetNewSubmissionsStatus in Operation GetNewSubmissionsStatus PortType GetSubmissionsStatus in Operation GetSubmissionsStatus PortType GetSubmissionStatus in Operation GetSubmissionStatus PortType GetAcksByMsgID in Operation GetAcksByMsgID PortType GetNew2290Schedule1s in Operation GetNew2290Schedule1s PortType Get2290Schedule1s in Operation Get2290Schedule1s PortType Get2290Schedule1 in Operation Get2290Schedule1 PortType Get2290Schedule1sByMsgID in Operation Get2290Schedule1sByMsgID
source	<code><wsdl:message name="ErrorExceptionDetail"></code> <code><wsdl:part name="ErrorExceptionDetail" element="mefhdr: ErrorExceptionDetail"/></code> <code></wsdl:message></code>

Table 1.11.11-4 *ErrorExceptionDetail* for State Services

parts	ErrorExceptionDetail element mefhdr: ErrorExceptionDetail
used by	PortType GetNewSubmissions in Operation GetNewSubmissions PortType GetSubmissions in Operation GetSubmissions PortType GetSubmission in Operation GetSubmission PortType SendSubmissionReceipts in Operation SendSubmissionReceipts PortType SendAcks in Operation SendAcks PortType GetNewAckNotifications in Operation GetNewAckNotifications PortType GetAckNotifications in Operation GetAckNotifications PortType GetAckNotification in Operation GetAckNotification PortType GetSubmissionsByMsgID in Operation GetSubmissionsByMsgID PortType GetSubmissionReconciliationList in Operation GetSubmissionReconciliationList
source	<code><wsdl:message name="ErrorExceptionDetail"></code>



<pre><wsdl:part name="ErrorExceptionDetail" element="mefhdr: ErrorExceptionDetail"/> </wsdl:message></pre>
--

1.11.2.2.1 element ErrorExceptionDetail

Table 1.11.11-5 Element ErrorExceptionDetail Details

Diagram	
Namespace	http://www.irs.gov/a2a/mef/MeFHeader.xsd
type	mefhdr:ErrorExceptionDetailType
properties	content complex
children	mefhdr:ErrorMessageTxt mefhdr:ErrorClassificationCd mefhdr:ErrorMessageCd mefhdr:HostNm
source	<xs:element name="ErrorExceptionDetail" type="mefhdr:ErrorExceptionDetailType"/>

An example of a SOAP fault for a GetAck request with an invalid Submission ID is shown below.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<e:Envelope>
  <e:Body>
    <e:Fault>
      <faultcode>e:Server</faultcode>
      <faultstring>class ErrorExceptionDetailType: Invalid SubmissionId
      </faultstring>
      <detail>
        <wn0:ErrorExceptionDetaili:type="wn0:ErrorExceptionDetailType">
          <ErrorMessageTxt i:type="d:string">
            Failed to get acknowledgement for message id
            001232006123msg00012. Reason: Could not find
            acknowledgement for submission ID 98792345076929 for this
            ETIN 00123.
          </ErrorMessageTxt>
          <ErrorClassificationCd i:type="wn0:ErrorClassificationCdType">
            REQUEST_ERROR
          </ErrorClassificationCd>
          <ErrorMessageCdi:type="d:string">
            REQSTInnnnnnnnn</ErrorMessageCd>
          <HostNm i:type="d:string">localhost</HostNm>
        </wn0:ErrorExceptionDetaili >
      </detail>
    </e:Fault>
```



```
</e:Body>  
</e:Envelope>
```



2 Data Mapping

2.1 Data Mapping for A2A Web Services

The Web Services provided by the MeF system are described in WSDL and XML Schemas. Only the file names are presented in Table 2-1, A2A WSDL and XML Schema Files. The actual WSDL and XML Schema files are available from the IRS elsewhere.

Table 2.1-1: A2A WSDL and XML Schema Files

WSDL Files	XML Schema Files
	MeFHeader.xsd
MeFMSIServices.wsdl	MeFMSIServices.xsd
MeFStateServicesMIME.wsdl	MeFStateService.xsd
MeFStateServicesMTOM.wsdl	MeFStateServiceMTOM.xsd
MeFTransmitterServicesMIME.wsdl	MeFTransmitterService.xsd
MeFTransmitterServicesMTOM.wsdl	MeFTransmitterServiceMTOM.xsd
ETECTransmitterServicesMIME.wsdl	ETECTransmitterService.xsd
ETECTransmitterServicesMTOM.wsdl	ETECTransmitterServiceMTOM.xsd
	xmime.xsd

2.1.1 Namespaces Used for A2A Web Services

The namespaces used for the MeF A2A services follow a logical naming organization. The XML Schema target namespace is always the XML Schema file name in the IRS/MeF domain (e.g., “http://www.irs.gov/a2a/mef/MeFMSIServices.xsd”). The WSDL target namespace is the WSDL name in the IRS/MeF domain (e.g., “http://www.irs.gov/a2a/mef/MeFStateService”). The namespaces used in the WSDL and XML Schema files are presented in Table 2-2, A2A Target Namespaces. The XML namespace prefixes used are presented in Table 2-3, A2A XML Namespace Prefixes. The schemas whose name ends with “MTOM.xsd” are to be used as the default namespace and thus without a prefix, using attribute declarations of the form `xmlns="http://www.irs.gov/a2a/mef/SpecificServiceMTOM.xsd"` in the request’s top element. The “Send” requests in those schemas are subject to errors if a namespace prefix is used for their message definition.

Table 2.1-2: A2A Target Namespaces

XML Schemas	
File	targetNamespace
MeFHeader.xsd	http://www.irs.gov/a2a/mef/MeFHeader.xsd
MeFMSIServices.xsd	http://www.irs.gov/a2a/mef/MeFMSIServices.xsd
MeFStateService.xsd	http://www.irs.gov/a2a/mef/MeFStateService.xsd
MeFStateServiceMTOM.xsd	http://www.irs.gov/a2a/mef/MeFStateServiceMTOM.xsd
MeFTransmitterService.xsd	http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd
MeFTransmitterServiceMTOM.xsd	http://www.irs.gov/a2a/mef/MeFTransmitterServiceMTOM.xsd
ETECTransmitterService.xsd	http://www.irs.gov/a2a/mef/ETECTransmitterService.xsd



ETECTransmitterServiceMTOM.xsd	http://www.irs.gov/a2a/mef/ETECTransmitterServiceMTOM.xsd
WSDLs	
File	targetNamespace
MeFMSIServices.wsdl	http://www.irs.gov/a2a/mef/MeFMSIServices
MeFStateServicesMIME.wsdl	http://www.irs.gov/a2a/mef/MeFStateService
MeFStateServicesMTOM.wsdl	http://www.irs.gov/a2a/mef/MeFStateServiceMTOM
MeFTransmitterServicesMIME.wsdl	http://www.irs.gov/a2a/mef/MeFTransmitterService
MeFTransmitterServicesMTOM.wsdl	http://www.irs.gov/a2a/mef/MeFTransmitterServiceMTOM
ETECTransmitterServicesMIME.wsdl	http://www.irs.gov/a2a/mef/ETECTransmitterService
ETECTransmitterServicesMTOM.wsdl	http://www.irs.gov/a2a/mef/ETECTransmitterServiceMTOM

Table 2.1-3: A2A XML Namespace Prefixes

XML Schemas	
Prefix	Namespace
xsd	http://www.w3.org/2001/XMLSchema
xs	http://www.w3.org/2001/XMLSchema
mefhdr	http://www.irs.gov/a2a/mef/MeFHeader.xsd
mef	http://www.irs.gov/a2a/mef/MeFMSIServices.xsd
msx	http://www.irs.gov/a2a/mef/MeFStateService.xsd
ds	http://www.w3.org/2000/09/xmlsig#
wsse	http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd
xmime	http://www.w3.org/2005/05/xmlmime
WSDLs	
Prefix	Namespace
wsdl	http://schemas.xmlsoap.org/wsdl/
xsd	http://www.w3.org/2001/XMLSchema
xs	http://www.w3.org/2001/XMLSchema
soap	http://schemas.xmlsoap.org/wsdl/soap/
soapbind	http://schemas.xmlsoap.org/wsdl/soap/
mime	http://schemas.xmlsoap.org/wsdl/mime/
xmime	http://www.w3.org/2005/05/xmlmime
mefhdr	http://www.irs.gov/a2a/mef/MeFHeader.xsd
mef	http://www.irs.gov/a2a/mef/MeFMSIServices.xsd
tns	http://www.irs.gov/a2a/mef/MeFMSIServices
mef	http://www.irs.gov/a2a/mef/MeFStateService
msx	http://www.irs.gov/a2a/mef/MeFStateService.xsd
mef	http://www.irs.gov/a2a/mef/MeFTransmitterService
mtx	http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd
etec	http://www.irs.gov/a2a/mef/ETECTransmitterService



2.1.2 Data Mapping for the IFA Services

In the IFA channel the data are exchanged in files with SOAP data structures. The contents of these files are defined as attachments. Only the schema file names are presented in Table 2-4, IFA XML Schema Files. The actual XML Schema files are available from the IRS elsewhere.

Table 2.1-4: IFA XML Schema Files

XML Schema Files
BinaryAttachment.xsd
efileAttachments.xsd
efileMessageCommon.xsd
efileMessageIFA.xsd
efileTypes.xsd
SOAP.xsd

2.1.3 Namespaces Used for the IFA Services

The namespaces used for the MeF IFA services follow a logical naming organization. The XML Schema target namespace for most of the files is “efile” in the IRS domain (i.e., “http://www.irs.gov/efile”). The namespaces used in the XML Schema files are presented in Table 2-5, IFA Target Namespaces. The XML namespace prefixes used are presented in Table 2-6, IFA XML Namespace Prefixes.

Table 2.1-5: IFA Target Namespaces

XML Schemas	
File	targetNamespace
BinaryAttachment.xsd	http://www.irs.gov/efile
efileAttachments.xsd	http://www.irs.gov/efile
efileMessageCommon.xsd	http://www.irs.gov/efile
efileMessageIFA.xsd	http://www.irs.gov/efile
efileTypes.xsd	http://www.irs.gov/efile
SOAP.xsd	http://schemas.xmlsoap.org/soap/envelope/

Table 2.1-6: IFA XML Namespace Prefixes

XML Schemas	
Prefix	Namespace
xsd	http://www.w3.org/2001/XMLSchema
xs	http://www.w3.org/2001/XMLSchema
tns	http://schemas.xmlsoap.org/soap/envelope/
<none>	http://www.irs.gov/efile



3 IFA Interface Specification

3.1 Transmitter Operations

A transmitter can upload transmission files to and retrieve submission acknowledgements from the IFA. The transmission file format and submission acknowledgement format are specified in the following sections. The IFA channel supports all of the same transmitter services as the A2A channel, except for the GetAcksByMsgID service.

NOTE: The interactions between a transmitter and the IFA are defined in the *MeF Internet Filing Application (IFA) Interface Control Document*.

3.2 Transmission File Specification

A transmission file contains a SOAP message with one or more attachments, where each attachment contains an IRS or State submission. A sample IFA transmission file message is shown below:

MIME-Version: 1.0

Content-Type: multipart/related; boundary="MIMEBoundary"; type="text/xml"

Content-Description: This Transmission File is created with Pegasus Test Suite

X-eFileRoutingCode: MEF

--MIMEBoundary

Content-Type: text/xml

Content-Transfer-Encoding: 8bit

Content-Location: SoapEnvelope

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP:Envelope xmlns="http://www.irs.gov/efile"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:SOAP="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:efile="http://www.irs.gov/efile"
```



```

xsi:schemaLocation="http://schemas.xmlsoap.org/soap/envelope/
../message/SOAP.xsd http://www.irs.gov/efile ../message/efileMessage.xsd">

  <SOAP:Header>
    <TransmissionHeader>
      <TransmissionId>00044</TransmissionId>
      <Timestamp>2008-03-18T11:36:49-05:00</Timestamp>
      <Transmitter>
        <ETIN>00852</ETIN>
      </Transmitter>
    </TransmissionHeader>
  </SOAP:Header>
  <SOAP:Body>
    <TransmissionManifest>
      <SubmissionDataList>
        <Count>1</Count>
        <SubmissionData>
          <SubmissionId>00000020080778014775</SubmissionId>
          <ElectronicPostmark>2008-03-18T11:36:49-
05:00</ElectronicPostmark>
        </SubmissionData>
      </SubmissionDataList>
    </TransmissionManifest>
  </SOAP:Body>
</SOAP:Envelope>

--MIMEBoundary
Content-Type: application/octet-stream
Content-Transfer-Encoding: Binary
Content-Location: SubmissionZip

```

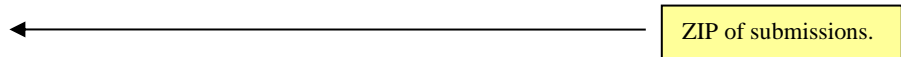


Figure 3-1: Sample Transmission File Message Submitted to IFA

Some transmitters may need to note that the MIME headers supplied here are found within the transmission file. The browser is expected to send its POST command containing the file using a content type of multipart/form-data rather than any of the three values contained within the sample file.



4 Example A2A Web Service Messages

This section presents request and response message examples for all A2A Web Services. The messages are divided into three classifications:

1. MSI Services Messages
2. State Services Messages
3. Transmitter Services Messages

NOTE: All SOAP request and response messages are identical for MIME and MTOM, except for the attachment transport mechanism.

To focus on what's different in each request and response, the examples only include the full WS-Security XML element for services:

- Login
- Logout
- EtinRetrieval
- GetNewSubmissions
- SendSubmissions

For other services, the WS-Security XML element has been omitted for clarity.

4.1 MSI Services Messages

This section presents SOAP request and response message examples for the following MSI services:

- Login
- Logout
- EtinRetrieval
- EtinStatus
- GetStateParticipantsList

4.1.1 Login Service Request Message

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope
  xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
utility-1.0.xsd"
  xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Header>
    <h:MeFHeader Id="MeFHeader" u:Id="_2"
xmlns:h="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
  xmlns="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd">
    <MessageID>123102024267zcsxjrc1</MessageID>
    <Action>Login</Action>
```



```

<MessageTs>2024-09-23T22:33:04.7848388Z</MessageTs>
<ETIN>12310</ETIN>
<SessionKeyCd>Y</SessionKeyCd>
<TestCd>T</TestCd>
<AppSysID>112320001</AppSysID>
<WSDLVersionNum>10.8</WSDLVersionNum>
<ClientSoftwareTxt>MeFA2ADNetToolkit2024v15.0</ClientSoftwareTxt>
</h:MeFHeader>
<o:Security s:mustUnderstand="1"
  xmlns:o="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd">
  <u:Timestamp
    u:Id="uuid-bf4d4ab4-d5d3-4906-a156-f9ae259c67d0-1">
    <u:Created>2024-09-23T22:33:05.114Z</u:Created>
    <u:Expires>2024-09-23T22:38:05.114Z</u:Expires>
    </u:Timestamp>
  <o:BinarySecurityToken
    u:Id="uuid-37649ac6-5b64-49b5-9f85-90f72c29816f-2"
    ValueType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-
token-profile-1.0#X509v3"
    EncodingType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
soap-message-security-1.0#Base64Binary">MIIH+CertTextInBase64
  </o:BinarySecurityToken>
  <Signature xmlns="http://www.w3.org/2000/09/xmldsig#">
    <SignedInfo>
      <CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-
c14n#" />
      <SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-
sha256" />
      <Reference URI="#_1">
        <Transforms>
          <Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
        </Transforms>
        <DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256" />
        <DigestValue>SkJtTeS3015KQ3izk/nOgH8IPc0V9m5TUz1dfujdxqU=
        </DigestValue>
      </Reference>
      <Reference URI="#_2">
        <Transforms>
          <Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
        </Transforms>
        <DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256" />
        <DigestValue>P4UIOMxAkabo69ZrdCkvstRK00BDyd93/Nj+ieJE6u0=
        </DigestValue>
      </Reference>
      <Reference URI="#uuid-bf4d4ab4-d5d3-4906-a156-f9ae259c67d0-1">
        <Transforms>
          <Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
        </Transforms>
        <DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256" />
        <DigestValue>oEPFUERKL8bhQ3bDwzV82Gm3xbK+44dFlq49MGX7EAA=
        </DigestValue>
      </Reference>
    </SignedInfo>
  </Signature>
</o:Security>

```



```

</SignedInfo>
<SignatureValue>SignatureInBase64
</SignatureValue>
<KeyInfo>
  <o:SecurityTokenReference>
    <o:Reference
      ValueType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
x509-token-profile-1.0#X509v3"
      URI="#uuid-37649ac6-5b64-49b5-9f85-90f72c29816f-2" />
    </o:SecurityTokenReference>
  </KeyInfo>
</Signature>
</o:Security>
</s:Header>
<s:Body u:Id="_1" xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd">
  <LoginRequest
  xmlns="http://www.irs.gov/a2a/mef/MeFMSIServices.xsd"></LoginRequest>
</s:Body>
</s:Envelope>

```

4.1.2 Login Service Response Message

```

<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  <soapenv:Header>
    <ns1:MeFHeader
      xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd">
      <ns1:MessageID>123102024267zcsxjrc1R</ns1:MessageID>
      <ns1:RelatesTo>123102024267zcsxjrc1</ns1:RelatesTo>
      <ns1:Action>Login</ns1:Action>
      <ns1:MessageTs>2024-09-23T22:33:04.7848388Z</ns1:MessageTs>
      <ns1:ETIN>12310</ns1:ETIN>
      <ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
      <ns1:TestCd>T</ns1:TestCd>
      <ns1:AppSysID>112320001</ns1:AppSysID>
      <ns1:WSDLVersionNum>10.8</ns1:WSDLVersionNum>
      <ns1:ClientSoftwareTxt>MeFA2ADNetToolkit2024v15.0
      </ns1:ClientSoftwareTxt>
    </ns1:MeFHeader>
    <o:Security
      xmlns:o="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd">
      <o:UsernameToken
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd"
        wsu:Id="id-d2acc8bcd0eda58da9d312d515053ec15f55f87d">

```



```

    <o:Username>112320001</o:Username>
  </o:UsernameToken>
  <saml:Assertion
    xmlns:saml="urn:oasis:names:tc:SAML:1.0:assertion"
    AssertionID="id-5eb4861e4c447f1b358600dfaa605215e72cd6b2"
    IssueInstant="2024-09-23T22:33:11.683Z"
    Issuer="http://www.forumsys.com/sentry" MajorVersion="1"
    MinorVersion="1">
    <saml:Conditions
      NotBefore="2024-09-23T22:33:11.683Z"
      NotOnOrAfter="2024-09-24T00:33:11.683Z" />
    <saml:AuthenticationStatement
      AuthenticationInstant="2024-09-23T22:33:11.683Z"
      AuthenticationMethod="urn:oasis:names:tc:SAML:1.0:am:password">
      <saml:Subject>
        <saml:NameIdentifier
          Format="urn:oasis:names:tc:SAML:1.1:nameid-
format:unspecified">UID=112320001+DNQ=300757, OU=A2A, O=US Government, C=US
        </saml:NameIdentifier>
        <saml:SubjectConfirmation>
          <saml:ConfirmationMethod>urn:oasis:names:tc:SAML:1.0:cm:bearer
          </saml:ConfirmationMethod>
        </saml:SubjectConfirmation>
      </saml:Subject>
    </saml:AuthenticationStatement>
    <saml:AttributeStatement>
      <saml:Subject>
        <saml:NameIdentifier
          Format="urn:oasis:names:tc:SAML:1.1:nameid-
format:unspecified">UID=112320001+DNQ=300757, OU=A2A, O=US Government, C=US
        </saml:NameIdentifier>
        <saml:SubjectConfirmation>
          <saml:ConfirmationMethod>urn:oasis:names:tc:SAML:1.0:cm:bearer
          </saml:ConfirmationMethod>
        </saml:SubjectConfirmation>
      </saml:Subject>
      <saml:Attribute AttributeName="SMSESSION"
AttributeNamespace="http://www.forumsys.com/sentry">
        <saml:AttributeValue>jdh00oaGPo7W0bL5ryXBpmY1RXGS
        </saml:AttributeValue>
      </saml:Attribute>
    </saml:AttributeStatement>
    <ds:Signature
      xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
      <ds:SignedInfo>
        <ds:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-
exc-c14n#" />
        <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-
more#rsa-sha256" />
        <ds:Reference URI="#id-5eb4861e4c447f1b358600dfaa605215e72cd6b2">
          <ds:Transforms>
            <ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-
signature" />

```



```

        <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
    </ds:Transforms>
    <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"
/>
    <ds:DigestValue>Z3XeR/fJzxSgbvRdec16qN0it391LgH9p7iR3zcnXs4=
    </ds:DigestValue>
    </ds:Reference>
</ds:SignedInfo>
<ds:SignatureValue>
    SignatureValueInBase64
</ds:SignatureValue>
<ds:KeyInfo>
    <ds:X509Data>
        <ds:X509Certificate>
            MIIG+CertificateInBase64
        </ds:X509Certificate>
    </ds:X509Data>
    </ds:KeyInfo>
</ds:Signature>
</saml:Assertion>
<u:Timestamp
    xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd">
    <u:Created>2024-09-23T22:33:05.114Z</u:Created>
    <u:Expires>2024-09-23T22:38:05.114Z</u:Expires>
    </u:Timestamp>
</o:Security>
</soapenv:Header>
<soapenv:Body>
    <wn1:LoginResponse
        xmlns:wn1="http://www.irs.gov/a2a/mef/MeFMSIServices.xsd">
        <wn1:StatusTxt>session counter updated</wn1:StatusTxt>
    </wn1:LoginResponse>
</soapenv:Body>
</soapenv:Envelope>

```

4.1.3 Logout Service Request Message

