Centers for Medicare & Medicaid Services (CMS)
7500 Security Boulevard
Baltimore, MD 21244-1850

HIPAA Eligibility Transaction System (HETS)

Submitter SOAP/MIME Connectivity Instructions

Version: 11.0

Last Modified: June 2026

Document Number: SYS-9005

Table of Contents

List of Tables

List of Figures

1. Introduction

This document provides information on how to connect to the Health Insurance Portability and Accountability Act (HIPAA) Eligibility Transaction System (HETS) 270/271 application using support of Simple Object Access Protocol + Web Services Description Language envelope standards (SOAP+WSDL) and support of Hypertext Transfer Protocol/Multipurpose Internet Mail Extensions (HTTP/MIME) Multi-part envelope standards. The SOAP and MIME protocols are offered in addition to the Centers for Medicare & Medicaid Services (CMS) Extranet connection. HETS Trading Partners will have the option to use one of the available connection methods to submit and receive eligibility data. The HETS 270/271 application will continue to accept only real-time transactions.

The Department of Health and Human Services (HHS) has designated the Council for Affordable Quality Healthcare/Committee on Operating Rules for Information Exchange (CAQH/CORE) as the authoring entity for the Operating Rules mandated under the Patient Protection and Affordable Care Act (ACA). The HETS 270/271 follows the federally mandated Phase I CORE 153: Eligibility and Benefits Connectivity Rule and the Phase II CORE 270: Connectivity Rule.

Specifically, HETS 270/271:

- Supports SOAP/MIME protocol and associated errors
- Requires Trading Partners transmitting with SOAP or MIME to obtain a digital certificate and send the transaction to the HETS 270/271 application via a secure internet connection
- Requires Trading Partners transmitting with SOAP or MIME to include the X-Forwarded-For HTTP header in all their 270/271 requests
- Requires Trading Partners to maintain annual renewal requirements of the Trading Partner Agreements (TPA) and SOAP or MIME digital certificate

This document is intended for use by a technical professional with experience implementing secure web-based connectivity.

The HETS 270/271 application authenticates the Trading Partner via a unique HETS 270/271 Submitter ID and ensures the Trading Partner is associated with valid National Provider IDs (NPIs) in the HETS database. If the Trading Partner is not authorized or is not associated with valid NPIs, then the appropriate X12 error response is returned.

Please refer to the HETS Companion Guide found in the 'Downloads' section on the HETS Help website for the errors returned in the above situations.
A Web Submitter ID indicates the Submitter has been set up to submit 270 requests to the HETS 270/271 application using SOAP or MIME. Before submitting a 270 request to the HETS 270/271 application, the Submitter must ensure that all providers have completed the HETS EDI Enrollment with your organization's Unique ID. The user manual for the HDT application is available on the CMS.gov website.

Figure 1: HETS 270/271 Communication Process illustrates the high-level process for communicating with the HETS 270/271 application. Each lock icon represents a system checkpoint that must be passed before eligibility information is returned on the 271 response.