```

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope
    xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
utility-1.0.xsd"
    xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
    <s:Header>
        <h:MeFHeader Id="MefHeader"
            xmlns="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
            xmlns:xsd="http://www.w3.org/2001/XMLSchema"
            xmlns:h="http://www.irs.gov/a2a/mef/MeFHeader.xsd">
            <h:MessageID>123102024267lakj4ik1</h:MessageID>
            <h>Action>Logout</h>Action>

```



```

<h:MessageTs>2024-09-23T22:33:11.999104Z</h:MessageTs>
<h:ETIN>12310</h:ETIN>
<h:SessionKeyCd>N</h:SessionKeyCd>
<h:TestCd>T</h:TestCd>
<h:AppSysID>12320001</h:AppSysID>
<h:WSDLVersionNum>10.8</h:WSDLVersionNum>
<h:ClientSoftwareTxt>MeFA2ADNetToolkit2023v14.1</h:ClientSoftwareTxt>
</h:MeFHeader>
<o:Security s:mustUnderstand="1"
  xmlns:o="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-secext-1.0.xsd">
  <u:Timestamp u:Id="_0">
    <u:Created>2024-09-23T22:33:12.030Z</u:Created>
    <u:Expires>2024-09-23T22:38:12.030Z</u:Expires>
  </u:Timestamp>
  <o:UsernameToken
    u:Id="uuid-37649ac6-5b64-49b5-9f85-90f72c29816f-4">
    <o:Username>12320001</o:Username>
  </o:UsernameToken>
  <saml:Assertion xmlns:saml="urn:oasis:names:tc:SAML:1.0:assertion"
    AssertionID="id-5eb4861e4c447f1b358600dfaa605215e72cd6b2"
    IssueInstant="2024-09-23T22:33:11.683Z"
    Issuer="http://www.forumsys.com/sentry" MajorVersion="1"
MinorVersion="1">
    <saml:Conditions
      NotBefore="2024-09-23T22:33:11.683Z"
      NotOnOrAfter="2024-09-24T00:33:11.683Z" />
    <saml:AuthenticationStatement
      AuthenticationInstant="2024-09-23T22:33:11.683Z"
      AuthenticationMethod="urn:oasis:names:tc:SAML:1.0:am:password">
      <saml:Subject>
        <saml:NameIdentifier
          Format="urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified">
          UID=12320001+DNQ=300757, OU=A2A, O=US Government, C=US
        </saml:NameIdentifier>
        <saml:SubjectConfirmation>
          <saml:ConfirmationMethod>urn:oasis:names:tc:SAML:1.0:cm:bearer
          </saml:ConfirmationMethod>
        </saml:SubjectConfirmation>
      </saml:Subject>
    </saml:AuthenticationStatement>
    <saml:AttributeStatement>
      <saml:Subject>
        <saml:NameIdentifier
          Format="urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified">
          UID=12320001+DNQ=300757, OU=A2A, O=US Government, C=US
        </saml:NameIdentifier>
        <saml:SubjectConfirmation>
          <saml:ConfirmationMethod>urn:oasis:names:tc:SAML:1.0:cm:bearer
          </saml:ConfirmationMethod>
        </saml:SubjectConfirmation>
      </saml:Subject>
    </saml:AttributeStatement>
  </saml:Assertion>

```




```

    <saml:Attribute AttributeName="SMSESSION"
AttributeNamespace="http://www.forumsys.com/sentry">
    <saml:AttributeValue>jdh00oaGPo7W0bL5ryXBpmY1RXGS</saml:AttributeValue>
    </saml:Attribute>
</saml:AttributeStatement>
<ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
    <ds:SignedInfo>
    <ds:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-
exc-c14n#" />
    <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-
more#rsa-sha256" />
    <ds:Reference URI="#id-5eb4861e4c447f1b358600dfaa605215e72cd6b2">
    <ds:Transforms>
    <ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-
signature" />
    <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
    </ds:Transforms>
    <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmldsig#sha256"
/>
</ds:Reference>
</ds:SignedInfo>
<ds:SignatureValue>
SignatureInbase64
</ds:SignatureValue>
<ds:KeyInfo>
    <ds:X509Data>
    <ds:X509Certificate>
    MII+CertInBase64
    </ds:X509Certificate>
    </ds:X509Data>
    </ds:KeyInfo>
</ds:Signature>
</saml:Assertion>
</o:Security>
</s:Header>
<s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <LogoutRequest xmlns="http://www.irs.gov/a2a/mef/MeFMSIServices.xsd" />
</s:Body>
</s:Envelope>

```

4.1.4 Logout Service Response Message

```

<?xml version="1.0" encoding="utf-8"?>
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
<soapenv:Header>
    <ns1:MeFHeader
xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd">

```



```

<ns1:MessageID>1231020242671akj4ik1R</ns1:MessageID>
<ns1:RelatesTo>1231020242671akj4ik1</ns1:RelatesTo>
<ns1:Action>Logout</ns1:Action>
<ns1:MessageTs>2024-09-23T22:33:11.999104Z</ns1:MessageTs>
<ns1:ETIN>12310</ns1:ETIN>
<ns1:SessionKeyCd>N</ns1:SessionKeyCd>
<ns1:TestCd>T</ns1:TestCd>
<ns1:AppSysID>12320001</ns1:AppSysID>
<ns1:WSDLVersionNum>10.8</ns1:WSDLVersionNum>
<ns1:ClientSoftwareTxt>MeFA2ADNetToolkit2024v15.1
</ns1:ClientSoftwareTxt>
</ns1:MeFHeader>
</soapenv:Header>
<soapenv:Body>
  <wn1:LogoutResponse
    xmlns:wn1="http://www.irs.gov/a2a/mef/MeFMSIServices.xsd">
    <wn1:StatusTxt>Session counter decremented</wn1:StatusTxt>
  </wn1:LogoutResponse>
</soapenv:Body>
</soapenv:Envelope>

```

4.1.5 EtinRetrieval Service Request Message

```

<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Header>
    <ns1:MeFHeader xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd">
      <ns1:MessageID>12320202418264999738</ns1:MessageID>
      <ns1:Action>EtinRetrieval</ns1:Action>
      <ns1:MessageTs>2024-09-23T23:05:08.586Z</ns1:MessageTs>
      <ns1:ETIN>12320</ns1:ETIN>
      <ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
      <ns1:TestCd>T</ns1:TestCd>
      <ns1:AppSysID></ns1:AppSysID>
      <ns1:WSDLVersionNum>10.8</ns1:WSDLVersionNum>
      <ns1:ClientSoftwareTxt>SOATEST</ns1:ClientSoftwareTxt>
    </ns1:MeFHeader>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
      <wsu:Timestamp wsu:Id="id-6b447644120b342f6cc4cdb01314269f1063a580"
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
        <wsu:Created>2024-09-23T23:02:12Z</wsu:Created>
        <wsu:Expires>2024-09-24T01:02:12Z</wsu:Expires>
      </wsu:Timestamp>
      <wsse:UsernameToken wsu:Id="id-8c4c54e2ce94590f1fc14e621267ba75c18ccbc5"
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">

```



```

    <wsse:Username>12320002</wsse:Username>
  </wsse:UsernameToken>
  <saml:Assertion AssertionID="id-459c1324bf9c1b1e32701fd9e9bd5fb82c1fab85"
    IssueInstant="2024-09-23T23:02:12.811Z"
    Issuer="http://www.forumsys.com/sentry" MajorVersion="1"
    MinorVersion="1" xmlns:saml="urn:oasis:names:tc:SAML:1.0:assertion">
    <saml:Conditions
      NotBefore="2024-09-23T23:02:12.811Z"
      NotOnOrAfter="2024-09-24T01:02:12.811Z" />
    <saml:AuthenticationStatement
      AuthenticationInstant="2024-09-23T23:02:12.811Z"
      AuthenticationMethod="urn:oasis:names:tc:SAML:1.0:am:password">
      <saml:Subject>
        <saml:NameIdentifier Format="urn:oasis:names:tc:SAML:1.1:nameid-
format:unspecified">
          UID=12320002+DNQ=300757, OU=A2A, O=US Government, C=US
        </saml:NameIdentifier>
        <saml:SubjectConfirmation>
          <saml:ConfirmationMethod>urn:oasis:names:tc:SAML:1.0:cm:bearer</saml:Confirma
tionMethod>
          </saml:SubjectConfirmation>
        </saml:Subject>
      </saml:AuthenticationStatement>
      <saml:AttributeStatement>
        <saml:Subject>
          <saml:NameIdentifier Format="urn:oasis:names:tc:SAML:1.1:nameid-
format:unspecified">
            UID=12320002+DNQ=300757, OU=A2A, O=US Government, C=US
          </saml:NameIdentifier>
          <saml:SubjectConfirmation>
            <saml:ConfirmationMethod>urn:oasis:names:tc:SAML:1.0:cm:bearer</saml:Confirma
tionMethod>
            </saml:SubjectConfirmation>
          </saml:Subject>
          <saml:Attribute AttributeName="SMSESSION"
            AttributeNamespace="http://www.forumsys.com/sentry">
            <saml:AttributeValue>jdh0PEc6pdrvdaDRcNMxfwlJ40XO
            </saml:AttributeValue>
          </saml:Attribute>
        </saml:AttributeStatement>
      <ds:Signature
        xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
        <ds:SignedInfo>
          <ds:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-
exc-c14n#" />
          <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-
more#rsa-sha256" />
          <ds:Reference URI="#id-459c1324bf9c1b1e32701fd9e9bd5fb82c1fab85">
            <ds:Transforms>
              <ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-
signature" />

```



```

    <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
  </ds:Transforms>
  <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"
/>
  <ds:DigestValue>VegMdiGBxFjjoaEe0KlMIA3F+qlfk2pM8hNcnCOFaBLg=
  </ds:DigestValue>
  </ds:Reference>
</ds:SignedInfo>
<ds:SignatureValue>SignatureInBase64</ds:SignatureValue>
<ds:KeyInfo>
  <ds:X509Data>
    <ds:X509Certificate>MII+CertificateInBase64</ds:X509Certificate>
  </ds:X509Data>
  </ds:KeyInfo>
</ds:Signature>
</saml:Assertion>
</wsse:Security>
</SOAP-ENV:Header>
<SOAP-ENV:Body>
  <ETINRetrievalRequest
xmlns="http://www.irs.gov/a2a/mef/MeFMSIServices.xsd">
    <TestCd>T</TestCd>
  </ETINRetrievalRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

4.1.6 EtinRetrieval Service Response Message

```

<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Header>
    <ns1:MeFHeader
xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd">
      <ns1:MessageID>12320202418264999738R</ns1:MessageID>
      <ns1:RelatesTo>12320202418264999738</ns1:RelatesTo>
      <ns1:Action>EtinRetrieval</ns1:Action>
      <ns1:MessageTs>2024-09-23T23:05:08.586Z</ns1:MessageTs>
      <ns1:ETIN>12320</ns1:ETIN>
      <ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
      <ns1:TestCd>T</ns1:TestCd>
      <ns1:AppSysID />
      <ns1:WSDLVersionNum>10.8</ns1:WSDLVersionNum>
      <ns1:ClientSoftwareTxt>SOATEST</ns1:ClientSoftwareTxt>
    </ns1:MeFHeader>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-
200401-wss-wssecurity-secext-1.0.xsd">
      <wsu:Timestamp xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-
200401-wss-wssecurity-utility-1.0.xsd"
wsu:Id="id-6b447644120b342f6cc4cdb01314269f1063a580">

```



```
<wsu:Created>2024-09-23T23:02:12Z</wsu:Created>
<wsu:Expires>2024-09-24T01:02:12Z</wsu:Expires>
</wsu:Timestamp>
<wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
wsu:Id="id-8c4c54e2ce94590f1fc14e621267ba75c18ccbc5">
<wsse:Username>12320002</wsse:Username>
</wsse:UsernameToken>
<saml:Assertion xmlns:saml="urn:oasis:names:tc:SAML:1.0:assertion"
AssertionID="id-459c1324bf9c1b1e32701fd9e9bd5fb82c1fab85"
IssueInstant="2024-09-23T23:02:12.811Z"
Issuer="http://www.forumsys.com/sentry" MajorVersion="1" MinorVersion="1">
<saml:Conditions
NotBefore="2024-09-23T23:02:12.811Z"
NotOnOrAfter="2024-09-24T01:02:12.811Z" />
<saml:AuthenticationStatement
AuthenticationInstant="2024-09-23T23:02:12.811Z"
AuthenticationMethod="urn:oasis:names:tc:SAML:1.0:am:password">
<saml:Subject>
<saml:NameIdentifier Format="urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified">
UID=12320002+DNQ=300757, OU=A2A, O=US Government, C=US
</saml:NameIdentifier>
<saml:SubjectConfirmation>
<saml:ConfirmationMethod>urn:oasis:names:tc:SAML:1.0:cm:bearer
</saml:ConfirmationMethod>
</saml:SubjectConfirmation>
</saml:Subject>
</saml:AuthenticationStatement>
<saml:AttributeStatement>
<saml:Subject>
<saml:NameIdentifier Format="urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified">
UID=12320002+DNQ=300757, OU=A2A, O=US Government, C=US
</saml:NameIdentifier>
<saml:SubjectConfirmation>
<saml:ConfirmationMethod>urn:oasis:names:tc:SAML:1.0:cm:bearer
</saml:ConfirmationMethod>
</saml:SubjectConfirmation>
</saml:Subject>
<saml:Attribute AttributeName="SMSESSION"
AttributeNameSpace="http://www.forumsys.com/sentry">
<saml:AttributeValue>jdh0PEc6pdrvdaDRcNMxfwlJ40XO</saml:AttributeValue>
</saml:Attribute>
</saml:AttributeStatement>
<ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
<ds:SignedInfo>
<ds:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
<ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256" />
<ds:Reference URI="#id-459c1324bf9c1b1e32701fd9e9bd5fb82c1fab85">
<ds:Transforms>
```



```

<ds:Transform Algorithm="http://www.w3.org/2000/09/xmlsig#enveloped-
signature" />
<ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
</ds:Transforms>
<ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256" />
<ds:DigestValue>VegMdiGBxFjjoaEe0KlMIA3F+qlfk2pM8hNcnCOFaBLg=
</ds:DigestValue>
</ds:Reference>
</ds:SignedInfo>
<ds:SignatureValue>SignatureInBase64</ds:SignatureValue>
<ds:KeyInfo>
<ds:X509Data>
<ds:X509Certificate>MII+CertificateInBase64</ds:X509Certificate>
</ds:X509Data>
</ds:KeyInfo>
</ds:Signature>
</saml:Assertion>
</wsse:Security>
</soapenv:Header>
<soapenv:Body>
<wn1:ETINRetrievalResponse
xmlns:wn1="http://www.irs.gov/a2a/mef/MeFMSIServices.xsd">
<wn1:Cnt>6</wn1:Cnt>
<wn1:ETINStatusGrp>
<wn1:ETIN>10511</wn1:ETIN>
<wn1:StatusTxt>T</wn1:StatusTxt>
<wn1:FormStatusGrp>
<wn1:FormNum>2290</wn1:FormNum>
<wn1:FormStatusTxt>P</wn1:FormStatusTxt>
</wn1:FormStatusGrp>
<wn1:FormStatusGrp>
<wn1:FormNum>720</wn1:FormNum>
<wn1:FormStatusTxt>P</wn1:FormStatusTxt>
</wn1:FormStatusGrp>
<wn1:FormStatusGrp>
<wn1:FormNum>8849</wn1:FormNum>
<wn1:FormStatusTxt>P</wn1:FormStatusTxt>
</wn1:FormStatusGrp>
</wn1:ETINStatusGrp>
<wn1:ETINStatusGrp>
<wn1:ETIN>10736</wn1:ETIN>
<wn1:StatusTxt>T</wn1:StatusTxt>
<wn1:FormStatusGrp>
<wn1:FormNum>2290</wn1:FormNum>
<wn1:FormStatusTxt>P</wn1:FormStatusTxt>
</wn1:FormStatusGrp>
<wn1:FormStatusGrp>
<wn1:FormNum>720</wn1:FormNum>
<wn1:FormStatusTxt>P</wn1:FormStatusTxt>
</wn1:FormStatusGrp>
<wn1:FormStatusGrp>
<wn1:FormNum>8849</wn1:FormNum>
<wn1:FormStatusTxt>P</wn1:FormStatusTxt>

```



```

</wn1:FormStatusGrp>
</wn1:ETINStatusGrp>
<!-- Reply truncated for example -->
</wn1:ETINRetrievalResponse>
</soapenv:Body></soapenv:Envelope>

```

4.1.7 EtinStatus Service Request Message

```

<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Header>
    <ns1:MeFHeader xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
      Id="hdr">
      <ns1:MessageID>00123200603900000011</ns1:MessageID>
      <ns1:Action>EtinStatus</ns1:Action>
      <ns1:MessageTs>2006-02-08T06:24:38.290Z</ns1:MessageTs>
      <ns1:ETIN>00123</ns1:ETIN>
      <ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
      <ns1:TestCd>P</ns1:TestCd>
      <ns1:AppSysID>username</ns1:AppSysID>
      <ns1:WSDLVersionNum>10.8</ns1:WSDLVersionNum>
    </ns1:MeFHeader>
    <!-- wsse:Security omitted from example, similar to EtinRetrieval ->
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <EtinStatusRequest xmlns="http://www.irs.gov/a2a/mef/MeFMSIServices.xsd">
      <ETIN>00456</ETIN>
    </EtinStatusRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

4.1.8 EtinStatus Service Response Message

```

<?xml version="1.0" encoding="UTF-8"?>
<e:Envelope xmlns:d="http://www.w3.org/2001/XMLSchema"
  xmlns:e="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:wn0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
  xmlns:wn1="http://www.irs.gov/a2a/mef/MeFMSIServices.xsd">
  <e:Header>
    <ns0:MeFHeader xmlns:ns0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
      Id="hdr">
      <ns0:MessageID>00123200603900000011R</ns0:MessageID>
      <ns0:RelatesTo>00123200603900000011</ns0:RelatesTo>
      <ns0:Action>EtinStatus</ns0:Action>
      <ns0:MessageTs>2006-02-08T20:07:54Z</ns0:MessageTs>
      <ns0:ETIN>00123</ns0:ETIN>
      <ns0:SessionKeyCd>Y</ns0:SessionKeyCd>
      <ns0:TestCd>P</ns0:TestCd>
      <ns0:AppSysID>username</ns0:AppSysID>
    </ns0:MeFHeader>
  </e:Header>

```



```

        <ns0:WSDLVersionNum>10.8</ns0:WSDLVersionNum>
    </ns0:MeFHeader>
    <!-- wsse:Security omitted from example ->
</e:Header>
<e:Body>
    <ETINStatusResponse xmlns="http://www.irs.gov/a2a/mef/MeFMSIServices.xsd"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
        <ETIN>10736</ETIN>
        <StatusTxt>V</StatusTxt>
        <FormStatusGrp>
            <FormNum>1120</FormNum>
            <FormStatusTxt>P</FormStatusTxt>
        </FormStatusGrp>
        <FormStatusGrp>
            <FormNum>1120POL</FormNum>
            <FormStatusTxt>P</FormStatusTxt>
        </FormStatusGrp>
    </ETINStatusResponse>
</e:Body>
</e:Envelope>

```

4.1.9 GetStateParticipantsList Service Request Message

```

<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <SOAP-ENV:Header>
        <ns1:MeFHeader xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
            Id="hdr">
            <ns1:MessageID>00123200603900000013</ns1:MessageID>
            <ns1:Action>GetStateParticipantsList</ns1:Action>
            <ns1:MessageTs>2006-01-28T00:19:26.393Z</ns1:MessageTs>
            <ns1:ETIN>00123</ns1:ETIN>
            <ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
            <ns1:TestCd>P</ns1:TestCd>
            <ns1:AppSysID>username</ns1:AppSysID>
            <ns1:WSDLVersionNum>10.8</ns1:WSDLVersionNum>
        </ns1:MeFHeader>
        <!-- wsse:Security omitted from example, similar to EtinRetrieval ->
    </SOAP-ENV:Header>
    <SOAP-ENV:Body>
        <GetStateParticipantsListRequest
            xmlns="http://www.irs.gov/a2a/mef/MeFMSIServices.xsd"/>
    </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

4.1.10 GetStateParticipantsList Service Response Message

```

<?xml version="1.0" encoding="UTF-8"?>
<e:Envelope xmlns:d="http://www.w3.org/2001/XMLSchema"
    xmlns:e="http://schemas.xmlsoap.org/soap/envelope/"

```




```

xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns:wn0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
xmlns:wn1="http://www.irs.gov/a2a/mef/MeFMSIServices.xsd">
<e:Header>
  <ns0:MeFHeader xmlns:ns0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
Id="hdr">
  <ns0:MessageID>00123200603900000013R</ns0:MessageID>
  <ns0:RelatesTo>00123200603900000013</ns0:RelatesTo>
  <ns0:Action>GetStateParticipantsList</ns0:Action>
  <ns0:MessageTs>2006-02-08T20:08:21Z</ns0:MessageTs>
  <ns0:ETIN>00123</ns0:ETIN>
  <ns0:SessionKeyCd>Y</ns0:SessionKeyCd>
  <ns0:TestCd>P</ns0:TestCd>
  <ns0:AppSysID>username</ns0:AppSysID>
  <ns0:WSDLVersionNum>10.8</ns0:WSDLVersionNum>
</ns0:MeFHeader>
  <!-- wsse:Security omitted from example -->
</e:Header>
<e:Body>
  <wn1:GetStateParticipantsListResponse
  xmlns:wn1="http://www.irs.gov/a2a/mef/MeFMSIServices.xsd"
  i:type="wn1:GetStateParticipantsListResponseType">
  <wn1:Cnt i:type="d:int">6</wn1:Cnt>
  <wn1:StateParticipantGrp i:type="wn1:StateParticipantGrpType">
    <wn1:StateCd i:type="d:string">IL</wn1:StateCd>
    <wn1:ParticipantProgramsTxt i:type="d:string">ILSTCB !
P</wn1:ParticipantProgramsTxt>
  </wn1:StateParticipantGrp>
  <wn1:StateParticipantGrp i:type="wn1:StateParticipantGrpType">
    <wn1:StateCd i:type="d:string">IL</wn1:StateCd>
    <wn1:ParticipantProgramsTxt i:type="d:string">ILSTCF !
P</wn1:ParticipantProgramsTxt>
  </wn1:StateParticipantGrp>
  <wn1:StateParticipantGrp i:type="wn1:StateParticipantGrpType">
    <wn1:StateCd i:type="d:string">IL</wn1:StateCd>
    <wn1:ParticipantProgramsTxt i:type="d:string">ILSTCF !
T</wn1:ParticipantProgramsTxt>
  </wn1:StateParticipantGrp>
  <wn1:StateParticipantGrp i:type="wn1:StateParticipantGrpType">
    <wn1:StateCd i:type="d:string">KS</wn1:StateCd>
    <wn1:ParticipantProgramsTxt i:type="d:string">KSSTCF !
P</wn1:ParticipantProgramsTxt>
  </wn1:StateParticipantGrp>
  <wn1:StateParticipantGrp i:type="wn1:StateParticipantGrpType">
    <wn1:StateCd i:type="d:string">KS</wn1:StateCd>
    <wn1:ParticipantProgramsTxt i:type="d:string">KSSTCB !
P</wn1:ParticipantProgramsTxt>
  </wn1:StateParticipantGrp>
  <wn1:StateParticipantGrp i:type="wn1:StateParticipantGrpType">
    <wn1:StateCd i:type="d:string">KS</wn1:StateCd>
    <wn1:ParticipantProgramsTxt i:type="d:string">KSSTCB !
T</wn1:ParticipantProgramsTxt>
  </wn1:StateParticipantGrp>
  </wn1:GetStateParticipantsListResponse>
</e:Body>
</e:Envelope>

```



4.2 State Services SOAP Messages

This section presents SOAP request and response message examples for the following State services:

- **GetNewSubmissions**
- **GetSubmissionsByMsgID**
- **GetSubmissions**
- **GetSubmission**
- **SendSubmissionReceipts**
- **SendAcks**
- **GetNewAckNotifications**
- **GetAckNotifications**
- **GetAckNotification**
- **GetSubmissionReconciliationList**

NOTE: All of the SOAP request and response messages are identical for MIME and MTOM, except for the attachment transport mechanism.