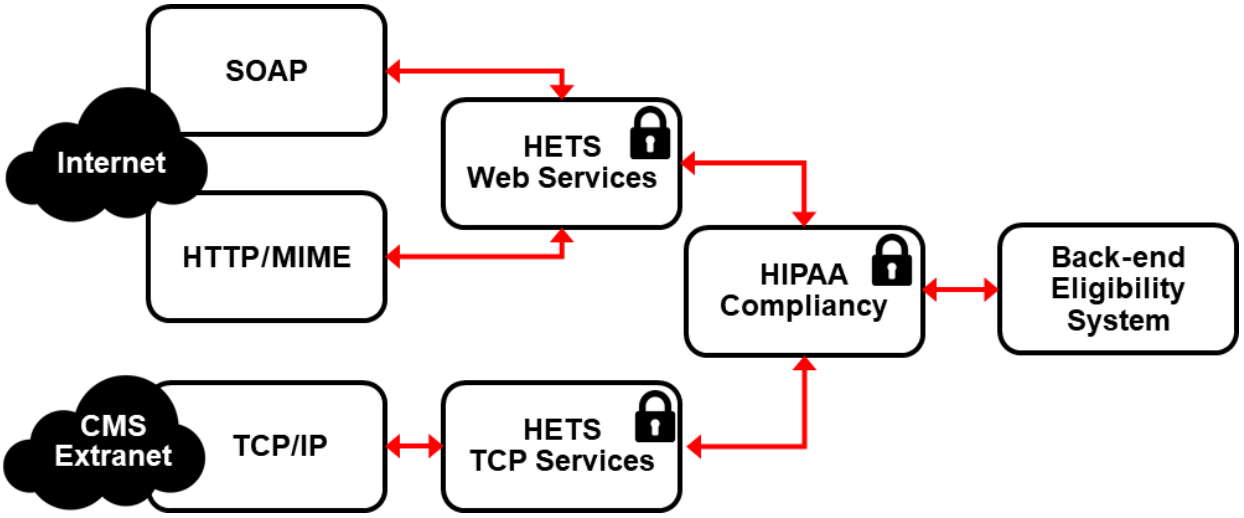

Figure 1: HETS 270/271 Communication Process

2. Authentication & Authorization Handling

To connect to the HETS 270/271 application via SOAP or MIME, Trading Partners shall authenticate using an X.509 Digital Certificate, employing the Transport Layer Security (TLS) 1.2 open standard for client certificate-based authentication. TLS 1.2 is required for compliance with the federally mandated National Institute of Standards and Technology (NIST) Special Publication NIST SP 800-52 Rev. 2. The Trading Partner's connecting IP address will be verified by CMS before the 270 inquiry is routed to the HETS 270/271 application. The Trading Partner's connecting IP address must be an address from the organization's Production (not testing) environment. The supplied Trading Partner connecting IP address must be a public address and listed on the Trading Partner's Trading Partner Agreement (TPA).

2.1 X.509 Digital Certificates

The information provided in the following steps should enable the Trading Partners to locate the appropriate digital certificates for HETS connectivity. Trading Partners will need to generate a Certificate Signing Request (CSR) to obtain the digital certificate for their organization. The CSR generation process is platform-specific. Please review the CSR generation process for your Certificate Authority (CA) carefully, as outlined in the links provided in the following three subsections, and contact the CA directly to obtain the digital certificate. CMS requires that all Trading Partners using SOAP or MIME use a SHA2-256 digital certificate.

The Trading Partners will need to procure a digital certificate from one of the CAs listed in sections 2.1.1 through 2.1.2 to enable their infrastructure to connect to the HETS servers. Information on certificate procurement and platform-specific CSR generation processes can be found on each CA's webpage. Links to their home pages are provided in sections 2.1.1 through 2.1.2.

The digital certificate obtained by the Trading Partner must be provided to CMS in advance by contacting the Medicare Customer Assistance Regarding Eligibility (MCARE) Help Desk during the onboarding process.

MCARE Contact Information:
Monday to Friday 7:00 am to 7:00 pm ET
1-866-324-7315
MCARE@cms.hhs.gov

MCARE will verify the digital certificate and the HETS Trading Partner Agreement and initiate the process to properly configure Trading Partner access to the HETS system. The same digital certificate is also required to digitally sign the SOAP message's timestamp and payload fields, as specified in Section 3.1.1. The SOAP response will also be digitally signed by CMS to verify message authenticity.

Trading Partners that acquire a new Digital Certificate for HETS 270/271 SOAP or MIME MUST provide a copy of the new Digital Certificate to CMS by contacting MCARE. The Trading Partner will also be required to complete an updated HETS Trading Partner Agreement that includes the new Digital Certificate details. To ensure an uninterrupted transition, CMS requires that

Trading Partners begin this process at least 30 days prior to the expiration of the existing Digital Certificate.

2.1.1 DigiCert

Information on digital certificates is provided by DigiCert. HETS Submitters must procure a new DigiCert certificate and must choose one of the options in Section 2.1.1.1. Some HETS Submitters may currently be using the older certificate types specified in Section 2.1.1.2; Submitters will not be able to renew these certificates after the current certificate expires.2.1.1.1 DigiCert - Intermediate Certificates (Currently Supported for New or Renewing HETS Submitters)

Digital certificates issued by the following DigiCert Intermediate certificates (and their root) are accepted:

- DigiCert Assured G2 SMIME RSA4096 SHA384 2024 CA1
- DigiCert Assured ID SMIME RSA2048 SHA256 2021 CA1

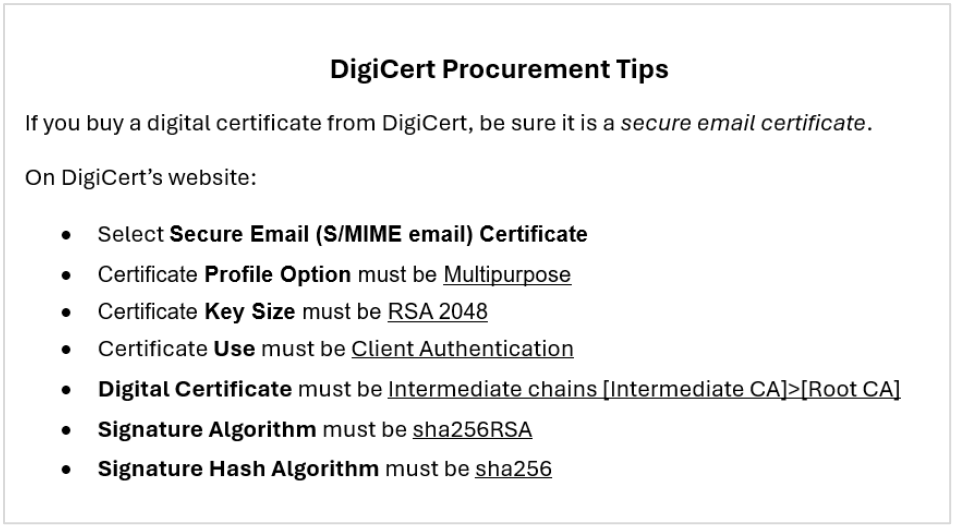

Figure 2: DigiCert Procurement Tips

2.1.1.2 DigiCert – Legacy Intermediate Certificates (which cannot be renewed)

The following DigiCert Intermediate certificates will be accepted until the HETS Submitter's current intermediate certificate expires. No new versions of these certificates will be accepted:

- DigiCert SHA2 Assured ID CA
- DigiCert SHA2 Secure Server CA
- DigiCert EV RSA CA G21
- DigiCert SHA2 High Assurance Server CA
- DigiCert Assured ID CA G2
- DigiCert Global CA G2

2.1.2 IdenTrust

Information on digital certificates provided by IdenTrust is available on their website.

Trading Partners should ensure they obtain an Organization Validated (OV) digital certificate from IdenTrust.

Digital certificates issued by the following IdenTrust Intermediate certificates are accepted:

- Intermediate CA: TrustID Server CA O1

2.2 Overall HETS Web Services Security Policy

HETS Web Services Security Policy assertions use both transport-level and message-level security bindings. The information provided for Transport Level Security applies to SOAP and MIME requests. The information provided for Message Level Security applies only to SOAP.