4.2.1 GetNewSubmissions Service Request Message

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Header>
    <ns1:MeFHeader
      xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd">
      <ns1:MessageID>1233020152188714434y</ns1:MessageID>
      <ns1:Action>GetNewSubmissions</ns1:Action>
      <ns1:MessageTs>2024-09-23T23:21:24.701Z</ns1:MessageTs>
      <ns1:ETIN>12330</ns1:ETIN>
      <ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
      <ns1:TestCd>P</ns1:TestCd>
      <ns1:AppSysID>12320003</ns1:AppSysID>
      <ns1:WSDLVersionNum>10.8</ns1:WSDLVersionNum>
      <ns1:ClientSoftwareTxt>SOATest</ns1:ClientSoftwareTxt>
    </ns1:MeFHeader>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
      <wsu:Timestamp wsu:Id="id-cc8902df6def4d129fe659d7a8f9e4b6b9013173"
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
        <wsu:Created>2024-09-23T23:20:30Z</wsu:Created>
        <wsu:Expires>2024-09-24T01:20:30Z</wsu:Expires>
      </wsu:Timestamp>
      <wsse:UsernameToken wsu:Id="id-592f13fd0614f4d7fdd0d039bb6197934c0bf519"
        xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
        <wsse:Username>12320003</wsse:Username>
      </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <ns1:NewSubmissionsRequest>
      <ns1:AppSysID>12320003</ns1:AppSysID>
      <ns1:ClientSoftwareTxt>SOATest</ns1:ClientSoftwareTxt>
      <ns1:ETIN>12330</ns1:ETIN>
      <ns1:MessageID>1233020152188714434y</ns1:MessageID>
      <ns1:MessageTs>2024-09-23T23:21:24.701Z</ns1:MessageTs>
      <ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
      <ns1:TestCd>P</ns1:TestCd>
      <ns1:WSDLVersionNum>10.8</ns1:WSDLVersionNum>
      <ns1:YieldCode>Y</ns1:YieldCode>
    </ns1:NewSubmissionsRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```



```

</wsse:UsernameToken>
<saml:Assertion AssertionID="id-4038a5ce66f374007c24669f1f374004d3b9fd09"
  IssueInstant="2024-09-23T23:20:30.055Z"
  Issuer="http://www.forumsys.com/sentry" MajorVersion="1"
  MinorVersion="1" xmlns:saml="urn:oasis:names:tc:SAML:1.0:assertion">
  <saml:Conditions
    NotBefore="2024-09-23T23:20:30.055Z"
    NotOnOrAfter="2024-09-24T01:20:30.055Z" />
  <saml:AuthenticationStatement
    AuthenticationInstant="2024-09-23T23:20:30.055Z"
    AuthenticationMethod="urn:oasis:names:tc:SAML:1.0:am:password">
    <saml:Subject>
      <saml:NameIdentifier Format="urn:oasis:names:tc:SAML:1.1:nameid-
format:unspecified">
        UID=12320003+DNQ=301159, OU=A2A, O=US Government, C=US
      </saml:NameIdentifier>
      <saml:SubjectConfirmation>
        <saml:ConfirmationMethod>urn:oasis:names:tc:SAML:1.0:cm:bearer
      </saml:ConfirmationMethod>
      </saml:SubjectConfirmation>
    </saml:Subject>
  </saml:AuthenticationStatement>
  <saml:AttributeStatement>
    <saml:Subject>
      <saml:NameIdentifier Format="urn:oasis:names:tc:SAML:1.1:nameid-
format:unspecified">
        UID=12320003+DNQ=301159, OU=A2A, O=US Government, C=US
      </saml:NameIdentifier>
      <saml:SubjectConfirmation>
        <saml:ConfirmationMethod>urn:oasis:names:tc:SAML:1.0:cm:bearer
      </saml:ConfirmationMethod>
      </saml:SubjectConfirmation>
    </saml:Subject>
    <saml:Attribute AttributeName="SMSESSION"
      AttributeNamespace="http://www.forumsys.com/sentry">
      <saml:AttributeValue>jdh0CMNawStY6D2qQ00qQLD1pkzF
      </saml:AttributeValue>
    </saml:Attribute>
  </saml:AttributeStatement>
  <ds:Signature
    xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
    <ds:SignedInfo>
      <ds:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-
exc-c14n#" />
      <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-
more#rsa-sha256" />
      <ds:Reference URI="#id-4038a5ce66f374007c24669f1f374004d3b9fd09">
      <ds:Transforms>
        <ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-
signature" />
        <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
      </ds:Transforms>
    </ds:Reference>
  </ds:SignedInfo>
  <ds:SignatureValue>
  </ds:SignatureValue>
  </ds:Signature>
</saml:Assertion>

```



```

    <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"
/>
    <ds:DigestValue>K02LR9JqjtnYyn1xliaczy4qFHUaXlfrK9nmj46+vCg=
    </ds:DigestValue>
    </ds:Reference>
  </ds:SignedInfo>
  <ds:SignatureValue>SignatureInBase64</ds:SignatureValue>
  <ds:KeyInfo>
    <ds:X509Data>
      <ds:X509Certificate>MII+CertInBase64</ds:X509Certificate>
    </ds:X509Data>
  </ds:KeyInfo>
</ds:Signature>
</saml:Assertion>
</wsse:Security>
</SOAP-ENV:Header>
<SOAP-ENV:Body>
  <GetNewSubmissionsRequestArgs
    xmlns="http://www.irs.gov/a2a/mef/MeFStateService.xsd">
    <SubsetStSubmissionCategoryCd>all</SubsetStSubmissionCategoryCd>
  </GetNewSubmissionsRequestArgs>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

4.2.2 GetNewSubmissions Service Response Message

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  <soapenv:Header>
    <ns1:MeFHeader
      xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd">
      <ns1:MessageID>1233020152188714434yR</ns1:MessageID>
      <ns1:RelatesTo>1233020152188714434y</ns1:RelatesTo>
      <ns1:Action>GetNewSubmissions</ns1:Action>
      <ns1:MessageTs>2024-09-23T23:21:24.701Z</ns1:MessageTs>
      <ns1:ETIN>12330</ns1:ETIN>
      <ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
      <ns1:TestCd>P</ns1:TestCd>
      <ns1:AppSysID>12320003</ns1:AppSysID>
      <ns1:WSDLVersionNum>10.8</ns1:WSDLVersionNum>
      <ns1:ClientSoftwareTxt>SOATest</ns1:ClientSoftwareTxt>
    </ns1:MeFHeader>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-
200401-wss-wssecurity-secext-1.0.xsd">
      <wsu:Timestamp xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-
200401-wss-wssecurity-utility-1.0.xsd"
        wsu:Id="id-cc8902df6def4d129fe659d7a8f9e4b6b9013173">
          <wsu:Created>2024-09-23T23:20:30Z</wsu:Created>
          <wsu:Expires>2024-09-24T01:20:30Z</wsu:Expires>
        </wsu:Timestamp>
      <wsse:UsernameToken

```



```

    xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd"
    wsu:Id="id-592f13fd0614f4d7fdd0d039bb6197934c0bf519">
    <wsse:Username>12320003</wsse:Username>
</wsse:UsernameToken>
<saml:Assertion
    xmlns:saml="urn:oasis:names:tc:SAML:1.0:assertion"
    AssertionID="id-4038a5ce66f374007c24669f1f374004d3b9fd09"
    IssueInstant="2024-09-23T23:20:30.055Z"
    Issuer="http://www.forumsys.com/sentry" MajorVersion="1"
MinorVersion="1">
    <saml:Conditions
    NotBefore="2024-09-23T23:20:30.055Z"
    NotOnOrAfter="2024-09-24T01:20:30.055Z" />
<saml:AuthenticationStatement
    AuthenticationInstant="2024-09-23T23:20:30.055Z"
    AuthenticationMethod="urn:oasis:names:tc:SAML:1.0:am:password">
    <saml:Subject>
    <saml:NameIdentifier Format="urn:oasis:names:tc:SAML:1.1:nameid-
format:unspecified">
    UID=12320003+DNQ=301159, OU=A2A, O=US Government, C=US
    </saml:NameIdentifier>
    <saml:SubjectConfirmation>
    <saml:ConfirmationMethod>urn:oasis:names:tc:SAML:1.0:cm:bearer
    </saml:ConfirmationMethod>
    </saml:SubjectConfirmation>
    </saml:Subject>
</saml:AuthenticationStatement>
<saml:AttributeStatement>
    <saml:Subject>
    <saml:NameIdentifier
    Format="urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified">
    UID=12320003+DNQ=301159, OU=A2A, O=US Government, C=US
    </saml:NameIdentifier>
    <saml:SubjectConfirmation>
    <saml:ConfirmationMethod>urn:oasis:names:tc:SAML:1.0:cm:bearer
    </saml:ConfirmationMethod>
    </saml:SubjectConfirmation>
    </saml:Subject>
    <saml:Attribute AttributeName="SMSESSION"
    AttributeNamespace="http://www.forumsys.com/sentry">
    <saml:AttributeValue>jdh0CMNawStY6D2qQ00qQLD1pkzF
    </saml:AttributeValue>
    </saml:Attribute>
</saml:AttributeStatement>
<ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
    <ds:SignedInfo>
    <ds:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-
exc-c14n#" />
    <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-
more#rsa-sha256" />
    <ds:Reference URI="#id-4038a5ce66f374007c24669f1f374004d3b9fd09">
    <ds:Transforms>

```



```

        <ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-
signature" />
        <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
        </ds:Transforms>
        <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"
/>
        <ds:DigestValue>K02LR9JqjtnYyn1xliaczy4qFHUaXlfrK9nmj46+vCg=
        </ds:DigestValue>
        </ds:Reference>
    </ds:SignedInfo>
    <ds:SignatureValue>SignatureInBase64</ds:SignatureValue>
    <ds:KeyInfo>
        <ds:X509Data>
            <ds:X509Certificate>MIIG+CertificateInBase64</ds:X509Certificate>
        </ds:X509Data>
    </ds:KeyInfo>
    </ds:Signature>
</saml:Assertion>
</wsse:Security>
</soapenv:Header>
<soapenv:Body>
    <c:GetNewSubmissionsResponse
xmlns:c="http://www.irs.gov/a2a/mef/MeFStateService.xsd"
xmlns:a="http://www.irs.gov/a2a/mef/TransactionContext.xsd"
xmlns:b="http://www.irs.gov/a2a/mef/MeFHeader.xsd">
        <c:MoreAvailableInd>true</c:MoreAvailableInd>
        <c:IRSDataForStateSubmissionList>
            <c:Cnt>1</c:Cnt>
            <c:IRSDataForStateSubmission>
                <c:SubmissionId>1234502023265a167rzp</c:SubmissionId>
                <c:ETIN>12340</c:ETIN>
                <c:ReceivedAtIRSTs>2024-01-03T06:32:16-05:00</c:ReceivedAtIRSTs>
                <c:SSNValidationStatus>
                    <c:PrimarySSNValidationStatusCd>Found Match
                    </c:PrimarySSNValidationStatusCd>
                    <c:SecondarySSNValidationStatusCd>Found Match
                    </c:SecondarySSNValidationStatusCd>
                </c:SSNValidationStatus>
                <c:ElectronicPostmarkTs>2024-01-03T06:32:16.460-05:00
                </c:ElectronicPostmarkTs>
                <c:SourceIRSSubmissionId>1234502023265a167rzo
                </c:SourceIRSSubmissionId>
                <c:ITINMismatchCd>false</c:ITINMismatchCd>
                <c:ImperfectReturnInd>false</c:ImperfectReturnInd>
                <c:TransmitterIPAddressTxt>10.204.72.31</c:TransmitterIPAddressTxt>
                <c:TransmitterTs>2024-01-03T06:32:16-05:00</c:TransmitterTs>
            </c:IRSDataForStateSubmission>
        </c:IRSDataForStateSubmissionList>
    </c:GetNewSubmissionsResponse>
</soapenv:Body>
</soapenv:Envelope>

```



4.2.3 Get Submissions by Message ID Service Request Message

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <SOAP-ENV:Header>
    <ns1:MeFHeader
      xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd">
      <ns1:MessageID>5432120201142345648p</ns1:MessageID>
      <ns1:Action>GetSubmissionsByMsgID</ns1:Action>
      <ns1:MessageTs>2020-04-23T15:51:31.107Z</ns1:MessageTs>
      <ns1:ETIN>54321</ns1:ETIN>
      <ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
      <ns1:TestCd>P</ns1:TestCd>
      <ns1:AppSysID>12345678</ns1:AppSysID>
      <ns1:WSDLVersionNum>10.3</ns1:WSDLVersionNum>
      <ns1:ClientSoftwareTxt>SampleDoc</ns1:ClientSoftwareTxt>
    </ns1:MeFHeader>
    <!-- wsse:Security omitted in examples, similar to GetNewSubmissions -->
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <GetSubmissionsByMsgIDRequestArgs
      xmlns="http://www.irs.gov/a2a/mef/MeFStateService.xsd">
      <MessageId>5432120200550912234j</MessageId>
    </GetSubmissionsByMsgIDRequestArgs>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

4.2.4 Get Submissions by Message ID Service Response Message

```
<?xml version='1.0' encoding='utf-8'?>
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Header>
    <ns1:MeFHeader
      xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd">
      <ns1:MessageID>5432120201142347248pR</ns1:MessageID>
      <ns1:RelatesTo>5432120201142347248p</ns1:RelatesTo>
      <ns1:Action>GetSubmissionsByMsgID
    </ns1:Action>
      <ns1:MessageTs>2020-04-23T15:51:31.107Z</ns1:MessageTs>
      <ns1:ETIN>54321</ns1:ETIN>
      <ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
      <ns1:TestCd>P</ns1:TestCd>
      <ns1:AppSysID>12345678</ns1:AppSysID>
      <ns1:WSDLVersionNum>10.3</ns1:WSDLVersionNum>
      <ns1:ClientSoftwareTxt>SampleDoc</ns1:ClientSoftwareTxt>
    </ns1:MeFHeader>
```



```

    <!-- wsse:Security omitted in examples -->
</soapenv:Header>
<soapenv:Body>
  <ns1:GetSubmissionsByMsgIDResponse
    xmlns:ns1="http://www.irs.gov/a2a/mef/MeFStateService.xsd">
    <ns1:MoreAvailableInd>false</ns1:MoreAvailableInd>
    <ns1:IRSDataForStateSubmissionList>
      <ns1:Cnt>1</ns1:Cnt>
      <ns1:IRSDataForStateSubmission>
        <ns1:SubmissionId>1234562020114a091209</ns1:SubmissionId>
        <ns1:ETIN>10736</ns1:ETIN>
        <ns1:ReceivedAtIRSTs>2020-04-23T10:31:45.000-04:00
        </ns1:ReceivedAtIRSTs>
        <ns1:FederaleEINStatusCd>Found Match</ns1:FederaleEINStatusCd>
        <ns1:ElectronicPostmarkTs>2020-04-23T14:31:40.427+00:00
        </ns1:ElectronicPostmarkTs>
        <ns1:SourceIRSSubmissionId> 1234562020114a091208
        </ns1:SourceIRSSubmissionId>
        <!-- Note: The last 4 elements are optional and may or may not be present
          in the actual response -->
        <ns1:ITINMismatchCd>false</ns1:ITINMismatchCd>
        <ns1:ImperfectReturnInd>true</ns1:ImperfectReturnInd>
        <ns1:TransmitterIPAddressTxt>192.168.6.89
        </ns1:TransmitterIPAddressTxt>
        <ns1:TransmitterTs>2020-01-17T09:30:47.0Z</ns1:TransmitterTs>
      </ns1:IRSDataForStateSubmission>
    </ns1:IRSDataForStateSubmissionList>
  </ns1:GetSubmissionsByMsgIDResponse>
</soapenv:Body>
</soapenv:Envelope>

```

4.2.5 Get Submissions Service Request Message

```

<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Header>
    <ns1:MeFHeader
      xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd">
      <ns1:MessageID>5432120201145678756p</ns1:MessageID>
      <ns1:Action>GetSubmissions</ns1:Action>
      <ns1:MessageTs>2020-04-23T15:49:55.980Z</ns1:MessageTs>
      <ns1:ETIN>54321</ns1:ETIN>
      <ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
      <ns1:TestCd>P</ns1:TestCd>
      <ns1:AppSysID>12345678</ns1:AppSysID>
      <ns1:WSDLVersionNum>10.3</ns1:WSDLVersionNum>
      <ns1:ClientSoftwareTxt>SampleDoc</ns1:ClientSoftwareTxt>
    </ns1:MeFHeader>
    <!-- wsse:Security omitted in examples, similar to GetNewSubmissions -->

```




```

</SOAP-ENV:Header>
<SOAP-ENV:Body>
  <GetSubmissionsRequestArgs
    xmlns="http://www.irs.gov/a2a/mef/MeFStateService.xsd">
    <SubmissionIdList>
      <SubmissionId>1234562020112a091207</SubmissionId>
      <SubmissionId>1234562020112a091209</SubmissionId>
    </SubmissionIdList>
  </GetSubmissionsRequestArgs>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

4.2.6 GetSubmissions Service Response Message

```

<?xml version='1.0' encoding='utf-8'?>
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Header>
    <ns1:MeFHeader
      xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd">
      <ns1:MessageID>5432120201144871756pR</ns1:MessageID>
      <ns1:RelatesTo>5432120201144871756p</ns1:RelatesTo>
      <ns1:Action>GetSubmissions</ns1:Action>
      <ns1:MessageTs>2020-04-23T15:49:55.980Z</ns1:MessageTs>
      <ns1:ETIN>54321</ns1:ETIN>
      <ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
      <ns1:TestCd>P</ns1:TestCd>
      <ns1:AppSysID>12345678</ns1:AppSysID>
      <ns1:WSDLVersionNum>10.3</ns1:WSDLVersionNum>
      <ns1:ClientSoftwareTxt>SampleDoc</ns1:ClientSoftwareTxt>
    </ns1:MeFHeader> <!-- wsse:Security omitted in examples, similar to
GetNewSubmissions -->
  </soapenv:Header>
  <soapenv:Body>
    <ns1:GetSubmissionsResponse
      xmlns:ns1="http://www.irs.gov/a2a/mef/MeFStateService.xsd">
      <ns1:IRSDataForStateSubmissionList>
        <ns1:Cnt>1</ns1:Cnt>
        <ns1:IRSDataForStateSubmission>
          <ns1:SubmissionId>1234562020114a091209 </ns1:SubmissionId>
          <ns1:ETIN>10736</ns1:ETIN>
          <ns1:ReceivedAtIRSTs>2020-04-23T10:31:45.000-04:00
          </ns1:ReceivedAtIRSTs>
          <ns1:FederaleEINStatusCd>Found Match</ns1:FederaleEINStatusCd>
          <ns1:ElectronicPostmarkTs>2020-04-23T14:31:40.427+00:00
          </ns1:ElectronicPostmarkTs>
          <ns1:SourceIRSSubmissionId>1234562020114a091208
          </ns1:SourceIRSSubmissionId>
          <!-- Note: The last 4 elements are optional and may or may not be present
in the actual response -->
          <ns1:ITINMismatchCd>false</ns1:ITINMismatchCd>
          <ns1:ImperfectReturnInd>true</ns1:ImperfectReturnInd>

```