Transport Level Security (Transport Binding) – SOAP and MIME

- Create an SSL connection using an RSA 2048-bit certificate
- CMS requires TLSv1.2 and supports the following cipher suites:
  - TLS_RSA_WITH_AES_128_CBC_SHA
  - TLS_RSA_WITH_AES_256_CBC_SHA
  - TLS_RSA_WITH_AES_128_GCM_SHA256
  - TLS_RSA_WITH_AES_256_GCM_SHA384
  - TLS_RSA_WITH_AES_128_CBC_SHA256
  - TLS_RSA_WITH_AES_256_CBC_SHA256

Message Level Security (Asymmetric Binding) – SOAP ONLY

- Digitally sign the timestamp and payload using an RSA-SHA256 signature algorithm.
- Include a Binary Security Token inside the Web Services Security Header.
- Include a TimeStamp node in the Web Services Security Header.

3. SOAP

The HETS 270/271 application supports transactions formatted according to SOAP Version 1.2, conforming to the WSDL standards for Extensible Markup Language (XML) envelope formatting, submission, and retrieval. The X12 payload data MUST be embedded using the inline method (Character Data (CDATA) element), the XML schema, and WSDL definitions formatted according to Phase II CORE 270: Connectivity Rule. The following key resources should be used as a reference:

- SOAP XML Schema
- WSDL Schema
- Phase II CORE 270: Connectivity Rule

These resources are available for download via the following website:

CAQH CORE – Committee on Operating Rules for Information Exchange

HETS 270/271 Submitters connecting via SOAP will need to use a specific URL to access HETS. Please contact the MCARE Help Desk to obtain the URL.

3.1 SOAP Data Requirements

Submitters should specify appropriate SOAP headers. SOAP specifications are precise and require that the headers and body be constructed perfectly. Any incorrectly constructed SOAP headers will fail and result in an error.

SOAP Header

The SOAP Header must include the timestamp element, which must be digitally signed. The Web Services Security Binary Security Token must be added to the SOAP Header to verify the signature. The CORE Connectivity Rule referenced in section 3 should be used as a reference when constructing the SOAP Header.

SOAP Body

The W3C Recommendation link should be used as a reference when constructing the SOAP Body.

Only those characters referenced in the Basic and Extended Character Sets noted in the Appendix of the ASCX12 270/271 version 005010X279A1 TR3, including the 005010X279E1 Errata, are acceptable within a HETS 270 inquiry.

Table 1: Required Body Elements for 270 Requests Using SOAP and *Table 2: Required Body Elements for X12 Responses Using SOAP* describe the required HETS-specific body elements for 270 SOAP requests and X12 responses.

Table 1: Required Body Elements for 270 Requests Using SOAP

Element Name	Description
PayloadType	X12_270_Request_005010X279A1
ProcessingMode	RealTime
PayloadID	Refer to Section 4.4.2 of Phase II CORE 270: Connectivity Rule for structural guidelines for CORE envelope metadata.
TimeStamp	Format is YYYY-MM-DDTHH:MMSSZ. Refer to http://www.w3.org/TR/xmlschema11-2/ for more information. SenderID
SenderID	This is a Submitter-defined alphanumeric field. The value must be 10 characters in length. Recommended value is your HETS 270/271 SOAP Submitter ID plus trailing zeros for a total of 10 characters.
ReceiverID	CMS
CORERuleVersion	2.2.0
Payload	X12 request. This element must be digitally signed, and the entire payload should be enclosed within a CDATA tag.

Table 2: Required Body Elements for X12 Responses Using SOAP

Element Name	Description
PayloadType	X12_271_Response_005010X279A1, X12_TA1_Response_00501X231A1, X12_999_Response_005010X231A1
ProcessingMode	RealTime
PayloadID	Refer to Section 4.4.2 of Phase II CORE 270: Connectivity Rule for structural guidelines for CORE envelope metadata.
TimeStamp	Format is YYYY-MM-DDTHH:MMSSZ. Refer to http://www.w3.org/TR/xmlschema11-2/ for more information.
SenderID	CMS
ReceiverID	This field must have 10 characters in length, the same as the 270 Sender ID.
CORERuleVersion	2.2.0
Payload	X12 response

3.1.1 SOAP Digital Signature

The SOAP communication protocol requires Trading Partners to embed their certificate within the eligibility request and digitally sign the SOAP Body Payload and SOAP Header Timestamp using their private key. CMS will embed its certificate in the 271 response, enabling the

Trading Partner to verify that it came from CMS. Trading Partners can obtain a copy of CMS's Certificate in advance by contacting the MCARE Help Desk.

Trading Partners sending via SOAP must use the Exclusive Without Comments canonicalization algorithm for signature. Signatures using algorithms that are Exclusive With Comments, Inclusive With Comments, or Inclusive Without Comments will *not* be accepted.

Refer to the SOAP Security Extensions: Digital Signature link for details on digital signatures in SOAP.

3.2 SOAP Examples

Examples of a real-time SOAP request and response can be found in Sections 4.2.2.3 and 4.2.2.4 of the CORE Phase II Connectivity Rule (link to that Rule available in section 3 of this document).

Table 3: SOAP Request Message Structure provides an example of a 270 request in SOAP format. Carriage returns should NOT be used in the SOAP Body Payload field. They appear in the example information in the HETS Companion Guide for readability purposes only. Also, it is important that the Content-Type line of the HTTP Header and the namespace declaration in the Envelope begin tag contain values associated with SOAP 1.2, as shown below. Using SOAP 1.1 values or different values may cause the SOAP message to be rejected by HETS.

Note: The example below is for illustrative purposes only. All the variable data will be unique per transaction and should not be copied verbatim and sent to HETS. Lastly, it is highly recommended that the encoding Style attribute for the Envelope begin tag not be specified.

Table 3: SOAP Request Message Structure

SOAP Structure Element	Content
HTTP Header	POST https://soap.hetsp-haa.cms.gov HTTP/1.1 Accept-Encoding: gzip,deflate Content-Type: application/soap+xml;charset="UTF-8";action="RealTimeTransaction" Content-Length: 4808 Host: soap.hetsp-haa.cms.gov Connection: Keep-Alive User-Agent: Apache-HttpClient/4.1.1 (java 1.5)
SOAP Envelope Begin	<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
SOAP Header Begin	<soap:Header>
SOAP Header Web Services Security	<wsse:Security soap:mustUnderstand="true" xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">

SOAP Structure Element	Content
SOAP Header TIMESTAMP	<wsu:Timestamp wsu:Id="id-155"> <wsu:Created> yyyy-MM-dd'T'hh:mm:ss'Z'</wsu:Created> <wsu:Expires> yyyy-MM-dd'T'hh:mm:ss'Z'</wsu:Expires> </wsu:Timestamp>
SOAP Header Binary Security Token	<wsse:BinarySecurityToken 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="X509-0E4E74F95B0421C31C135515946875040">{{{BASE-64 Encoded Certificate}}} </wsse:BinarySecurityToken>
SOAP Header Signature	<ds:Signature Id="SIG-44" 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-43"> <ds:Transforms> <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"> <InclusiveNamespaces PrefixList="ns1 soap" xmlns="http://www.w3.org/2001/10/xml-exc-c14n#"/> </ds:Transform> </ds:Transforms> <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/> <ds:DigestValue>cKtVDws5KS70zUTfNB90jcz/F5K/GwIiDF09aEV2fMA=</ds:DigestValue> </ds:Reference> <ds:Reference URI="#id-155"> <ds:Transforms> <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"> <InclusiveNamespaces PrefixList="ns1 soap" xmlns="http://www.w3.org/2001/10/xml-exc-c14n#"/> </ds:Transform> </ds:Transforms> <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/> <ds:DigestValue>tu65ngGe0dl2f2f3iwN/phOQBDXEPFVw2u6/1ZKmX/A=</ds:DigestValue> </ds:Reference> </ds:SignedInfo>