```

    <ns1:TransmitterIPAddressTxt>192.168.6.89
  </ns1:TransmitterIPAddressTxt>
  <ns1:TransmitterTs>2020-01-17T09:30:47.0Z</ns1:TransmitterTs>
</ns1:IRSDataForStateSubmission>
</ns1:IRSDataForStateSubmissionList>
<ns1:SubmissionErrorList>
  <ns1:Cnt>1</ns1:Cnt>
  <ns1:SubmissionError>
    <ns1:SubmissionId>
      1234562020114a091207
    </ns1:SubmissionId>
    <ns1:ErrorClassificationDetail>
      <ns1:ErrorMessageTxt>Backend system error</ns1:ErrorMessageTxt>
      <ns1:ErrorClassificationCd>SYSTEM_ERROR
    </ns1:ErrorClassificationCd>
      <ns1:ErrorMessageCd>SYSTMIO07500002</ns1:ErrorMessageCd>
    </ns1:ErrorClassificationDetail>
  </ns1:SubmissionError>
  <!-- Note: The SubmissionError element is optional and may or may not
    be present in the actual response -->
</ns1:SubmissionErrorList>
</ns1:GetSubmissionsResponse>
</soapenv:Body>
</soapenv:Envelope>

```

4.2.7 GetSubmission Service Request Message

```

<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Header>
    <ns1:MeFHeader xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
      Id="hdr">
      <ns1:MessageID>00123200603900000005</ns1:MessageID>
      <ns1:Action>GetSubmission</ns1:Action>
      <ns1:MessageTs>2006-01-29T23:12:33.073Z</ns1:MessageTs>
      <ns1:ETIN>00123</ns1:ETIN>
      <ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
      <ns1:TestCd>P</ns1:TestCd>
      <ns1:AppSysID>username</ns1:AppSysID>
      <ns1:WSDLVersionNum>10.8</ns1:WSDLVersionNum>
    </ns1:MeFHeader>
    <!-- wsse:Security omitted in examples, similar to GetNewSubmissions -->
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <GetSubmissionRequestArgs
      xmlns="http://www.irs.gov/a2a/mef/MeFStateService.xsd">
      <SubmissionId>0600082013261007926f</SubmissionId>
    </GetSubmissionRequestArgs>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```



4.2.8 GetSubmission Service Response Message

```
<?xml version="1.0" encoding="UTF-8"?>
<e:Envelope xmlns:wn1="http://www.irs.gov/a2a/mef/MeFStateService.xsd"
  xmlns:wn0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:e="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:d="http://www.w3.org/2001/XMLSchema">
  <e:Header>
    <ns0:MeFHeader xmlns:ns0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
      Id="hdr">
      <ns0:MessageID>00123200603900000005R</ns0:MessageID>
      <ns0:RelatesTo>00123200603900000005</ns0:RelatesTo>
      <ns0:Action>GetSubmission</ns0:Action>
      <ns0:MessageTs>2006-02-08T22:44:05Z</ns0:MessageTs>
      <ns0:ETIN>00123</ns0:ETIN>
      <ns0:SessionKeyCd>Y</ns0:SessionKeyCd>
      <ns0:TestCd>P</ns0:TestCd>
      <ns0:AppSysID>username</ns0:AppSysID>
      <ns0:WSDLVersionNum>10.8</ns0:WSDLVersionNum>
    </ns0:MeFHeader>
    <!-- wsse:Security omitted in examples, similar to GetNewSubmissions -->
  </e:Header>
  <e:Body>
    <ns10:GetSubmissionResponse
      xmlns:ns10="http://www.irs.gov/a2a/mef/MeFStateService.xsd">
      <ns10:IRSDataForStateSubmission>
        <ns10:SubmissionId>0600082013261007926f</ns10:SubmissionId>
        <ns10:ETIN>11022</ns10:ETIN>
        <ns10:ReceivedAtIRSTs>2013-09-18T11:22:01-
04:00</ns10:ReceivedAtIRSTs>
        <ns10:SSNValidationStatus>
          <ns10:PrimarySSNValidationStatusCd>Found Match
          </ns10:PrimarySSNValidationStatusCd>
          <ns10:SecondarySSNValidationStatusCd>Found Match
          </ns10:SecondarySSNValidationStatusCd>
        </ns10:SSNValidationStatus>
        <ns10:SourceIRSSubmissionId>UNKNOWN</ns10:SourceIRSSubmissionId>

        <ns10:TransmitterIPAddressTxt>12.34.56.78</ns10:TransmitterIPAddressTxt >
        <ns10:TransmitterTs>2013-09-18T15:22:00-04:00</ns10:TransmitterTs>
      </ns10:IRSDataForStateSubmission>
    </ns10:GetSubmissionResponse>
  </e:Body>
</e:Envelope>
```

4.2.9 SendSubmissionReceipts Service Request Message

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Header>
```



```

    <ns1:MeFHeader xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
Id="hdr">
    <ns1:MessageID>00123200603900000006</ns1:MessageID>
    <ns1:Action>SendSubmissionReceipts</ns1:Action>
    <ns1:MessageTs>2006-01-22T21:26:49.912Z</ns1:MessageTs>
    <ns1:ETIN>00123</ns1:ETIN>
    <ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
    <ns1:TestCd>P</ns1:TestCd>
    <ns1:AppSysID>username</ns1:AppSysID>
    <ns1:WSDLVersionNum>10.8</ns1:WSDLVersionNum>
  </ns1:MeFHeader>
  <!-- wsse:Security omitted from example -->
</SOAP-ENV:Header>
<SOAP-ENV:Body>
  <SendSubmissionReceiptsRequestArgs
xmlns="http://www.irs.gov/a2a/mef/MeFStateService.xsd"/>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

4.2.10 SendSubmissionReceipts Service Response Message

```

<?xml version="1.0" encoding="UTF-8"?>
<e:Envelope xmlns:d="http://www.w3.org/2001/XMLSchema"
  xmlns:e="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:wn0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
  xmlns:wn1="http://www.irs.gov/a2a/mef/MeFStateService.xsd">
  <e:Header>
    <ns0:MeFHeader xmlns:ns0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
Id="hdr">
      <ns0:MessageID>00123200603900000006R</ns0:MessageID>
      <ns0:RelatesTo>00123200603900000006</ns0:RelatesTo>
      <ns0:Action>SendSubmissionReceipts</ns0:Action>
      <ns0:MessageTs>2006-02-08T22:44:46Z</ns0:MessageTs>
      <ns0:ETIN>00123</ns0:ETIN>
      <ns0:SessionKeyCd>Y</ns0:SessionKeyCd>
      <ns0:TestCd>P</ns0:TestCd>
      <ns0:AppSysID>username</ns0:AppSysID>
      <ns0:WSDLVersionNum>10.8</ns0:WSDLVersionNum>
    </ns0:MeFHeader>
    <!-- wsse:Security omitted from example -->
  </e:Header>
  <e:Body>
    <ns9:SendSubmissionReceiptsResponse
xmlns:ns9="http://www.irs.gov/a2a/mef/MeFStateService.xsd">
      <ns9:Cnt>2</ns9:Cnt>
    </ns9:SendSubmissionReceiptsResponse>
  </e:Body>
</e:Envelope>

```



4.2.11 SendAcks Service Request Message

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Header>
    <ns1:MeFHeader xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
      Id="hdr">
      <ns1:MessageID>00123200603900000007</ns1:MessageID>
      <ns1:Action>SendAcks</ns1:Action>
      <ns1:MessageTs>2006-02-16T15:13:56.390Z</ns1:MessageTs>
      <ns1:ETIN>00123</ns1:ETIN>
      <ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
      <ns1:TestCd>P</ns1:TestCd>
      <ns1:AppSysID>username</ns1:AppSysID>
      <ns1:WSDLVersionNum>10.8</ns1:WSDLVersionNum>
    </ns1:MeFHeader>
    <!-- wsse:Security omitted from example, similar to GetNewSubmissions-->
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <SendAcksRequestArgs
      xmlns="http://www.irs.gov/a2a/mef/MeFStateService.xsd"/>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

4.2.12 SendAcks Service Response Message

```
<?xml version="1.0" encoding="UTF-8"?>
<e:Envelope xmlns:d="http://www.w3.org/2001/XMLSchema"
  xmlns:e="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:wn0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
  xmlns:wn1="http://www.irs.gov/a2a/mef/MeFStateService.xsd">
  <e:Header>
    <ns0:MeFHeader xmlns:ns0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
      Id="hdr">
      <ns0:MessageID>00123200603900000007R</ns0:MessageID>
      <ns0:RelatesTo>00123200603900000007</ns0:RelatesTo>
      <ns0:Action>SendAcks</ns0:Action>
      <ns0:MessageTs>2006-02-08T22:44:58Z</ns0:MessageTs>
      <ns0:ETIN>00123</ns0:ETIN>
      <ns0:SessionKeyCd>Y</ns0:SessionKeyCd>
      <ns0:TestCd>P</ns0:TestCd>
      <ns0:AppSysID>username</ns0:AppSysID>
      <ns0:WSDLVersionNum>10.8</ns0:WSDLVersionNum>
    </ns0:MeFHeader>
    <!-- wsse:Security omitted from example, similar to GetNewSubmissions-->
  </e:Header>
  <e:Body>
    <ns9:SendAcksResponse
      xmlns:ns9="http://www.irs.gov/a2a/mef/MeFStateService.xsd">
      <ns9:SubmissionErrorList>
```



```

        <ns9:Cnt>0</ns9:Cnt>
      </ns9:SubmissionErrorList>
    </ns9:SendAcksResponse>
  </e:Body>
</e:Envelope>

```

4.2.13 GetNewAckNotifications Service Request Message

```

<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Header>
    <ns1:MeFHeader xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
      Id="hdr">
      <ns1:MessageID>00123200603900000041</ns1:MessageID>
      <ns1:Action>GetNewAckNotifications</ns1:Action>
      <ns1:MessageTs>2006-02-18T13:53:51.288Z</ns1:MessageTs>
      <ns1:ETIN>00123</ns1:ETIN>
      <ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
      <ns1:TestCd>P</ns1:TestCd>
      <ns1:AppSysID>username</ns1:AppSysID>
      <ns1:WSDLVersionNum>10.8</ns1:WSDLVersionNum>
    </ns1:MeFHeader>
    <!-- wsse:Security omitted from example, similar to GetNewSubmissions-->
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <GetNewAckNotificationsRequestArgs
      xmlns="http://www.irs.gov/a2a/mef/MeFStateService.xsd">
      <MaxResultCnt>1</MaxResultCnt>
    </GetNewAckNotificationsRequestArgs>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

4.2.14 GetNewAckNotifications Service Response Message

```

<?xml version="1.0" encoding="UTF-8"?>
<e:Envelope xmlns:wn1="http://www.irs.gov/a2a/mef/MeFStateService.xsd"
  xmlns:wn0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:e="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:d="http://www.w3.org/2001/XMLSchema">
  <e:Header>
    <ns0:MeFHeader xmlns:ns0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
      Id="hdr">
      <ns0:MessageID>00123200603900000041R</ns0:MessageID>
      <ns0:RelatesTo>00123200603900000041</ns0:RelatesTo>
      <ns0:Action>GetNewAckNotifications</ns0:Action>
      <ns0:MessageTs>2006-02-08T22:56:48Z</ns0:MessageTs>
      <ns0:ETIN>00123</ns0:ETIN>
      <ns0:SessionKeyCd>Y</ns0:SessionKeyCd>
      <ns0:TestCd>P</ns0:TestCd>
      <ns0:AppSysID>username</ns0:AppSysID>
    </ns0:MeFHeader>
  </e:Header>
  <e:Body>
    <GetNewAckNotificationsResponseArgs
      xmlns="http://www.irs.gov/a2a/mef/MeFStateService.xsd">
      <MaxResultCnt>1</MaxResultCnt>
      <Result>
        <ns0:MessageID>00123200603900000041R</ns0:MessageID>
        <ns0:RelatesTo>00123200603900000041</ns0:RelatesTo>
        <ns0:Action>GetNewAckNotifications</ns0:Action>
        <ns0:MessageTs>2006-02-08T22:56:48Z</ns0:MessageTs>
        <ns0:ETIN>00123</ns0:ETIN>
        <ns0:SessionKeyCd>Y</ns0:SessionKeyCd>
        <ns0:TestCd>P</ns0:TestCd>
        <ns0:AppSysID>username</ns0:AppSysID>
      </ns0:MessageID>
    </GetNewAckNotificationsResponseArgs>
  </e:Body>
</e:Envelope>

```



```

        <ns0:WSDLVersionNum>10.8</ns0:WSDLVersionNum>
    </ns0:MeFHeader>
    <!-- wsse:Security omitted from example, similar to GetNewSubmissions-->
</e:Header>
<e:Body>
    <ns10:GetNewAckNotificationsResponse
        xmlns:ns10="http://www.irs.gov/a2a/mef/MeFStateService.xsd">
        <ns10:MoreAvailableInd>true</ns10:MoreAvailableInd>
    </ns10:GetNewAckNotificationsResponse>
</e:Body>
</e:Envelope>

```

4.2.15 GetAckNotifications Service Request Message

```

<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <SOAP-ENV:Header>
        <ns1:MeFHeader xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
            Id="hdr">
            <ns1:MessageID>00123200603900000042</ns1:MessageID>
            <ns1:Action>GetAckNotifications</ns1:Action>
            <ns1:MessageTs>2006-01-22T18:50:10.043Z</ns1:MessageTs>
            <ns1:ETIN>00123</ns1:ETIN>
            <ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
            <ns1:TestCd>P</ns1:TestCd>
            <ns1:AppSysID>username</ns1:AppSysID>
            <ns1:WSDLVersionNum>10.8</ns1:WSDLVersionNum>
        </ns1:MeFHeader>
        <!-- wsse:Security omitted from example, similar to GetNewSubmissions-->
    </SOAP-ENV:Header>
    <SOAP-ENV:Body>
        <GetAckNotificationsRequestArgs
            xmlns="http://www.irs.gov/a2a/mef/MeFStateService.xsd">
            <SubmissionIdList>
                <SubmissionId>00123420060380000097</SubmissionId>
                <SubmissionId>00123420060380000098</SubmissionId>
            </SubmissionIdList>
        </GetAckNotificationsRequestArgs>
    </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

4.2.16 GetAckNotifications Service Response Message

```

<e:Envelope xmlns:wn1="http://www.irs.gov/a2a/mef/MeFStateService.xsd"
    xmlns:wn0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
    xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:e="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:d="http://www.w3.org/2001/XMLSchema">
    <e:Header>

```



```

    <ns0:MeFHeader xmlns:ns0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
Id="hdr">
    <ns0:MessageID>00123200603900000042R</ns0:MessageID>
    <ns0:RelatesTo>00123200603900000042</ns0:RelatesTo>
    <ns0:Action>GetAckNotifications</ns0:Action>
    <ns0:MessageTs>2006-02-08T22:56:54Z</ns0:MessageTs>
    <ns0:ETIN>00123</ns0:ETIN>
    <ns0:SessionKeyCd>Y</ns0:SessionKeyCd>
    <ns0:TestCd>P</ns0:TestCd>
    <ns0:AppSysID>username</ns0:AppSysID>
    <ns0:WSDLVersionNum>10.8</ns0:WSDLVersionNum>
  </ns0:MeFHeader>
  <!-- wsse:Security omitted from example, similar to GetNewSubmissions-->
</e:Header>
<e:Body>
  <ns10:GetAckNotificationsResponse
xmlns:ns10="http://www.irs.gov/a2a/mef/MeFStateService.xsd"/>
  <ns10:SubmissionErrorList>
    <ns10:Cnt>2</ns10:Cnt>
    <ns10:SubmissionError>
      <ns10:SubmissionId>00123420060380000097</ns10:SubmissionId>
      <ns10:ErrorClassificationDetail>
        <ns10:ErrorMessageTxt>Backend system
error</ns10:ErrorMessageTxt>

        <ns10:ErrorClassificationCd>SYSTEM_ERROR</ns10:ErrorClas
sificationCd>
        <ns10:ErrorMessageCd>SYSTMIO07500002</ns10:ErrorMessageCd>
      </ns10:ErrorClassificationDetail>
    </ns10:SubmissionError>
    <ns10:SubmissionError>
      <ns10:SubmissionId>00123420060380000098</ns10:SubmissionId>
      <ns10:ErrorClassificationDetail>
        <ns10:ErrorMessageTxt>Backend system
error</ns10:ErrorMessageTxt>

        <ns10:ErrorClassificationCd>SYSTEM_ERROR</ns10:ErrorClas
sificationCd>
        <ns10:ErrorMessageCd>SYSTMIO07500002</ns10:ErrorMessageCd>
      </ns10:ErrorClassificationDetail>
    </ns10:SubmissionError>
  </ns10:SubmissionErrorList>
</ns10:GetAckNotificationsResponse>
</e:Body>
</e:Envelope>

```

4.2.17 GetAckNotifications Service Request Message

```

<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Header>

```




```

    <ns1:MeFHeader xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
Id="hdr">
      <ns1:MessageID>00123200603900000043</ns1:MessageID>
      <ns1:Action>GetAckNotification</ns1:Action>
      <ns1:MessageTs>2006-02-19T00:10:08.518Z</ns1:MessageTs>
      <ns1:ETIN>00123</ns1:ETIN>
      <ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
      <ns1:TestCd>P</ns1:TestCd>
      <ns1:AppSysID>username</ns1:AppSysID>
      <ns1:WSDLVersionNum>10.8</ns1:WSDLVersionNum>
    </ns1:MeFHeader>
    <!-- wsse:Security omitted from example, similar to GetNewSubmissions-->
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <GetAckNotificationRequestArgs
xmlns="http://www.irs.gov/a2a/mef/MeFStateService.xsd">
      <SubmissionId>00123420060380000097</SubmissionId>
    </GetAckNotificationRequestArgs>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

4.2.18 GetAckNotifications Service Response Message

```

<?xml version="1.0" encoding="UTF-8"?>
<e:Envelope xmlns:wn1="http://www.irs.gov/a2a/mef/MeFStateService.xsd"
  xmlns:wn0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:e="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:d="http://www.w3.org/2001/XMLSchema">
  <e:Header>
    <ns0:MeFHeader xmlns:ns0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
Id="hdr">
      <ns0:MessageID>00123200603900000043R</ns0:MessageID>
      <ns0:RelatesTo>00123200603900000043</ns0:RelatesTo>
      <ns0:Action>GetAckNotification</ns0:Action>
      <ns0:MessageTs>2006-02-08T22:57:04Z</ns0:MessageTs>
      <ns0:ETIN>00123</ns0:ETIN>
      <ns0:SessionKeyCd>Y</ns0:SessionKeyCd>
      <ns0:TestCd>P</ns0:TestCd>
      <ns0:AppSysID>username</ns0:AppSysID>
      <ns0:WSDLVersionNum>10.8</ns0:WSDLVersionNum>
    </ns0:MeFHeader>
    <!-- wsse:Security omitted from example, similar to GetNewSubmissions-->
  </e:Header>
  <e:Body>
    <ns10:GetAckNotificationResponse
xmlns:ns10="http://www.irs.gov/a2a/mef/MeFStateService.xsd"/>
  </e:Body>
</e:Envelope>

```



4.2.19 GetSubmissionReconciliationList Service Request Message

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Header>
    <ns1:MeFHeader xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
      Id="hdr">
      <ns1:MessageID>00123200603900000002</ns1:MessageID>
      <ns1:Action>GetSubmissionReconciliationList</ns1:Action>
      <ns1:MessageTs>2006-02-24T09:53:51.237Z</ns1:MessageTs>
      <ns1:ETIN>00123</ns1:ETIN>
      <ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
      <ns1:TestCd>P</ns1:TestCd>
      <ns1:AppSysID>username</ns1:AppSysID>
      <ns1:WSDLVersionNum>10.8</ns1:WSDLVersionNum>
    </ns1:MeFHeader>
    <!-- wsse:Security omitted from example, similar to GetNewSubmissions-->
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <GetSubmissionReconciliationListRequest
      xmlns="http://www.irs.gov/a2a/mef/MeFStateService.xsd">
      <MaxResultCnt>2</MaxResultCnt>
      <SubsetStSubmissionCategoryCd>CORP</SubsetStSubmissionCategoryCd>
      <StartDt>2006-02-02T09:30:47.0Z</StartDt>
      <EndDt>2006-02-17T09:30:47.0Z</EndDt>
      <MeFSubmissionStatusCd>RECEIPTNOACK</MeFSubmissionStatusCd>
    <!-- Note: The last 4 elements are optional and need not be present in the
      actual request -->
    </GetSubmissionReconciliationListRequest >
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

4.2.20 GetSubmissionReconciliationList Service Response Message

```
<?xml version="1.0" encoding="UTF-8"?>
<e:Envelope xmlns:wn1="http://www.irs.gov/a2a/mef/MeFStateService.xsd"
  xmlns:ns0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:e="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:d="http://www.w3.org/2001/XMLSchema">
  <e:Header>
    <ns0:MeFHeader xmlns:ns0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
      Id="hdr">
      <ns0:MessageID>00123200603900000002R</ns0:MessageID>
      <ns0:RelatesTo>00123200603900000002</ns0:RelatesTo>
      <ns0:Action>GetSubmissionReconciliationList</ns0:Action>
      <ns0:MessageTs>2006-02-08T22:42:58Z</ns0:MessageTs>
      <ns0:ETIN>00123</ns0:ETIN>
      <ns0:SessionKeyCd>Y</ns0:SessionKeyCd>
```



```

        <ns0:TestCd>P</ns0:TestCd>
        <ns0:AppSysID>username</ns0:AppSysID>
        <ns0:WSDLVersionNum>10.8</ns0:WSDLVersionNum>
    </ns0:MeFHeader>
    <!-- wsse:Security omitted from example, similar to GetNewSubmissions-->
</e:Header>
<e:Body>
    <ns10:GetSubmissionReconciliationListResponse
        xmlns:ns10="http://www.irs.gov/a2a/mef/MeFStateService.xsd">
        <ns10:SubmissionId>00123420060320000057</ns10:SubmissionId>
        <ns10:SubmissionId>00123420060320000058</ns10:SubmissionId>
    <!-- Note: The above elements are optional and may or may not be present in
the actual response -->
        <ns10:MoreAvailableInd>true</ns10:MoreAvailableInd>
        <ns10:TotalCnt>50</ns10:TotalCnt>
    </ns10:GetSubmissionReconciliationListResponse>
</e:Body>
</e:Envelope>

```

4.3 Transmitter Services SOAP Messages

This section presents SOAP request and response message examples for the following Transmitter services:

- **SendSubmissions**
- **GetNewAcks**
- **GetAcksByMsgID**
- **GetAcks**
- **GetAck**
- **GetNewSubmissionsStatus**
- **GetSubmissionsStatus**
- **GetSubmissionStatus**
- **GetNew2290Schedule1s**
- **Get2290Schedule1sByMsgID**
- **Get2290Schedule1s**
- **Get2290Schedule1**

NOTE: All of the SOAP request and response messages are identical for MIME and MTOM, except for the attachment transport mechanism.

4.3.1 SendSubmissions Service Request Message

```

<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Header>
    <ns1:MeFHeader

```



```

xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd">
<ns1:MessageID>1232020161772321970y</ns1:MessageID>
<ns1:Action>SendSubmissions</ns1:Action>
<ns1:MessageTs>2024-09-23T23:03:11.155Z</ns1:MessageTs>
<ns1:ETIN>12320</ns1:ETIN>
<ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
<ns1:TestCd>P</ns1:TestCd>
<ns1:AppSysID />
<ns1:WSDLVersionNum>10.8</ns1:WSDLVersionNum>
<ns1:ClientSoftwareTxt>SOATest</ns1:ClientSoftwareTxt>
</ns1:MeFHeader>
<wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-
200401-wss-wssecurity-secext-1.0.xsd">
  <wsu:Timestamp wsu:Id="id-98d90235a251d1804e76301ef65620f41e1a477c"
    xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd">
    <wsu:Created>2024-09-23T23:01:58Z</wsu:Created>
    <wsu:Expires>2024-09-24T01:01:58Z</wsu:Expires>
  </wsu:Timestamp>
  <wsse:UsernameToken wsu:Id="id-570235cb3b276ac7429563bd095a7ecf4c45f3e4"
    xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd">
    <wsse:Username>12320002</wsse:Username>
  </wsse:UsernameToken>
  <saml:Assertion
    AssertionID="id-cf18f9cb13ee18825777a5060b14ab9e3c6c1c33"
    IssueInstant="2024-09-23T23:01:58.282Z"
    Issuer="http://www.forumsys.com/sentry" MajorVersion="1"
    MinorVersion="1" xmlns:saml="urn:oasis:names:tc:SAML:1.0:assertion">
    <saml:Conditions
      NotBefore="2024-09-23T23:01:58.282Z"
      NotOnOrAfter="2024-09-24T01:01:58.282Z" />
    <saml:AuthenticationStatement
      AuthenticationInstant="2024-09-23T23:01:58.282Z"
      AuthenticationMethod="urn:oasis:names:tc:SAML:1.0:am:password">
      <saml:Subject>
        <saml:NameIdentifier Format="urn:oasis:names:tc:SAML:1.1:nameid-
format:unspecified">
          UID=12320002+DNQ=300757, OU=A2A, O=US Government, C=US
        </saml:NameIdentifier>
        <saml:SubjectConfirmation>
          <saml:ConfirmationMethod>urn:oasis:names:tc:SAML:1.0:cm:bearer
        </saml:ConfirmationMethod>
        </saml:SubjectConfirmation>
      </saml:Subject>
    </saml:AuthenticationStatement>
    <saml:AttributeStatement>
      <saml:Subject>
        <saml:NameIdentifier Format="urn:oasis:names:tc:SAML:1.1:nameid-
format:unspecified">
          UID=12320002+DNQ=300757, OU=A2A, O=US Government, C=US
        </saml:NameIdentifier>
        <saml:SubjectConfirmation>

```



```

    <saml:ConfirmationMethod>urn:oasis:names:tc:SAML:1.0:cm:bearer
  </saml:ConfirmationMethod>
</saml:SubjectConfirmation>
</saml:Subject>
<saml:Attribute AttributeName="SMSESSION"
  AttributeNamespace="http://www.forumsys.com/sentry">
  <saml:AttributeValue>jdh0aPEnIYFWhIuAl1KPjOUfMAVm
  </saml:AttributeValue>
</saml:Attribute>
</saml:AttributeStatement>
<ds:Signature
  xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
  <ds:SignedInfo>
    <ds:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-
exc-c14n#" />
    <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-
more#rsa-sha256" />
    <ds:Reference URI="#id-cf18f9cb13ee18825777a5060b14ab9e3c6c1c33">
      <ds:Transforms>
        <ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-
signature" />
        <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
      </ds:Transforms>
      <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"
/>
      <ds:DigestValue>MeqZyQ+osoCXFNxid6Z93UuncqFHEBGEr/AmvJeD3x0=
      </ds:DigestValue>
    </ds:Reference>
  </ds:SignedInfo>
  <ds:SignatureValue>SignatureB64</ds:SignatureValue>
  <ds:KeyInfo>
    <ds:X509Data>
      <ds:X509Certificate>MII+CertInB64</ds:X509Certificate>
    </ds:X509Data>
  </ds:KeyInfo>
</ds:Signature>
</saml:Assertion>
</wsse:Security>
</SOAP-ENV:Header>
<SOAP-ENV:Body>
  <SendSubmissionsRequest
xmlns="http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd">
  <SubmissionDataList>
    <SubmissionData>
      <SubmissionId>1100052024264a104687</SubmissionId>
      <ElectronicPostmarkTs>2024-09-23T23:03:11.158Z
      </ElectronicPostmarkTs>
    </SubmissionData>
  </SubmissionDataList>
</SendSubmissionsRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```



4.3.2 SendSubmissions Service Response Message

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Header>
    <ns1:MeFHeader
      xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd">
      <ns1:MessageID>1232020161772321970yR</ns1:MessageID>
      <ns1:RelatesTo>1232020161772321970y</ns1:RelatesTo>
      <ns1:Action>SendSubmissions</ns1:Action>
      <ns1:MessageTs>2024-09-23T23:03:11.155Z</ns1:MessageTs>
      <ns1:ETIN>12320</ns1:ETIN>
      <ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
      <ns1:TestCd>P</ns1:TestCd>
      <ns1:AppSysID />
      <ns1:WSDLVersionNum>10.8</ns1:WSDLVersionNum>
      <ns1:ClientSoftwareTxt>SOATest</ns1:ClientSoftwareTxt>
    </ns1:MeFHeader>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
      <wsu:Timestamp xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
        wsu:Id="id-98d90235a251d1804e76301ef65620f41e1a477c">
        <wsu:Created>2024-09-23T23:01:58Z</wsu:Created>
        <wsu:Expires>2024-09-24T01:01:58Z</wsu:Expires>
      </wsu:Timestamp>
      <wsse:UsernameToken xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
        wsu:Id="id-570235cb3b276ac7429563bd095a7ecf4c45f3e4">
        <wsse:Username>12320002</wsse:Username>
      </wsse:UsernameToken>
      <saml:Assertion xmlns:saml="urn:oasis:names:tc:SAML:1.0:assertion"
        AssertionID="id-cf18f9cb13ee18825777a5060b14ab9e3c6c1c33"
        IssueInstant="2024-09-23T23:01:58.282Z"
        Issuer="http://www.forumsys.com/sentry" MajorVersion="1"
        MinorVersion="1">
        <saml:Conditions
          NotBefore="2024-09-23T23:01:58.282Z"
          NotOnOrAfter="2024-09-24T01:01:58.282Z" />
        <saml:AuthenticationStatement
          AuthenticationInstant="2024-09-23T23:01:58.282Z"
          AuthenticationMethod="urn:oasis:names:tc:SAML:1.0:am:password">
          <saml:Subject>
            <saml:NameIdentifier Format="urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified">
              UID=12320002+DNQ=300757, OU=A2A, O=US Government, C=US
            </saml:NameIdentifier>
            <saml:SubjectConfirmation>
              <saml:ConfirmationMethod>urn:oasis:names:tc:SAML:1.0:cm:bearer
            </saml:ConfirmationMethod>
            </saml:SubjectConfirmation>
          </saml:Subject>
        </saml:AuthenticationStatement>
      </saml:Assertion>
    </wsse:Security>
  </soapenv:Header>
  <soapenv:Body>
  </soapenv:Body>
</soapenv:Envelope>
```



```

</saml:AuthenticationStatement>
<saml:AttributeStatement>
  <saml:Subject>
    <saml:NameIdentifier Format="urn:oasis:names:tc:SAML:1.1:nameid-
format:unspecified">
      UID=12320002+DNQ=300757, OU=A2A, O=US Government, C=US
    </saml:NameIdentifier>
    <saml:SubjectConfirmation>
      <saml:ConfirmationMethod>urn:oasis:names:tc:SAML:1.0:cm:bearer
    </saml:ConfirmationMethod>
    </saml:SubjectConfirmation>
  </saml:Subject>
  <saml:Attribute AttributeName="SMSESSION"
AttributeNamespace="http://www.forumsys.com/sentry">
    <saml:AttributeValue>jdh0aPEnIYFWhIuAl1KPjOUfMAVm
    </saml:AttributeValue>
  </saml:Attribute>
</saml:AttributeStatement>
<ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
  <ds:SignedInfo>
    <ds:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-
exc-c14n#" />
    <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-
more#rsa-sha256" />
    <ds:Reference URI="#id-cf18f9cb13ee18825777a5060b14ab9e3c6c1c33">
      <ds:Transforms>
        <ds:Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-
signature" />
        <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#" />
      </ds:Transforms>
      <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"
/>
      <ds:DigestValue>MeqZyQ+osoCXFNxid6Z93UuncqFHEBGEr/AmvJeD3x0=
    </ds:DigestValue>
    </ds:Reference>
  </ds:SignedInfo>
  <ds:SignatureValue>SignatureInB64</ds:SignatureValue>
  <ds:KeyInfo>
    <ds:X509Data>
      <ds:X509Certificate>MII+CertInB64</ds:X509Certificate>
    </ds:X509Data>
  </ds:KeyInfo>
</ds:Signature>
</saml:Assertion>
</wsse:Security>
</soapenv:Header>
<soapenv:Body>
  <ns2:SendSubmissionsResponse
xmlns:ns2="http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd">
    <ns2:DepositId>NEFAFKN8X33W94PSJGQ6HN7M2SJC9X01JARBD4VAE2NBS
    </ns2:DepositId>
  </ns2:SendSubmissionsResponse>
</soapenv:Body>

```



```
</soapenv:Envelope>
```

4.3.3 GetNewAcks Service Request Message

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Header>
    <ns1:MeFHeader xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
      Id="hdr">
      <ns1:MessageID>00123200603900000031</ns1:MessageID>
      <ns1:Action>GetNewAcks</ns1:Action>
      <ns1:MessageTs>2006-02-21T23:03:19.055Z</ns1:MessageTs>
      <ns1:ETIN>00123</ns1:ETIN>
      <ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
      <ns1:TestCd>P</ns1:TestCd>
      <ns1:AppSysID>username</ns1:AppSysID>
      <ns1:WSDLVersionNum>10.8</ns1:WSDLVersionNum>
    </ns1:MeFHeader>
    <!-- wsse:Security omitted from example -->
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <GetNewAcksRequest
      xmlns="http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd">
      <MaxResultCnt>2</MaxResultCnt>
      <ExtndAcknowledgementCategoryCd>CORP</ExtndAcknowledgementCategoryCd>
      <GovernmentAgencyTypeCd>Federal</GovernmentAgencyTypeCd>
    </GetNewAcksRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

4.3.4 GetNewAcks Service Response Message

```
<?xml version="1.0" encoding="UTF-8"?>
<e:Envelope xmlns:wn1="http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd"
  xmlns:wn0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:e="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:d="http://www.w3.org/2001/XMLSchema">
  <e:Header>
    <ns0:MeFHeader xmlns:ns0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
      Id="hdr">
      <ns0:MessageID>00123200603900000031R</ns0:MessageID>
      <ns0:RelatesTo>00123200603900000031</ns0:RelatesTo>
      <ns0:Action>GetNewAcks</ns0:Action>
      <ns0:MessageTs>2006-02-08T22:54:19Z</ns0:MessageTs>
      <ns0:ETIN>00123</ns0:ETIN>
      <ns0:SessionKeyCd>Y</ns0:SessionKeyCd>
      <ns0:TestCd>P</ns0:TestCd>
      <ns0:AppSysID>username</ns0:AppSysID>
    </ns0:MeFHeader>
  </e:Header>
  <e:Body>
    <GetNewAcksResponse
      xmlns="http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd">
      <MaxResultCnt>2</MaxResultCnt>
      <ExtndAcknowledgementCategoryCd>CORP</ExtndAcknowledgementCategoryCd>
      <GovernmentAgencyTypeCd>Federal</GovernmentAgencyTypeCd>
    </GetNewAcksResponse>
  </e:Body>
</e:Envelope>
```




```

        <ns0:WSDLVersionNum>10.8</ns0:WSDLVersionNum>
    </ns0:MeFHeader>
    <!-- wsse:Security omitted from example, similar to SendSubmissions-->
</e:Header>
<e:Body>
    <ns1:GetNewAcksResponse
        xmlns:ns1="http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd">
        <ns1:MoreAvailableInd>true</ns1:MoreAvailableInd>
    </ns1:GetNewAcksResponse>
</e:Body>
</e:Envelope>

```

4.3.5 Get Acks by Message ID Service Request Message

```

<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <SOAP-ENV:Header>
        <ns1:MeFHeader xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
            Id="hdr">
            <ns1:MessageID>00123200603900000032</ns1:MessageID>
            <ns1:Action>GetAcksByMsgID</ns1:Action>
            <ns1:MessageTs>2006-01-30T19:12:40.557Z</ns1:MessageTs>
            <ns1:ETIN>00123</ns1:ETIN>
            <ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
            <ns1:TestCd>P</ns1:TestCd>
            <ns1:AppSysID>username</ns1:AppSysID>
            <ns1:WSDLVersionNum>10.8</ns1:WSDLVersionNum>
        </ns1:MeFHeader>
        <!-- wsse:Security omitted from example, similar to SendSubmissions-->
    </SOAP-ENV:Header>
    <SOAP-ENV:Body>
        <GetAcksByMsgIDRequest
            xmlns="http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd">
            <MessageId>00123200603900000031</MessageId>
        </GetAcksByMsgIDRequest>
    </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

4.3.6 Get Acks by Message ID Service Response Message

```

<?xml version="1.0" encoding="UTF-8"?>
<e:Envelope xmlns:wn1="http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd"
    xmlns:wn0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
    xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:e="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:d="http://www.w3.org/2001/XMLSchema">
    <e:Header>
        <ns0:MeFHeader xmlns:ns0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
            Id="hdr">
            <ns0:MessageID>00123200603900000032R</ns0:MessageID>

```



```

        <ns0:RelatesTo>00123200603900000032</ns0:RelatesTo>
        <ns0:Action>GetAcksByMsgID</ns0:Action>
        <ns0:MessageTs>2006-02-08T22:54:26Z</ns0:MessageTs>
        <ns0:ETIN>00123</ns0:ETIN>
        <ns0:SessionKeyCd>Y</ns0:SessionKeyCd>
        <ns0:TestCd>P</ns0:TestCd>
        <ns0:AppSysID>username</ns0:AppSysID>
        <ns0:WSDLVersionNum>10.8</ns0:WSDLVersionNum>
    </ns0:MeFHeader>
    <!-- wsse:Security omitted from example, similar to SendSubmissions-->
</e:Header>
<e:Body>
    <ns10:GetAcksByMsgIDResponse
        xmlns:ns10="http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd">
        <ns10:MoreAvailableInd>>false</ns10:MoreAvailableInd>
    </ns10:GetAcksByMsgIDResponse>
</e:Body>
</e:Envelope>

```

4.3.7 GetAcks Service Request Message

```

<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <SOAP-ENV:Header>
        <ns1:MeFHeader xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
            Id="hdr">
            <ns1:MessageID>00123200603900000033</ns1:MessageID>
            <ns1:Action>GetAcks</ns1:Action>
            <ns1:MessageTs>2006-02-26T14:20:42.526Z</ns1:MessageTs>
            <ns1:ETIN>00123</ns1:ETIN>
            <ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
            <ns1:TestCd>P</ns1:TestCd>
            <ns1:AppSysID>username</ns1:AppSysID>
            <ns1:WSDLVersionNum>10.8</ns1:WSDLVersionNum>
        </ns1:MeFHeader>
        <!-- wsse:Security omitted from example, similar to SendSubmissions-->
    </SOAP-ENV:Header>
    <SOAP-ENV:Body>
        <GetAcksRequest
            xmlns="http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd">
            <SubmissionIdList>
                <SubmissionId>00123420060380000096</SubmissionId>
                <SubmissionId>00123420060380000097</SubmissionId>
            </SubmissionIdList>
        </GetAcksRequest>
    </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```



4.3.8 GetAcks Service Response Message

```
<?xml version="1.0" encoding="UTF-8"?>
<e:Envelope xmlns:wn1="http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd"
  xmlns:wn0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:e="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:d="http://www.w3.org/2001/XMLSchema">
  <e:Header>
    <ns0:MeFHeader xmlns:ns0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
      Id="hdr">
      <ns0:MessageID>00123200603900000033R</ns0:MessageID>
      <ns0:RelatesTo>00123200603900000033</ns0:RelatesTo>
      <ns0:Action>GetAcks</ns0:Action>
      <ns0:MessageTs>2006-02-08T22:54:31Z</ns0:MessageTs>
      <ns0:ETIN>00123</ns0:ETIN>
      <ns0:SessionKeyCd>Y</ns0:SessionKeyCd>
      <ns0:TestCd>P</ns0:TestCd>
      <ns0:AppSysID>username</ns0:AppSysID>
      <ns0:WSDLVersionNum>10.8</ns0:WSDLVersionNum>
    </ns0:MeFHeader>
    <!-- wsse:Security omitted from example, similar to SendSubmissions-->
  </e:Header>
  <e:Body>
    <ns10:GetAcksResponse
      xmlns:ns10="http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd">
      <ns10:SubmissionErrorList>
        <ns10:Cnt>0</ns10:Cnt>
        <ns10:SubmissionError/>
        <!-- Note: The above element is optional and may or may not be present in the
          actual response -->
      </ns10:SubmissionErrorList>
    </ns10:GetAcksResponse>
  </e:Body>
</e:Envelope>
```

4.3.9 GetAck Service Request Message

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Header>
    <ns1:MeFHeader xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
      Id="hdr">
      <ns1:MessageID>00123200603900000034</ns1:MessageID>
      <ns1:Action>GetAck</ns1:Action>
      <ns1:MessageTs>2006-01-19T21:27:25.696Z</ns1:MessageTs>
      <ns1:ETIN>00123</ns1:ETIN>
      <ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
      <ns1:TestCd>P</ns1:TestCd>
      <ns1:AppSysID>username</ns1:AppSysID>
      <ns1:WSDLVersionNum>10.8</ns1:WSDLVersionNum>
    </ns1:MeFHeader>
```



```

        <!-- wsse:Security omitted from example, similar to SendSubmissions-->
    </SOAP-ENV:Header>
    <SOAP-ENV:Body>
        <GetAckRequest
xmlns="http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd">
            <SubmissionId>00123420060380000096</SubmissionId>
        </GetAckRequest>
    </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

4.3.10 GetAck Service Response Message

```

<?xml version="1.0" encoding="UTF-8"?>
<e:Envelope xmlns:wn1="http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd"
  xmlns:wn0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:e="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:d="http://www.w3.org/2001/XMLSchema">
  <e:Header>
    <ns0:MeFHeader xmlns:ns0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
Id="hdr">
      <ns0:MessageID>00123200603900000034R</ns0:MessageID>
      <ns0:RelatesTo>00123200603900000034</ns0:RelatesTo>
      <ns0:Action>GetAck</ns0:Action>
      <ns0:MessageTs>2006-02-08T22:54:36Z</ns0:MessageTs>
      <ns0:ETIN>00123</ns0:ETIN>
      <ns0:SessionKeyCd>Y</ns0:SessionKeyCd>
      <ns0:TestCd>P</ns0:TestCd>
      <ns0:AppSysID>username</ns0:AppSysID>
      <ns0:WSDLVersionNum>10.8</ns0:WSDLVersionNum>
    </ns0:MeFHeader>
    <!-- wsse:Security omitted from example, similar to SendSubmissions-->
  </e:Header>
  <e:Body>
    <ns10:GetAckResponse
xmlns:ns10="http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd"/>
  </e:Body>
</e:Envelope>

```

4.3.11 GetNewSubmissionsStatus Service Request Message