SOAP Structure Element	Content
SOAP Header Signature Value	<ds:SignatureValue>{{{Encoded Signature Value}}} </ds:SignatureValue> Note: The digest of the timestamp + payload is the final string that should be digitally signed to obtain the final signature.
SOAP Header KeyInfo	<ds:KeyInfo Id="KI-0E4E74F95B0421C31C135515946875041"> <wsse:SecurityTokenReference wsu:Id="STR0E4E74F95B0421C31C135515946875042"> <wsse:Reference URI="#X509-0E4E74F95B0421C31C135515946875040" ValueType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0#X509v3"/> </wsse:SecurityTokenReference> </ds:KeyInfo>
SOAP Header End	</ds:Signature> </wsse:Security> </soap:Header>
SOAP Body Begin	<soap:Body> <ns1:COREEnvelopeRealTimeRequest xmlns:ns1="http://www.caqh.org/SOAP/WSDL/CORERule2.2.0.xsd">
SOAP Body PayloadType	<PayloadType>X12_270_Request_005010X279A1</PayloadType>
SOAP Body ProcessingMode	<ProcessingMode>RealTime</ProcessingMode>
SOAP Body PayloadID	<PayloadID> d5cf23d4-240d-1d9e-b7d5-ab0f8185296b</PayloadID>
SOAP Body TimeStamp	<TimeStamp> yyyy-MM-ddThh:mm:ssZ</TimeStamp>
SOAP Body SenderID	<SenderID>ABCDEFGHIJ</SenderID>
SOAP Body ReceiverID	<ReceiverID>CMS</ReceiverID>
SOAP Body CORERuleVersion	<CORERuleVersion>2.2.0</CORERuleVersion>
SOAP Body Payload	<Payload wsu:Id="id-43"xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"><![CDATA[The 270 request will appear here beginning with the ISA segment and ending with the IEA segment as shown in the example from Appendix A of the HETS Companion Guide located on the HETSHelp site - *https://www.cms.gov/Research-Statistics-Data-and-Systems/CMS-Information-Technology/HETSHelp/Index.html*]]></Payload>
SOAP Body End	</ns1:COREEnvelopeRealTimeRequest> </soap:Body>

SOAP Structure Element	Content
SOAP Envelope End	</soap:Envelope>

Table 4: SOAP Response Message Structure provides an example of a 271 response in SOAP format. Carriage returns should NOT be used in the SOAP Body Payload field. They appear in the example information in the HETS Companion Guide for readability purposes only.

Table 4: SOAP Response Message Structure

SOAP Structure Element	Content
HTTP Header	HTTP/1.1 200 OK Server: Apache-Coyote/1.1 Content-Length: 4430 Content-Type: application/soap+xml Date: Mon, 27 Jan 2020 15:45:25 GMT Content-Type: application/soap+xml
SOAP Envelope Begin	<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
SOAP Header Begin	<soap:Header>
SOAP Header Web Services Security	<wsse:Security soap:mustUnderstand="true" xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
SOAP Header TIMESTAMP	<wsu:Timestamp wsu:Id="id-155"> <wsu:Created>2020-01-27T15:45:25Z</wsu:Created> <wsu:Expires>2020-01-27T15:46:25Z</wsu:Expires></wsu:Timestamp>
SOAP Header Binary Security Token	<wsse:BinarySecurityToken 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="X509-0E4E74F95B0421C31C135515946875040">{{{BASE-64 Encoded Certificate}}} </wsse:BinarySecurityToken>