```

<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Header>
    <ns1:MeFHeader xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
Id="hdr">
      <ns1:MessageID>00123200603900000021</ns1:MessageID>
      <ns1:Action>GetNewSubmissionsStatus</ns1:Action>
      <ns1:MessageTs>2006-01-18T20:42:13.189Z</ns1:MessageTs>
      <ns1:ETIN>00123</ns1:ETIN>
    </ns1:MeFHeader>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <ns1:NewSubmissionsStatusRequest xmlns:ns1="http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd">
      <SubmissionId>00123420060380000096</SubmissionId>
    </ns1:NewSubmissionsStatusRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```



```

        <ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
        <ns1:TestCd>P</ns1:TestCd>
        <ns1:AppSysID>username</ns1:AppSysID>
        <ns1:WSDLVersionNum>10.8</ns1:WSDLVersionNum>
    </ns1:MeFHeader>
    <!-- wsse:Security omitted from example, similar to SendSubmissions-->
</SOAP-ENV:Header>
<SOAP-ENV:Body>
    <GetNewSubmissionsStatusRequest
        xmlns="http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd">
        <MaxResultCnt>100</MaxResultCnt>
    </GetNewSubmissionsStatusRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

4.3.12 GetNewSubmissionsStatus Service Response Message

```

<?xml version="1.0" encoding="UTF-8"?>
<e:Envelope xmlns:wn1="http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd"
    xmlns:wn0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
    xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:e="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:d="http://www.w3.org/2001/XMLSchema">
    <e:Header>
        <ns0:MeFHeader xmlns:ns0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
            Id="hdr">
            <ns0:MessageID>00123200603900000021R</ns0:MessageID>
            <ns0:RelatesTo>00123200603900000021</ns0:RelatesTo>
            <ns0:Action>GetNewSubmissionsStatus</ns0:Action>
            <ns0:MessageTs>2006-02-08T22:33:22Z</ns0:MessageTs>
            <ns0:ETIN>00123</ns0:ETIN>
            <ns0:SessionKeyCd>Y</ns0:SessionKeyCd>
            <ns0:TestCd>P</ns0:TestCd>
            <ns0:AppSysID>username</ns0:AppSysID>
            <ns0:WSDLVersionNum>10.8</ns0:WSDLVersionNum>
        </ns0:MeFHeader>
        <!-- wsse:Security omitted from example, similar to SendSubmissions-->
    </e:Header>
    <e:Body>
        <ns10:GetNewSubmissionsStatusResponse
            xmlns:ns10="http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd">
            <ns10:MoreAvailableInd>true</ns10:MoreAvailableInd>
        </ns10:GetNewSubmissionsStatusResponse>
    </e:Body>
</e:Envelope>

```

4.3.13 GetSubmissionsStatus Service Request Message

```

<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

```



```

<SOAP-ENV:Header>
  <ns1:MeFHeader xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
Id="hdr">
    <ns1:MessageID>00123200603900000023</ns1:MessageID>
    <ns1:Action>GetSubmissionsStatus</ns1:Action>
    <ns1:MessageTs>2006-01-28T11:35:20.165Z</ns1:MessageTs>
    <ns1:ETIN>00123</ns1:ETIN>
    <ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
    <ns1:TestCd>P</ns1:TestCd>
    <ns1:AppSysID>username</ns1:AppSysID>
    <ns1:WSDLVersionNum>10.8</ns1:WSDLVersionNum>
  </ns1:MeFHeader>
  <!-- wsse:Security omitted from example, similar to SendSubmissions-->
</SOAP-ENV:Header>
<SOAP-ENV:Body>
  <GetSubmissionsStatusRequest
xmlns="http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd">
    <SubmissionIdList>
      <SubmissionId>0600082013261007926e</SubmissionId>
      <SubmissionId>0600082013261007926f</SubmissionId>
      <SubmissionId>0600082013261007926g</SubmissionId>
      <SubmissionId>0600082013261007926h</SubmissionId>
    </SubmissionIdList>
  </GetSubmissionsStatusRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

4.3.14 GetSubmissionsStatus Service Response Message

```

<?xml version="1.0" encoding="UTF-8"?>
<e:Envelope xmlns:wn1="http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd"
xmlns:wn0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns:e="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:d="http://www.w3.org/2001/XMLSchema">
  <e:Header>
    <ns0:MeFHeader xmlns:ns0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
Id="hdr">
      <ns0:MessageID>00123200603900000023R</ns0:MessageID>
      <ns0:RelatesTo>00123200603900000023</ns0:RelatesTo>
      <ns0:Action>GetSubmissionsStatus</ns0:Action>
      <ns0:MessageTs>2006-02-08T22:33:45Z</ns0:MessageTs>
      <ns0:ETIN>00123</ns0:ETIN>
      <ns0:SessionKeyCd>Y</ns0:SessionKeyCd>
      <ns0:TestCd>P</ns0:TestCd>
      <ns0:AppSysID>username</ns0:AppSysID>
      <ns0:WSDLVersionNum>10.8</ns0:WSDLVersionNum>
    </ns0:MeFHeader>
    <!-- wsse:Security omitted from example, similar to SendSubmissions-->
  </e:Header>
  <e:Body>
    <ns10:GetSubmissionsStatusResponse
xmlns:ns10="http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd">
      <ns10:SubmissionErrorList>

```



```

        <ns10:Cnt>0</ns10:Cnt>
        <ns10:SubmissionError/>
    <!-- Note: The above element is optional and may or may not be present in the
    actual response -->
    </ns10:SubmissionErrorList>
</ns10:GetSubmissionsStatusResponse>
</e:Body>
</e:Envelope>

```

4.3.15 GetSubmissionStatus Service Request Message

```

<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Header>
    <ns1:MeFHeader xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
      Id="hdr">
      <ns1:MessageID>00123200603900000022</ns1:MessageID>
      <ns1:Action>GetSubmissionStatus</ns1:Action>
      <ns1:MessageTs>2006-02-18T15:17:34.030Z</ns1:MessageTs>
      <ns1:ETIN>00123</ns1:ETIN>
      <ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
      <ns1:TestCd>P</ns1:TestCd>
      <ns1:AppSysID>username</ns1:AppSysID>
      <ns1:WSDLVersionNum>10.8</ns1:WSDLVersionNum>
    </ns1:MeFHeader>
    <!-- wsse:Security omitted from example, similar to SendSubmissions-->
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <GetSubmissionStatusRequest
      xmlns="http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd">
      <SubmissionId>00123420060380000096</SubmissionId>
    </GetSubmissionStatusRequest>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

4.3.16 GetSubmissionStatus Service Response Message

```

<?xml version="1.0" encoding="UTF-8"?>
<e:Envelope xmlns:wn1="http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd"
  xmlns:wn0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:e="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:d="http://www.w3.org/2001/XMLSchema">
  <e:Header>
    <ns0:MeFHeader xmlns:ns0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
      Id="hdr">
      <ns0:MessageID>00123200603900000022R</ns0:MessageID>
      <ns0:RelatesTo>00123200603900000022</ns0:RelatesTo>
      <ns0:Action>GetSubmissionStatus</ns0:Action>
      <ns0:MessageTs>2006-02-08T22:33:35Z</ns0:MessageTs>
    </ns0:MeFHeader>
  </e:Header>
  <e:Body>
    <GetSubmissionStatusResponse
      xmlns="http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd">
      <SubmissionId>00123420060380000096</SubmissionId>
      <SubmissionStatus>Success</SubmissionStatus>
    </GetSubmissionStatusResponse>
  </e:Body>
</e:Envelope>

```



```

        <ns0:ETIN>00123</ns0:ETIN>
        <ns0:SessionKeyCd>Y</ns0:SessionKeyCd>
        <ns0:TestCd>P</ns0:TestCd>
        <ns0:AppSysID>username</ns0:AppSysID>
        <ns0:WSDLVersionNum>10.8</ns0:WSDLVersionNum>
    </ns0:MeFHeader>
    <!-- wsse:Security omitted from example, similar to SendSubmissions-->
</e:Header>
<e:Body>
    <ns10:GetSubmissionStatusResponse
        xmlns:ns10="http://www.irs.gov/a2a/mef/MeFTransmitterService.xsd"/>
</e:Body>
</e:Envelope>

```

4.3.17 GetNew2290Schedule1 Service Request Message

```

<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <SOAP-ENV:Header>
        <ns1:MeFHeader xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
            Id="hdr">
            <ns1:MessageID>00123200703900000031</ns1:MessageID>
            <ns1:Action>GetNew2290Schedule1s</ns1:Action>
            <ns1:MessageTs>2007-02-21T23:03:19.055Z</ns1:MessageTs>
            <ns1:ETIN>00123</ns1:ETIN>
            <ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
            <ns1:TestCd>P</ns1:TestCd>
            <ns1:AppSysID>username</ns1:AppSysID>
            <ns1:WSDLVersionNum>10.8</ns1:WSDLVersionNum>
        </ns1:MeFHeader>
        <!-- wsse:Security omitted from example, similar to SendSubmissions-->
    </SOAP-ENV:Header>
    <SOAP-ENV:Body>
        <GetNew2290Schedule1sRequest
            xmlns="http://www.irs.gov/a2a/mef/ETECTransmitterService.xsd">
            <MaxResultCnt>3</MaxResultCnt>
        </GetNew2290Schedule1sRequest>
    </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

4.3.18 GetNew2290Schedule1s Service Response Message

```

<?xml version="1.0" encoding="UTF-8"?>
<e:Envelope xmlns:wn1="http://www.irs.gov/a2a/mef/ETECTransmitterService.xsd"
    xmlns:wn0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
    xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:e="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:d="http://www.w3.org/2001/XMLSchema">
    <e:Header>

```




```

    <ns0:MeFHeader xmlns:ns0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
Id="hdr">
    <ns0:MessageID>00123200703900000031R</ns0:MessageID>
    <ns0:RelatesTo>00123200703900000031</ns0:RelatesTo>
    <ns0:Action>GetNew2290Schedule1s</ns0:Action>
    <ns0:MessageTs>2007-02-08T22:54:19Z</ns0:MessageTs>
    <ns0:ETIN>00123</ns0:ETIN>
    <ns0:SessionKeyCd>Y</ns0:SessionKeyCd>
    <ns0:TestCd>P</ns0:TestCd>
    <ns0:AppSysID>username</ns0:AppSysID>
    <ns0:WSDLVersionNum>10.8</ns0:WSDLVersionNum>
    </ns0:MeFHeader>
    <!-- wsse:Security omitted from example, similar to SendSubmissions-->
</e:Header>
<e:Body>
    <ns10:GetNew2290Schedule1sResponse
    xmlns:ns10="http://www.irs.gov/a2a/mef/ETECTransmitterService.xsd">
    <ns10:Cnt>3</ns10:Cnt>
    <ns10:MoreAvailableInd>true</ns10:MoreAvailableInd>
    </ns10: GetNew2290Schedule1sResponse>
</e:Body>
</e:Envelope>

```

4.3.19 Get2290Schedule1sByMsgID Service Request Message

```

<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <SOAP-ENV:Header>
        <ns1:MeFHeader xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
Id="hdr">
            <ns1:MessageID>00123200703900000032</ns1:MessageID>
            <ns1:Action>Get2290Schedule1sByMsgID</ns1:Action>
            <ns1:MessageTs>2007-01-30T19:12:40.557Z</ns1:MessageTs>
            <ns1:ETIN>00123</ns1:ETIN>
            <ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
            <ns1:TestCd>P</ns1:TestCd>
            <ns1:AppSysID>username</ns1:AppSysID>
            <ns1:WSDLVersionNum>10.8</ns1:WSDLVersionNum>
        </ns1:MeFHeader>
        <!-- wsse:Security omitted from example, similar to SendSubmissions-->
    </SOAP-ENV:Header>
    <SOAP-ENV:Body>
        <Get2290Schedule1sByMsgIDRequest
            xmlns="http://www.irs.gov/a2a/mef/ETECTransmitterService.xsd">
            <MessageId>00123200703900000031</MessageId>
        </Get2290Schedule1sByMsgIDRequest>
    </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```



4.3.20 Get2290Schedule1sByMsgID Service Response Message

```
<?xml version="1.0" encoding="UTF-8"?>
<e:Envelope xmlns:wn1="http://www.irs.gov/a2a/mef/ETECTransmitterService.xsd"
  xmlns:wn0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:e="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:d="http://www.w3.org/2001/XMLSchema">
  <e:Header>
    <ns0:MeFHeader xmlns:ns0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
      Id="hdr">
      <ns0:MessageID>00123200703900000032R</ns0:MessageID>
      <ns0:RelatesTo>00123200703900000032</ns0:RelatesTo>
      <ns0:Action>Get2290Schedule1sByMsgID</ns0:Action>
      <ns0:MessageTs>2007-02-08T22:54:26Z</ns0:MessageTs>
      <ns0:ETIN>00123</ns0:ETIN>
      <ns0:SessionKeyCd>Y</ns0:SessionKeyCd>
      <ns0:TestCd>P</ns0:TestCd>
      <ns0:AppSysID>username</ns0:AppSysID>
      <ns0:WSDLVersionNum>10.8</ns0:WSDLVersionNum>
    </ns0:MeFHeader>
    <!-- wsse:Security omitted from example, similar to SendSubmissions-->
  </e:Header>
  <e:Body>
    <ns10:Get2290Schedule1sByMsgIDResponse
      xmlns:ns10="http://www.irs.gov/a2a/mef/ETECTransmitterService.xsd">
      <ns10:Cnt>3</ns10:Cnt>
      <ns10:MoreAvailableInd>true</ns10:MoreAvailableInd>
    </ns10:Get2290Schedule1sByMsgIDResponse>
  </e:Body>
</e:Envelope>
```

4.3.21 Get2290Schedule1s Service Request Message

```
<?xml version="1.0" encoding="UTF-8"?>
<e:Envelope xmlns:wn1="http://www.irs.gov/a2a/mef/ETECTransmitterService.xsd"
  xmlns:wn0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:e="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:d="http://www.w3.org/2001/XMLSchema">
  <e:Header>
    <ns0:MeFHeader xmlns:ns0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
      Id="hdr">
      <ns0:MessageID>00123200703900000033R</ns0:MessageID>
      <ns0:RelatesTo>00123200703900000033</ns0:RelatesTo>
      <ns0:Action>Get2290Schedule1s</ns0:Action>
      <ns0:MessageTs>2007-02-08T22:54:31Z</ns0:MessageTs>
      <ns0:ETIN>00123</ns0:ETIN>
      <ns0:SessionKeyCd>Y</ns0:SessionKeyCd>
      <ns0:TestCd>P</ns0:TestCd>
      <ns0:AppSysID>username</ns0:AppSysID>
      <ns0:WSDLVersionNum>10.8</ns0:WSDLVersionNum>
    </ns0:MeFHeader>
  </e:Header>
  <e:Body>
    <ns10:Get2290Schedule1sRequest
      xmlns:ns10="http://www.irs.gov/a2a/mef/ETECTransmitterService.xsd">
      <ns10:AppSysID>username</ns10:AppSysID>
      <ns10:ETIN>00123</ns10:ETIN>
      <ns10:SessionKeyCd>Y</ns10:SessionKeyCd>
      <ns10:TestCd>P</ns10:TestCd>
      <ns10:WSDLVersionNum>10.8</ns10:WSDLVersionNum>
    </ns10:Get2290Schedule1sRequest>
  </e:Body>
</e:Envelope>
```



```

    <!-- wsse:Security omitted from example, similar to SendSubmissions-->
  </e:Header>
  <e:Body>
    <ns10:Get2290Schedule1sResponse
      xmlns:ns10="http://www.irs.gov/a2a/mef/ETECTransmitterService.xsd" >
      <ns10:Cnt>1</ns10:Cnt>
      <ns10:ETECSubmissionErrorList>
        <ns10:SubmissionError>
          <ns10:SubmissionId>0034562007200seven12</ns10:SubmissionId>
          <ns10:ErrorClassificationDetail>
            <ns10:ErrorMessageTxt>Backend system
error</ns10:ErrorMessageTxt>

          <ns10:ErrorClassificationCd>SYSTEM_ERROR</ns10:ErrorClassificationCd>
            <ns10:ErrorMessageCd>SYSTMIO07500002</ns10:ErrorMessageCd>
          </ns10:ErrorClassificationDetail>
        </ns10:SubmissionError>
      </ns10:ETECSubmissionErrorList>
    </ns10:Get2290Schedule1sResponse>
  </e:Body>
</e:Envelope>

```

4.3.22 Get2290Schedule1s Service Response Message

```

<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Header>
    <ns1:MeFHeader xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
      Id="hdr">
      <ns1:MessageID>00123200703900000034</ns1:MessageID>
      <ns1:Action>Get2290Schedule1</ns1:Action>
      <ns1:MessageTs>2007-01-19T21:27:25.696Z</ns1:MessageTs>
      <ns1:ETIN>00123</ns1:ETIN>
      <ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
      <ns1:TestCd>P</ns1:TestCd>
      <ns1:AppSysID>username</ns1:AppSysID>
      <ns1:WSDLVersionNum>10.8</ns1:WSDLVersionNum>
    </ns1:MeFHeader>
    <!-- wsse:Security omitted from example, similar to SendSubmissions-->
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <Get2290Schedule1Request
      xmlns="http://www.irs.gov/a2a/mef/ETECTransmitterService.xsd">
      <SubmissionId>00123420070380000096</SubmissionId>
    </Get2290Schedule1Request>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```



4.3.23 Get2290Schedule1 Service Request Message

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Header>
    <ns1:MeFHeader xmlns:ns1="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
      Id="hdr">
      <ns1:MessageID>00123200703900000034</ns1:MessageID>
      <ns1:Action>Get2290Schedule1</ns1:Action>
      <ns1:MessageTs>2007-01-19T21:27:25.696Z</ns1:MessageTs>
      <ns1:ETIN>00123</ns1:ETIN>
      <ns1:SessionKeyCd>Y</ns1:SessionKeyCd>
      <ns1:TestCd>P</ns1:TestCd>
      <ns1:AppSysID>username</ns1:AppSysID>
      <ns1:WSDLVersionNum>10.8</ns1:WSDLVersionNum>
    </ns1:MeFHeader>
    <!-- wsse:Security omitted from example, similar to SendSubmissions-->
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <Get2290Schedule1Request
      xmlns="http://www.irs.gov/a2a/mef/ETECTransmitterService.xsd">
      <SubmissionId>00123420070380000096</SubmissionId>
    </Get2290Schedule1Request>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

4.3.24 Get2290Schedule1 Service Response Message

```
<?xml version="1.0" encoding="UTF-8"?>
<e:Envelope xmlns:wn1="http://www.irs.gov/a2a/mef/ETECTransmitterService.xsd"
  xmlns:wn0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:e="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:d="http://www.w3.org/2001/XMLSchema">
  <e:Header>
    <ns0:MeFHeader xmlns:ns0="http://www.irs.gov/a2a/mef/MeFHeader.xsd"
      Id="hdr">
      <ns0:MessageID>00123200703900000034R</ns0:MessageID>
      <ns0:RelatesTo>00123200703900000034</ns0:RelatesTo>
      <ns0:Action>Get2290Schedule1</ns0:Action>
      <ns0:MessageTs>2007-02-08T22:54:36Z</ns0:MessageTs>
      <ns0:ETIN>00123</ns0:ETIN>
      <ns0:SessionKeyCd>Y</ns0:SessionKeyCd>
      <ns0:TestCd>P</ns0:TestCd>
      <ns0:AppSysID>username</ns0:AppSysID>
      <ns0:WSDLVersionNum>10.8</ns0:WSDLVersionNum>
    </ns0:MeFHeader>
    <!-- wsse:Security omitted from example, similar to SendSubmissions-->
  </e:Header>
```



```
<e:Body>
  <ns10: Get2290Schedule1Response
    xmlns:ns10="http://www.irs.gov/a2a/mef/ETECTransmitterService.xsd"/>
</e:Body>
</e:Envelope>
```



5 Fault Responses and Internal-Use Error Messages

This section presents SOAP Fault response and log error message examples for the MeF Web Services.

5.1 Fault Responses Generated by the XML Gateway

The XML Gateway COTS product performs much of the SOAP message validation based on Intrusion Detection and Prevention (IDP) rules configured in the XML Gateway configuration setup. All the error messages generated by the XML gateway when a SOAP request is rejected by the XML Gateway are based on the IDP rules. If the SOAP Web Application Server (SWAS) application throws an exception (SOAP fault) while it is processing the SOAP request, the XML Gateway relays the SOAP fault to the client. The entire communication lifecycle of an XML document is captured in the XML Gateway logs and sent to the remote syslogs that are processed by the Log File Collector. The logs include transmission rejects, exceptions thrown by the SWAS application and all of the alert messages generated by the XML Gateway.

The XML Gateway has a default set of nine IDP rules, and five new custom IDP rules have been configured to meet the requirements to mitigate XML threats as defined in the MeF 3.2 XML Gateway Infrastructure Requirements spreadsheet.

The nine default IDP rules are

1. Authentication Failure IDP Rule (Authentication failed)
2. Authorization Failure IDP Rule (Unauthorized access)
3. Invalid HTTP Message IDP Rule (Document does not match any HTTP filter)
4. Invalid WSDL Message IDP Rule (Document does not match any WSDL message)
5. Large Payload IDP Rule (Maximum payload size)
6. Large XML IDP Rule (Maximum XML document size)
7. No Matching XML IDP Rule (Document does not match any XML filter)
8. Process Error IDP Rule (Document Processing Error)
9. Virus Detected IDP Rule (Virus Found)

The five custom IDP rules are

1. Maximum document count IDP Rule
2. Maximum Archive Recursion IDP Rule (too many levels of zip in attachment)
3. Maximum ZIP Payload size IDP Rule (uncompressed size inside zip)
4. Pattern Match Policy Violation IDP Rule (especially SQL injection)
5. Attempted XML external URI reference IDP Rule

The error messages returned in the **faultstring** tag of the **soap:Fault** are presented below. In several cases, such as **Authentication Failure** and **Virus Detected**, further detail is logged.

For the nine default IDP rules

1. Authentication Failure IDP Rule (Authentication failed):
Authentication Failure
2. Authorization Failure IDP Rule (Unauthorized access):
IDP Rule 'Authorization Failure' aborted processing: nnn
3. Invalid HTTP Message IDP Rule (Document does not match any HTTP filter):
IDP Rule 'Invalid HTTP Message on request' aborted processing: <field>
4. Invalid WSDL Message IDP Rule (Document does not match any WSDL message):
IDP Rule 'Invalid WSDL Message' aborted processing: <field>
5. Large Payload IDP Rule (Maximum payload size):
IDP Rule 'Large Payload on request' aborted processing : <size>



6. Large XML IDP Rule (Maximum XML document size):
IDP Rule 'Max. document size on request' aborted processing : <size>
7. No Matching XML IDP Rule (Document does not match any XML filter):
IDP Rule 'Request document does not match any XML Filter' aborted processing : nnn
8. Process Error IDP Rule (Document Processing Error):
IDP Rule 'MeF Process ErrorIDP Rule' aborted processing : nnn
9. Virus Detected IDP Rule (Virus Found):
IDP Rule 'Virus detected on request' ' aborted processing.

For the five custom IDP rules

1. Maximum document count IDP Rule:
IDP Rule 'Max doc count on request' aborted processing : <max>
2. Maximum Archive Recursion IDP Rule:
IDP Rule 'Maximum archive recursion on request' aborted processing : <max>
3. Maximum ZIP Payload IDP Rule:
IDP Rule 'Max zip payload size on request' aborted processing : <max>
4. Pattern Match IDP Rule:
IDP Rule 'Mef_PatternHit' aborted processing: Pattern Match Policy violation:
5. Attempted XML external URI reference IDP Rule:
IDP Rule 'Attempted XML external URI reference on request' aborted processing : <reference>

An example SOAP for a XML Gateway failure of custom rule #2 (Maximum Element Children IDP Rule) is shown below.

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <soap:Fault>
      <faultcode>soap:Client</faultcode>
      <faultstring xml:lang="en-US">
        IDP Rule 'Max element children on request' aborted processing : 6
      </faultstring>
      <detail>
        <fs:Detail xmlns:fs="http://www.forumsystems.com/2004/04/soap-fault-
detail">
          <fs:User/>
          <fs:Policy>Test4-Max_element_ch-Listener</fs:Policy>
          <fs:WebServiceRule/>
          <fs:Task/>
        </fs:Detail>
      </detail>
    </soap:Fault>
  </soap:Body>
</soap:Envelope>
```

An example of a SOAP fault generated by the XML Gateway default rule #2 (Authorization Failure IDP Rule [Unauthorized access]) for a Logout request when not logged in is shown below.

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <soap:Fault>
      <faultcode>soap:Client</faultcode>
```



```
<faultstring                                xml:lang="en-US">
  IDP Rule 'Authorization Failure' aborted processing: 002
</faultstring>
<detail>
  <fs:Detail xmlns:fs="http://www.forumsystems.com/2004/04/soap-fault-
detail">
    <fs:User/>
    <fs:Policy>Test4-Authorization-Listener</fs:Policy>
    <fs:WebServiceRule/>
    <fs:Task/>
  </fs:Detail>
</detail>
</soap:Fault>
</soap:Body>
</soap:Envelope>
```

A summary of the threats and fault responses generated by the XML Gateway is depicted in Table 5-1, XML Gateway Threats and Fault Responses.



Table 5.1-1: XML Gateway Threats and Fault Responses

Threat	Fault Response Message(s) and Log Error Message(s)
Virus Threats	
Incoming Message With a Virus (XML Gateway)	IDP Rule 'Virus detected on request' ' aborted processing
Outgoing Message With a Virus (XML Gateway)	IDP Rule 'Virus detected on request' ' aborted processing
Incoming Message With an HTTP, XML or SOAP Threats	
With XML parameter tampering	IDP Rule 'Invalid WSDL Message' aborted processing: nnn
With coercive parsing	<ul style="list-style-type: none"> • IDP Rule 'Max. document size on request' aborted processing: nnn • IDP Rule 'Max element children on request' aborted processing: nnn • IDP Rule 'Max element count on request' aborted processing: nnn • IDP Rule 'Max element depth on request' aborted processing: nnn
With recursive payload	<ul style="list-style-type: none"> • IDP Rule 'Max element children on request' aborted processing: nnn • IDP Rule 'Max element depth on request' aborted processing : nnn
With XML routing	IDP Rule 'Attempted XML external URI reference on request' aborted processing: nnn
With SQL injection	IDP Rule 'Process Error' aborted processing : nnn
With malicious content	IDP Rule 'Request document does not match any XML Filter' aborted processing: nnn
With replay attacks	<ul style="list-style-type: none"> • IDP Rule 'Authentication Failure' aborted processing : nnn • IDP Rule 'Authorization Failure' aborted processing: xxx
With external entity attacks	IDP Rule 'Attempted XML external URI reference on request' aborted processing: nnn
With text based attacks (not required for MeF)	IDP Rule 'Request document does not match any XML Filter' aborted processing : nnn
With WSDL format check errors	IDP Rule 'Invalid WSDL Message' aborted processing: nnn
With WSDL security validation failures	IDP Rule 'Invalid WSDL Message' aborted processing: nnn
With WSDL format check errors	IDP Rule 'Invalid WSDL Message' aborted processing The root element '<name>' found within the SOAP body does not match the name and namespace of any message defined in the WSDL file.
Incoming Message With Authentication/AuthorizationThreats	
Authentication Errors	Authentication for user <username> failed because 'Invalid credentials'
Authorization Errors	Authorization for user <username> failed because ' Authorization Failure '



5.2 Incoming Message Containing a Virus

Whenever possible, the MSI will include the name of the first virus found and the name of the message part containing the virus in the SOAP fault messages returned to the client and in the error messages posted to the log file. The decision has been made to have the XML Gateway appliance perform the virus checking on the incoming SOAP message attachment. Because the XML Gateway appliance is a COTS product, the format of the SOAP fault messages and log file messages is limited to the capabilities of the appliance.

5.2.1 Example SOAP Fault Response for the Virus Check Performed by the XML Gateway

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <soap:Fault>
      <faultcode>soap:Client</faultcode>
      <faultstring xml:lang="en-US">
        IDP Rule 'Virus detected on request' aborted processing'
      </faultstring>
      <detail>
        <fs:Detail xmlns:fs="http://www.forumsystems.com/2004/04/soap-
fault-detail">
          <fs:User/>
          <fs:Policy>Virus-Detection-Listener</fs:Policy>
          <fs:WebServiceRule/>
          <fs:Task/>
        </fs:Detail>
      </detail>
    </soap:Fault>
  </soap:Body>
</soap:Envelope>
```

5.2.2 Example Error Message for the Virus Check Performed by the XML Gateway

This request contained a virus XML Gateway Detected in part attachment and was rejected.

5.3 Outgoing Message Containing a Virus

Whenever possible, the Modernized System Infrastructure (MSI) will include the name of the first virus found and the name of the message part containing the virus. Any virus detected in an outgoing response will result in an immediate alert to the Computer Security Incident Response Center (CSIRC) and the Electronic Tax Administration (ETA). Also, none of the SOAP Fault, or other error response messages to the Third Party Transmitter, or State clients, will disclose any 6103 data, or other information, that could be used to compromise, or provide harm to, taxpayers, or to IRS systems or processing. Because of these requirements, all outgoing messages will be checked for viruses by the XML Gateway, and the SOAP fault response to the client will simply indicate that the request processing failed, with no details about any virus detected.



5.3.1 Example SOAP Fault Response

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<e:Envelope .....
  <e:Body>
    <e:Fault>
      <faultcode>e:Server</faultcode>
      <faultstring>class ExceptionDetailType:Backend system error
      </faultstring>
      <detail>
        <wn0:ExceptionDetail i:type="wn0:ExceptionDetailType">
          <ErrorMessageTxt i:type="d:string">Backend system error
          </ErrorMessageTxt>
          <ErrorClassificationCd i:type="wn0:ErrorClassificationCdType">
            SYSTEM_ERROR
          </ErrorClassificationCd>
          <ErrorMessageCd
i:type="d:string">SYSTEMInnnnnnnnn</ErrorMessageCd>
          <HostNm i:type="d:string">localhost</HostNm>
        </wn0:ExceptionDetail>
      </detail>
    </e:Fault>
  </e:Body>
</e:Envelope>
```

5.3.2 Example Error Message

This outgoing message contained a virus "name of virus detected" in part "message part/file name" and was rejected.

5.4 Incoming Messages Containing Service Request Argument Errors

This section presents SOAP Fault response and error message examples for incoming requests with errors in the request arguments. None of the SOAP Fault or other error response messages to the Third Party Transmitter or State clients will disclose any 6103 data, or other information, that could be used to compromise, or provide harm to, taxpayers, or to IRS systems or processing. All of the SOAP faults for service request errors follow the same model and indicate an exception category of REQUEST_ERROR.

An example of a SOAP Fault for a GetAck request with an invalid Submission ID is shown below.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<e:Envelope .....
  <e:Body>
    <e:Fault>
      <faultcode>e:Server</faultcode>
      <faultstring>class ExceptionDetailType: Invalid SubmissionId
      </faultstring>
      <detail>
        <wn0:ExceptionDetail i:type="wn0:ExceptionDetailType">
```



```

    <ErrorMessageTxt i:type="d:string">
      Failed to get acknowledgement for message id
      001232006123msg00012. Reason: Could not find
      acknowledgement for submission ID 98792345076929 for this
      ETIN 00123.
    </ErrorMessageTxt>
    <ErrorClassificationCd i:type="wn0:ErrorClassificationCdType">
      REQUEST_ERROR
    </ErrorClassificationCd>
    <ErrorMessageCdi:type="d:string">
      REQSTInnnnnnnnn</ErrorMessageCd>
      <HostNm i:type="d:string">localhost</HostNm>
    </wn0:ErrorExceptionDetaili >
  </detail>
</e:Fault>
</e:Body>
</e:Envelope>

```

A summary of the service requests, argument errors, and fault responses generated by the Web and MeF Application Servers is depicted in Table 5-2, Service Request Argument Errors and Fault Responses.

Table 5.4-1: Service Request Argument Errors and Fault Responses

Request	Fault Response Message(s) and Log Error Message(s)
MSI Services	
EtinRetrieval	Couldn't map a value '<value>' to a java field in a class Log – The value for the test indicator must me a 'T or P' as defined in the schema
EtinStatus	Invalid ETIN <value>
GetStateParticipation (IFA) (MeF System Exception)	Failed to access technical component
GetStateParticipation (IFA) (No participation data found)	Failed to get state participation for message id <msgId>. Reason: No records found in TPDS State Participation.
State Services	
GetNewSubmissions (Invalid ETIN)	Failed to get new submissions for message id <value>. Reason: The request contains an invalid ETIN <value>
GetNewSubmissions (The SubmissionCategory is not one of the registered Submission Categories for the given ETIN)	Failed to get new submissions for message id <msgId>. Reason: ETIN <value> is not registered with submission category <categoryValue>.
GetNewSubmissions (Duplicate message Id)	Failed to get new submissions for message id <msgId>. Duplicate message ID error – another message with same ID already exists.
GetNewSubmissions (Prebuilt file not found)	Failed to get new submissions for message id <msgId> . Reason: The state prebuild zip file mismatch condition. File name not null but zip does not exist. Filename: <filename>
GetSubmissions (Invalid Submission ID)	Failed to get submissions for message id <value>. Reason: Could not find submission for submission ID <value> for ETIN <value>.
GetSubmissions (Invalid maximum request count)	Failed to get submissions for message id <msgId>. Reason: The request count exceeds the maximum configuration count allowed for messageid <msgId>
GetSubmissions (Duplicate message Id)	Failed to get submissions for message id <msgId>. Duplicate message ID error – another message with same ID already exists.



Request	Fault Response Message(s) and Log Error Message(s)
GetSubmission (Invalid Submission ID)	Failed to get submission for message id <msgId>. Reason: Could not find any valid submission with ID: <value> for ETIN <value>.
GetSubmission (No submission found)	Failed to get submission for message id <msgId>.
GetSubmission (Duplicate message Id)	Failed to get submission for message id <msgId>. Duplicate message ID error – another message with same ID already exists.
SendSubmissionReceipts (Invalid Zip Archive)	Failed to send submission receipts for message id <value>. Reason: Unable to process Message with Id: <value> containing submission zip archive with entry name: <value> – data violates rule: X0000-015.
SendSubmissionReceipts (Invalid submission receipt XML)	Failed to send submission receipts for message id <msgId>. Reason: Invalid input submission receipt list document; XML schema validation failed.
SendSubmissionReceipts (Missing receipt element in input XML)	Failed to send submission receipts for message id <msgId>. Reason: Receipt list element does not exist in input receipt list XML document.
SendSubmissionReceipts (Missing count element in input XML)	Failed to send submission receipts for message id <msgId>. Reason: Count element does not exist in input receipt list XML document.
SendSubmissionReceipts (The actual count and receipt count specified in document do not match)	Failed to send submission receipts for message id <msgId>. Reason: Receipt count in the input document was stated as <Value>, but actual number of receipts found in the document was <value>.
SendSubmissionReceipts (The count exceeds max allowed)	Failed to send submission receipts for message id <msgId>. Reason: Receipt count exceeds the maximum allowed.
SendSubmissionReceipts (Missing submission ID or timestamp)	Failed to send submission receipts for message id <msgId>. Reason: No submission ID and/or timestamp element found in one of the state submission receipts.
SendSubmissionReceipts (Invalid timestamp)	Failed to send submission receipts for message id <msgId>. Reason: The timestamp in the receipt for submission ID: <value> is invalid.
SendSubmissionReceipts (Invalid submission or never been retrieved or ETIN is unauthorized)	Failed to send submission receipts for message id <msgId>. Reason: Could not find matching state submission, the state submission has not been previously retrieved, or this ETIN is not authorized to send receipt for submission ID <value>.
SendSubmissionReceipts (duplicate state receipt for same submission)	Failed to send acknowledgements for message id <msgId>. Reason: Duplicate state receipt. State has previously sent in receipt for submission ID <value>.
SendSubmissionReceipts (Duplicate message Id)	Failed to send submission receipts for message id <msgId>. Duplicate message ID error – another message with same ID already exists.
SendSubmissionReceipts (MeF System Exception)	Failed to access third-party component.
SendAcks (Invalid Zip Archive)	Failed to send acknowledgements for message id <value>. Reason: Unable to process Message with Id: <value> containing submission zip archive with entry name: <value> – data violates rule: X0000-015.
SendAcks (Invalid submission receipt XML)	Failed to send acknowledgements for message id <msgId>. Reason: Invalid input submission ack list document; XML schema validation failed.
SendAcks (Missing Acknowledgement list element)	Failed to send acknowledgements for message id <msgId>. Reason: Acknowledgement list element does not exist in input ack list XML document.
SendAcks (Missing count element in input XML)	Failed to send acknowledgements for message id <msgId>. Reason: Count element does not exist in input ack list XML document.



Request	Fault Response Message(s) and Log Error Message(s)
SendAcks (The actual count and acknowledgement count specified in document do not match)	Failed to send acknowledgements for message id <msgld>. Reason: Acknowledgement count in the input document was stated as <Value>, but actual number of acknowledgements found in the document was <value>.
SendAcks (The count exceeds max allowed)	Failed to send acknowledgements for message id <msgld>. Reason: Acknowledgement count exceeds the maximum allowed.
SendAcks (Missing submission ID)	Failed to send acknowledgements for message id <msgld>. Reason: No submission ID element found in one of the state acknowledgements.
SendAcks (Invalid submission or never been retrieved or ETIN is unauthorized)	Failed to send acknowledgements for message id <msgld>. Reason: Could not find matching state submission, the state submission has not been previously retrieved, or this ETIN is not authorized to send acknowledgement for submission ID <value>.
SendAcks (Duplicate state acknowledgement for same submission)	Failed to send acknowledgements for message id <msgld>. Reason: Duplicate state acknowledgement. State has previously sent in acknowledgement for submission ID <value>
SendAcks (MeF System Exception)	Failed to access third-party component.
SendAcks (Duplicate message Id)	Failed to send acknowledgements for message id <msgld>. Duplicate message ID error – another message with same ID already exists.
GetNewAckNotifications (Invalid MaxResults)	Failed to get new acknowledgement notifications for message id <value>. Reason: The request count exceeds the maximum configuration count allowed for messageid <value>
GetNewAckNotifications (Invalid ETIN)	Failed to get new acknowledgement notifications for message id <msgld>. Reason: ETIN <value> is not a valid participating state ETIN.
GetNewAckNotifications (MeF System Exception)	Failed to access technical component.
GetNewAckNotifications (Duplicate message Id)	Failed to get new acknowledgement notifications for message id <msgld>. Duplicate message ID error – another message with same ID already exists.
GetAckNotifications (Invalid Submission ID)	Failed to get acknowledgement notifications for message id <msgld>. Reason: Could not find ack notification for submission ID <value> for ETIN <value>.
GetAckNotifications (Invalid maximum request count)	Failed to get acknowledgement notifications for message id <msgld>. Reason: The request count exceeds the maximum configuration count allowed for messageid <msgld>.
GetAckNotifications (Duplicate message Id)	Failed to get new acknowledgement notifications for message id <msgld>. Duplicate message ID error – another message with same ID already exists.
GetAckNotification (Invalid Submission ID)	Failed to get acknowledgement notification for message id <msgld>. Reason: Could not find any ack notification for submission ID <value>, or this ETIN <value> is not authorized to pick up the notification.
GetAckNotification (Duplicate message ID)	Failed to get new acknowledgement notifications for message id <msgld>. Duplicate message ID error – another message with same ID already exists.
GetSubmissionsByMsgID (Invalid Message ID)	Failed to get submissions by Message ID for message id <msgld>. Reason: The message ID: <msgld> s not valid.
GetSubmissionsByMsgID (Mismatch ETIN)	Failed to get submissions by Message ID for message id <msgld>. Reason: The message ID <msgld> does not belong to the agency ETIN <value>.
GetSubmissionsByMsgID (The message ID not existing)	Failed to get submissions by Message ID for message id <msgld>. Reason: The message ID: <msgld> could not be found.



Request	Fault Response Message(s) and Log Error Message(s)
GetSubmissionsByMsgID (The message has an error response)	Failed to get submissions by Message ID for message id <msgId>. Reason: Cannot retrieve acks associated with message <msgId> because that message had an error response.
GetSubmissionsByMsgID (The message is not a GetNewSubmissions request)	Failed to get submissions by Message ID for message id <msgId>. Reason: Cannot retrieve submissions associated with message <msgId> because that message was not a GetNewSubmissions request.
GetSubmissionsByMsgID (Duplicate message Id)	Failed to get submissions by Message ID for message id <msgId>. for message id <msgId> Duplicate message ID error – another message with same ID already exists.
GetSubmissionReconciliationList (Invalid maximum request count)	Failed to get submission reconciliation list. Reason: Maximum request count <value> is greater than configured count <value>
GetSubmissionReconciliationList (Duplicate message Id)	Failed to get submission reconciliation list. for message id <msgId>. Duplicate message ID error – another message with same ID already exists
GetSubmissionReconciliationList (The SubmissionCategory is not one of the registered Submission Categories for the given ETIN)	Failed to get submission reconciliation list. for message id <msgId>. Reason: ETIN <value> is not registered with submission category <categoryValue>.
Transmitter Services	
SendSubmissions (Invalid Zip Archive)	Failed to send submissions for message id <value>. Reason: Unable to process Message with Id: <value> containing submission zip archive with entry name: <value> – data violates rule: X0000-015.
SendSubmissions (Invalid Submission ID)	Failed to send submissions for message id <value>. Reason: Unable to process Message with Id: <value> containing submission zip archive with entry name: <value> – data violates rule: X0000-015.
SendSubmissions (Invalid ElectronicPostmark)	Failed to send submissions for message id <value>. Reason: Unable to process message with ElectronicPostmark <value> has invalid ElectronicPostmark in SubmissionDataList.
SendSubmissions (Invalid maximum request count)	Failed to send submissions for message id <msgId>. Reason: The request count exceeds the maximum configuration count <value> allowed for messageid <msgId>.
SendSubmissions (Zip File Path is null)	Failed to send submissions for message id <msgId>. Reason: Message <msgId> contains null attachment file path.
SendSubmissions (Cannot find submission by Id)	Failed to send submissions for message id <msgId>. Reason: Error can not find submission object with id <value>.
SendSubmissions (Cannot find build mode)	Failed to send submissions for message id <msgId>. Reason: Error when storing attachment file <FileName>. reason: Mef application properties is not properly set, the build mode should be either set to ATS or PROD.
SendSubmissions (Message that is already processed)	Failed to send submissions for message id <msgId>. Reason: – Attempt to process message <msgId> that has been processed already.
SendSubmissions (MeF System Exception)	Unable to process attachment zip file: <zipFilePath>, <exceptionMessage>.
SendSubmissions (Duplicate message Id)	Failed to send submissions for message id <msgId Duplicate message ID error – another message with same ID already exists.
GetNewAcks (Invalid MaxResults)	Failed to get new acknowledgements for message id <value>. Reason: The request count exceeds the maximum configuration count allowed for messageid <value>.
GetNewAcks (Invalid ETIN)	Failed to get new acknowledgements for message id <value>. Reason: The request contains an invalid ETIN <value>.
GetNewAcks (MeF System Exception)	Failed to access technical component.



Request	Fault Response Message(s) and Log Error Message(s)
GetNewAcks (Duplicate message Id)	Failed to get new acknowledgements for message id <msgld>. Duplicate message ID error – another message with same ID already exists.
GetAcks (Invalid maximum request count)	Failed to get acknowledgements for message id <msgld>. Reason: The request count exceeds the maximum configuration count allowed for messageid <msgld>.
GetAcks (Duplicate message Id)	Failed to get acknowledgement for message id <msgld>. Duplicate message ID error – another message with same ID already exists.
GetAck (No Acknowledgement found)	Failed to get acknowledgement for message id <msgld>.
GetNewSubmissionsStatus (Invalid MaxResults)	Failed to get new submissions status for message id <value>. Reason: The request count exceeds the maximum configuration count allowed for messageid <value>.
GetNewSubmissionsStatus (Invalid ETIN)	Failed to get new submissions status for message id <value>. Reason: The request contains an invalid ETIN <value>.
GetNewSubmissionsStatus (Duplicate message Id)	Failed to get new submissions status for message id <msgld>. Duplicate message ID error – another message with same ID already exists.
GetSubmissionsStatus (Invalid maximum request count)	Failed to get submissions status for message id <msgld>. Reason: The request count exceeds the maximum configuration count allowed for messageid <msgld>.
GetSubmissionsStatus (Duplicate message Id)	Failed to get submissions status for message id <msgld>. for message id <msgld> Duplicate message ID error – another message with same ID already exists.
GetSubmissionStatus (No submission status found)	Failed to get submission status for message id <msgld>.
GetSubmissionStatus (Duplicate message Id)	Failed to get submission status for message id <msgld>. for message id <msgld> Duplicate message ID error – another message with same ID already exists.
GetAcksByMsgID (Invalid Message ID)	Failed to get acknowledgements by message id for message id <msgld>. Reason: The message ID: <msgld> is not valid.
GetAcksByMsgID (Mismatch ETIN)	Failed to get acknowledgements by message id for message id <msgld>. Reason: The message ID <msgld> does not belong to the transmitter <anETIN>.
GetAcksByMsgID (The message ID not existing)	Failed to get acknowledgements by message id for message id <msgld>. Reason: The message ID: <msgld> could not be found.
GetAcksByMsgID (The message has an error response)	Failed to get acknowledgements by message id for message id <msgld>. Reason: Cannot retrieve acks associated with message <msgld> because that message had an error response.
GetAcksByMsgID (The message is not a GetNewAcks request)	Failed to get acknowledgements by message id for message id <msgld>. Reason: Cannot retrieve acks associated with message <msgld> because that message was not a GetNewAcks request.
GetAcksByMsgID (Duplicate message Id)	Failed to get acknowledgements by message id for message id <msgld>. Duplicate message ID error – another message with same ID already exists.
GetNew2290Schedule1s (Invalid maximum request count)	Failed to get new 2290 Schedule1 for message id <msgld>. Reason: The request count exceeds the maximum configuration count allowed for messageid <msgld>.
GetNew2290Schedule1s (MeF System Exception)	Unable to use addZipArchiveEntries to compose the output attachment file for message <msgld>.
Get2290Schedule1s (Invalid maximum request count)	The request count exceeds the maximum configuration count allowed for messageid <msgld>.



Request	Fault Response Message(s) and Log Error Message(s)
Get2290Schedule1s (MeF System Exception)	Unable to use addZipArchiveEntries to compose the output attachment file for message <msgId>.
Get2290Schedule1sByMsgID (MeF System Exception)	Unable to use addZipArchiveEntries to compose the output attachment file for message <msgId>.
Get2290Schedule1sByMsgID (Invalid Message Id)	Message ID: <msgId> is not valid.
Get2290Schedule1sByMsgID (Mismatch ETIN)	Message ID <msgId> does not belong to the transmitter <etin>.
Get2290Schedule1sByMsgID (Message Id does not exist)	Message ID: <msgId> could not be found.
Get2290Schedule1sByMsgID (The message has an error response)	Cannot retrieve 2290 Schedule 1s associated with message <msgId> because that message had an error response.
Get2290Schedule1sByMsgID (The message is not for a GetNew2290Schedule1s message)	Cannot retrieve 2290 Schedule 1s associated with message <msgId> because that message was not a GetNew2290Schedule1s request.

A summary of the service requests, argument errors, and fault responses for account maintenance services generated by the XML Gateway and Web Application Server is depicted in Table 5-3, Service Request Argument Errors and Fault Responses for Account Maintenance Services. Messages indicating IDP rules are generated by the XML Gateway; the others are generated by the Web Application Server after the message has passed XML Gateway scrutiny.

Table 5.4-2: Service Request Argument Errors and Fault Responses

Request	Fault Response Message(s)	Log Error Message(s)
MSI Account Maintenance Services		
Login	Authentication for user <username> failed because 'Invalid credentials' The Timestamp in the Request message Header is not within the allowable range of current time.	Authentication for user <username> failed because 'Invalid credentials'. The Incoming Request Timestamp is outside the range allowed by Request_Msg_Timestamp_Interval.
Logout	IDP Rule 'Authorization Failure' aborted processing: Not Logged In	Not authorized for the service request.

5.4.1 Service Request Processing Errors

Any of the service requests can result in a processing error. Table 5-4, Service Request Processing Errors and Fault Responses, illustrates the fault responses and log messages for the service request processing errors. Table 5-5, Error Codes, Log Messages, Fault Responses, and Fault Types, shows the Error Codes, Log Messages, Fault Responses, and Fault Types.

If a service request contains an incorrect length MessageID, or if it does not contain a MessageID in the MeFHeader, then this presents a unique error situation. If the fault response simply appended an “E” to the request MessageID, it would also contain an incorrect MessageID. However, the MessageID in the fault response must be a valid format. Therefore, for this unique situation, the fault response MessageID is generated by the MeF service processing software. The fault response MessageID will be in the form “00000ccyydddxxxxxxxE” and the RelatesTo element will contain the incorrect



MessageID from the request. Example fault response MeFHeader elements for a MessageID too short situation are shown below.

```
<MeFHeader>  
  <MessageID>00000200610813588700E</MessageID>  
  <RelatesTo>799582006087023</RelatesTo>  
  <Action>EtinStatus</Action>  
  <MessageTs>2006-04-18T17:28:58.871Z</MessageTs>  
  <ETIN>00123</ETIN>  
  <SessionKeyCd>N</SessionKeyCd>  
  <TestCd>T</TestCd>  
</MeFHeader>
```



Table 5.4-3: Service Request Processing Errors and Fault Responses

Processing Error	Fault Response Message(s)	Log Error Message(s)
Session timeout expiration	Session timeout. – SYSTEM_ERROR	XML Gateway. Session timeout error.
Session limit was exceeded	Session limit exceeded. – REQUEST_ERROR	Session or service connection limit reached
Invalid Session key error	Invalid session.	XML Gateway. Session error. Invalid Session key error
Data container is not set	Data error – REQUEST_ERROR	SOAP Dispatcher transmission error. Data container has not been set.
Name of the data element is null or empty	Data element empty – REQUEST_ERROR	SOAP Dispatcher transmission error. Name of the data element is null or empty
File not found.	File not found – SYSTEM_ERROR	File not found
Unsupported file format	File not supported – REQUEST_ERROR	Unsupported file format
Virus found in file sent from client	IDP Rule 'Virus detected on request' ' aborted processing	This request contained a virus XML Gateway Detected in part attachment and was rejected
File IO Error	Encountered problem processing file – SYSTEM_ERROR	System File IO Error
File transfer error	ftp error – SYSTEM_ERROR	File transfer error
Required ETIN value not found in SOAP header	Required ETIN value not found in SOAP header. – REQUEST_ERROR	ETIN not found in SOAP header
Required Test Indicator value not found in SOAP header	Required Test Indicator value not found in SOAP header. – REQUEST_ERROR	Test Indicator not found in SOAP header
Required Session Indicator value not found in SOAP header	Required Session Indicator value not found in SOAP header. – REQUEST_ERROR	Session Indicator not found in SOAP header
SOAP Header 'Action' element does not match Service URI	Invalid Action element value – REQUEST_ERROR	SOAP Header 'Action' element does not match Service URI
SOAP Header 'Action' element is missing	Action element not found – REQUEST_ERROR	SOAP Header 'Action' element is missing
Backend threw SOAP Transfer Exception	Backend error – SYSTEM_ERROR	Backend threw SOAP Transfer Exception
Error storing SOAP information	Error storing SOAP information – SYSTEM_ERROR	DataHolder problem, need to resolve within the code on service side
SOAP Exception	SOAP Exception – SYSTEM_ERROR	System SOAP exception. See log file for details
Expected response from backend not found	Backend error – SYSTEM_ERROR	Expected response from backend not found or unavailable
Backend System Exception	Backend error – SYSTEM_ERROR	Backend System Exception. See log file for details
Backend Amdas Exception	Backend error – SYSTEM_ERROR	Backend Amdas Exception. See log file for details
Backend EDA Exception	Backend error – SYSTEM_ERROR	Backend EDA Exception. See log file for details



Processing Error	Fault Response Message(s)	Log Error Message(s)
User not found in backend repository	Backend error – SYSTEM_ERROR	User not found in backend repository.
Session or service connection limit exceeded	Session or service connection limit exceeded – REQUEST_ERROR	Session or service connection count is over limit
Database error for session.	System error. Please try again – SYSTEM_ERROR	Error with database for user session
Invalid Action Element Value	SYSTEM_ERROR	The action element value is incorrect or does not match the service request url endpoint
Invalid ETIN for this login name	SYSTEM_ERROR	The Etin in the service request is valid for the system id
File Not Supported	SYSTEM_ERROR Invalid file format	Invalid file format
No Matching XML Filter	IDP Rule; Invalid HTTP message	The incoming request did not match the wsdl's loaded in the xml gateway.
IDP Rule 'Process Error' aborted processing Illegal XML character: 0x3	IDP RULE: Process Error	The incoming request contains invalid xml character



Table 5.4-4: Error Codes, Log Messages, Fault Responses, and Fault Types

Error Code	Log Error Message	Fault Response	Error Type
REQSTISD6500001	SOAP Dispatcher transmission error. Data container has not been set	Data error	REQUEST_ERROR
REQSTISD6500002	SOAP Dispatcher transmission error. Name of the data element is null or empty	Data element empty	REQUEST_ERROR
SYSTMI003000001	File not found from backend	File not found from backend	SYSTEM_ERROR
REQSTI003000002	Unsupported file format	File not supported	REQUEST_ERROR
REQSTI003000005	The required Site Affinity dc cookie is missing from the HTTP header	The required Site Affinity dc cookie is missing from the HTTP header	REQUEST_ERROR
REQSTI003000004	Invalid SOAP message format: NNN	Invalid SOAP message format: NNN	REQUEST_ERROR
REQSTI007000001	Virus(es) found in file sent from client	Virus found in file, cannot process request	SECURITY_ERROR
SYSTMI003000002	System FileIO Error	Encountered problem processing file	SYSTEM_ERROR
SYSTMI005500001	File transfer error	File transfer error	SYSTEM_ERROR
REQSTI004000001	ETIN not found in SOAP header	Required ETIN value not found in SOAP header	REQUEST_ERROR
REQSTI004000002	Test Indicator not found in SOAP header	Required Test Indicator value not found in SOAP header	REQUEST_ERROR
REQSTI004000003	Session Indicator not found in SOAP header	Required Session Indicator value not found in SOAP header	REQUEST_ERROR
REQSTI004000004	SOAP Header 'Action' element does not match Service URI	Invalid Action element value	REQUEST_ERROR
REQSTI004000005	SOAP Header 'Action' element is missing	Action element not found	REQUEST_ERROR
REQSTI004000006	ETIN found in SOAP header could not be validated against customer ID	Invalid ETIN for this login name	REQUEST_ERROR
SYSTMI004000001	Backend threw SOAP Transfer Exception	Backend SOAP error	SYSTEM_ERROR
SYSTMI004000002	DataHolder problem; need to resolve within the code on service side	Error storing SOAP information	SYSTEM_ERROR
SYSTMI004000003	System SOAP exception. See log file for details	SOAP Exception	SYSTEM_ERROR
SYSTMI007500001	Expected response from backend not found or unavailable	Backend error	SYSTEM_ERROR
SYSTMI007500002	Backend System Exception. See log file for details	Backend system error	SYSTEM_ERROR
SYSTMIAM7500001	Backend Amdas Exception. See log file for details	Backend AMDAS error	SYSTEM_ERROR



Error Code	Log Error Message	Fault Response	Error Type
SYSTMIED7500001	Backend EDA Exception. See log file for details	Backend EDA error	REQUEST_ERROR
SYSTMIO07500003	User not found in backend repository.	System could not find user	SYSTEM_ERROR
SESSNIMF0000001	Session count is over limit	Session limit reached	REQUEST_ERROR
SESSNIMF0000002	Service count is over limit for SAML session	Service count is over limit for SAML session	REQUEST_ERROR
SESSNIMF0000003	SAML session not found in Session Database for this user. Check site affinity	SAML session not found on this site for this user. Check site affinity dc= cookie and retry	REQUEST_ERROR
SYSTMIO02000001	Error with database for user session. See exception message.	Session database error	SYSTEM_ERROR
REQSTIO04000007	Username not found in the SOAP header	Invalid or unknown username	REQUEST_ERROR
REQSTIO04000008	The Message ID in SOAP Header was not long enough per requirements	Incorrect length for Message ID	REQUEST_ERROR
REQSTIO04000009	The Message ID in SOAP Header contained an unexpected indicator	Message ID indicator should not be applied	REQUEST_ERROR
REQSTIO04000010	The Message ID in SOAP Header contained an indicator other than 'R' or 'E'	Message ID indicator is not valid	REQUEST_ERROR
REQSTIO04000011	The Message ID in SOAP Header did not have valid numbers for the ETIN and/or date	Unable to parse ETIN + date combination as a valid number in Message ID	REQUEST_ERROR
REQSTIO04000012	The date in the Message ID in SOAP Header could not be associated to a real date	Invalid date in Message ID	REQUEST_ERROR
REQSTIO04000013	Except for the message indicator; if any; the Message ID in SOAP header did not contain all lower case characters	Message ID does not contain all lower case characters	REQUEST_ERROR
REQSTIO04000014	The Message ID in SOAP Header contains non-alphanumeric (numbers and letters) characters	Found non-alphanumeric characters in Message ID	REQUEST_ERROR
REQSTIO04000015	Could not find Message ID in the SOAP Header	Message ID not found	REQUEST_ERROR
REQSTIO03000003	Did not find any attachment(s) in the client's request	Could not find expected attachment in request	REQUEST_ERROR
REQSTIO03000006	MeFHeader element not found in the SOAP header	MeFHeader element not found in the SOAP header	REQUEST_ERROR
REQSTIO03000007	WS Security element not found in SOAP header.",	WS Security element not found in SOAP header.",	REQUEST_ERROR
REQSTIO03000008	SOAP Body not found in the SOAP message.",	SOAP Body not found in the SOAP message.",	REQUEST_ERROR



Error Code	Log Error Message	Fault Response	Error Type
SYSTMIO06500001	Unknown System Exception. See log file or stack trace for details	Internal system error	SYSTEM_ERROR
REQSTIO04000017	Deserialization error on incoming SOAP Message; most likely due to unknown element in message body	Element not defined in the schema	REQUEST_ERROR
REQSTIO04000018	ETIN in header does not match Message ID	ETIN in MeF SOAP Header does not match the ETIN in the Message ID	REQUEST_ERROR
REQSTIO04000019	Test Indicator in SOAP header in not valid	Test-Indicator value found in SOAP header is not valid	REQUEST_ERROR
REQSTIO04000020	Username does not match with UID in message		REQUEST_ERROR
REQSTIO04000021	UID does not match with AppSysId in SOAP header		REQUEST_ERROR
REQSTIO04000022	UID not found inflow message		REQUEST_ERROR
REQSTIO04000023	CUSTID not found inflow message		REQUEST_ERROR
REQSTIO06500001	The Incoming Request Timestamp is outside the range allowed by Request_Msg_Timestamp_Interval.	The Timestamp in the Request message Header is not within the allowable range of current time.	REQUEST_ERROR
ISS_SOAP_FAULT	Failed to exchange message or attachment with main application	RunReceiver Secure FTP/Tectia error	SYSTEM_ERROR
ETINEIO07500001	Problem with retrieving ETIN from backend	ETIN retrieval error	SYSTEM_ERROR
SYSTMP2B7500001	Miscellaneous TLS communications error	P2B call error	SYSTEM_ERROR
SYSTMP2B7500002	P2B - Peer not verifiable		SYSTEM_ERROR
SYSTMP2B7500003	P2B - Bad key for certificate		SYSTEM_ERROR
SYSTMP2B7500004	P2B - TLS Handshake error		SYSTEM_ERROR
SYSTMP2B7500005	P2B - Connection refused		SYSTEM_ERROR
SYSTMP2B7500006	P2B - Connection timeout		SYSTEM_ERROR
SYSTMP2B7500007	P2B - Server not found		SYSTEM_ERROR
SYSTMP2B7500010	Miscellaneous P2B communications error		SYSTEM_ERROR
SYSTMIO07500005	Backend MeF Systems are not available now.		SYSTEM_ERROR



Error Code	Log Error Message	Fault Response	Error Type
SYSTMIO07500006	No response from SendSubmissions processor.		SYSTEM_ERROR
SYSTMIO07500007	Error during SendSubmissions processing.		SYSTEM_ERROR

5.5 Acknowledgements for Duplicate Messages and Submissions

Starting with Release 10.5, transmitters will get receipts for SendSubmissions requests before they are checked for being duplicates against an official repository. The detection of such duplicates, whether of Message ID or of Submission ID, will be reported in the Acknowledgement data structure in response to one of the Get requests for Acknowledgements, rather than as a SOAP fault. An acknowledgement created as a result of such a detection can be identified by the presence of either of the two following values in the RuleNum field within ValidationErrorGrp: T0000-014 (duplicate submission ID) and T0000-113 (submission within duplicate message).

An acknowledgement created as the result of a detected duplicate (a “duplicate ack”) may be sent before, with, or after an acknowledgement created for the actual processing of a return. A duplicate ack should not be considered as reflecting the final disposition of a given submission ID.



Glossary

Transport

<u>HTTP</u>	(Hypertext Transfer Protocol) An application-level protocol for distributed communication between Internet systems
<u>HTTPS</u>	(Secure HTTP) A protocol that offers the secure delivery of HTTP packets between the server and client using SSL
<u>SSL</u>	(Secure Sockets Layer) A secure session protocol that guarantees message privacy and message integrity
<u>MIME</u>	(Multimedia Internet Message Extensions) A message format used to encapsulate binary payloads into SOAP messages
<u>DIME</u>	(Direct Internet Message Encapsulation) A lightweight, binary message format designed to encapsulate one or more application-defined payloads into a single message construct
<u>MTOM</u>	(Message Transmission Optimization Mechanism) A lightweight, binary message format where attachments are streamed as binary data within a MIME message

Packaging and Extensions

<u>SOAP</u>	(Simple Object Access Protocol) A lightweight XML-based protocol for exchange of information in a decentralized, distributed environment;
<u>WS-Security</u>	(Web Services Security Language) A SOAP extension that enables secure interactions between Web Services providing credential exchange, message integrity, and message confidentiality;
<u>WS-Addressing</u>	(Web Services Addressing) A SOAP extension that provides transport-neutral mechanisms to address Web Services and messages.

Description

<u>WSDL</u>	(Web Services Description Language) An XML format for the description of network services as a set of endpoints operating on messages containing either document-oriented or procedure-oriented information;
<u>XML</u>	(Extensible Markup Language) A simple, flexible text format used in the exchange of a wide variety of data on the Web and elsewhere.
<u>CA</u>	(Certificate Authority) An entity which issues digital certificates for use by other parties like VeriSign or IdenTrust.
<u>RSA</u>	(Rivest, Sharmir and Adelman [algorithm]) An algorithm for public-key encryption
<u>DSA</u>	(Digital Signature Algorithm) A United States Federal Government standard for digital signatures.



Abbreviations and Acronyms

A2A	Application-to-Application
AppSysID	Application System ID
CA	certificate authority
CLIN	Contract Line Item Number
CSIRC	Computer Security Incident Response Center
CSRA	Merger of CSC and SRA International
DIME	Direct Internet Message Encapsulation
DSA	Digital Signal Algorithm
EFIN	Electronic Filer Identification Number
EIN	Employer Identification Number
EO	Exempt Organization
ETA	Electronic Tax Administration
ETEC	Excise Tax e-File and Compliance
HTTP	HyperText Transfer Protocol
HTTPS	HTTP Secure
ICD	Interface Control Document
ID	identification
IDP	Intrusion Detection and Prevention
IFA	Internet Filing Application
IRM	Internal Revenue Manual
IRS	Internal Revenue Service
ISS	Infrastructure Shared Services
MeF	Modernized e-File
MIME	Multimedia Internet Message Extensions
MSI	Modernized System Infrastructure
MTOM	Message Transmission Optimization Mechanism
R	Release
RSA	Rivest, Sharmir and Adelman [algorithm]
RUP	Registered User Portal
SAML	Security Assertion Markup Language
SOAP	Simple Object Access Protocol
SSL	Secure Sockets Layer
SSN	Social Security Number
SWAS	SOAP Web Application Server
TO	Task Order
WS	Web Services
WS-Security	Web Services Security
WSDL	Web Services Description Language
XML	eXtensible Markup Language