SOAP Structure Element	Content
SOAP Header Signature	<ds:Signature Id="SIG-44" 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-168"> <ds:Transforms> <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"> <InclusiveNamespaces PrefixList="ns1 soap" xmlns="http://www.w3.org/2001/10/xml-exc-c14n#"/> </ds:Transform> </ds:Transforms> <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256 "/> <ds:DigestValue>cKtVDws5KS70zUTfNB90jcz/F5K/GwIiDF09aEV2fMA=</ds:DigestValue> </ds:Reference> <ds:Reference URI="#id-155"> <ds:Transforms> <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"> <InclusiveNamespaces PrefixList="ns1 soap" xmlns="http://www.w3.org/2001/10/xml-exc-c14n#"/> </ds:Transform> </ds:Transforms> <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256 "/> <ds:DigestValue>tu65ngGe0dl2f2f3iwN/phOQBDXEPFVw2u6/1ZKmX/A=</ds:DigestValue> </ds:Reference> </ds:SignedInfo>
SOAP Header Signature Value	<ds:SignatureValue>{{{Encoded Signature Value }}} </ds:SignatureValue>
SOAP Header KeyInfo	<ds:KeyInfo Id="KI-0E4E74F95B0421C31C135515946875041"> <wsse:SecurityTokenReference wsu:Id="STR0E4E74F95B0421C31C135515946875042"> <wsse:Reference URI="#X509-0E4E74F95B0421C31C135515946875040" ValueType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0#X509v3"/> </wsse:SecurityTokenReference> </ds:KeyInfo>

SOAP Structure Element	Content
SOAP Header End	</ds:Signature> </wsse:Security> </soap:Header>
SOAP Body Begin	<soap:Body> <ns1: COREEnvelopeRealTimeResponse xmlns:ns1="http://www.caqh.org/SOAP/WSDL/CORERule2.2.0.xsd">
SOAP Body PayloadType	<PayloadType> X12_271_Response_005010X279A1</PayloadType>
SOAP Body ProcessingMode	<ProcessingMode>RealTime</ProcessingMode>
SOAP Body PayloadID	<PayloadID> d5cf23d4-240d-1d9e-b7d5-ab0f8185296b </PayloadID>
SOAP Body TimeStamp	<TimeStamp> yyyy-MM-dd'T'hh:mm:ss'Z'</TimeStamp>
SOAP Body SenderID	<SenderID>CMS</SenderID>
SOAP Body ReceiverID	<ReceiverID>ABCDEFGHIJ</ReceiverID>
SOAP Body CORERuleVersion	<CORERuleVersion>2.2.0</CORERuleVersion>
SOAP Body Payload	<Payload wsu:Id="id-168 " xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd""><![CDATA[The 271 response will appear here beginning with the ISA segment and ending with the IEA segment as shown in the example from Appendix B of the HETS Companion Guide located on the HETSHelp site - https://www.cms.gov/Research-Statistics-Data-and-Systems/CMS-Information-Technology/HETSHelp/index]]></Payload>
SOAP Body End	</ns1: COREEnvelopeRealTimeResponse> <ErrorCode>Success</ErrorCode> <ErrorMessage/> </soap:Body>

4. MIME

HETS will support standard MIME messages. The MIME format used MUST be multipart/form-data.

Only those characters referenced in the Basic and Extended Character Sets noted in the Appendix of the ASCX12 270/271 version 005010X279A1 TR3, including the 005010X279E1 Errata, are acceptable within a HETS 270 inquiry.

CORE does not mandate a naming convention. HETS will implement the MIME body parts with the same field names as the SOAP element nodes. The response will be returned as a MIME multipart/form-data message, with the Payload body part containing the X12 response.

HETS 270/271 Submitters connecting via MIME will need to use a specific URL to access HETS. Please contact the MCARE Help Desk to obtain the URL.

4.1 MIME Data Requirements

Submitters must specify appropriate MIME headers. The MIME specification is very precise and requires that the headers and the body be constructed perfectly. Any incorrectly constructed MIME headers will fail and result in an error.

The HETS implementation of MIME allows the use of the Basic and Extended Character Sets, as noted in the Appendix of the ASCX12 270/271 version 005010X279A1 TR3, including only the 005010X279E1 Errata. Please refer to the Request for Comments (RFC) 2388 – returning values from Forms: multipart/form-data to review header and body specifications.

MIME Header

MIME Messages will have standard HTTP header data elements, such as POST, HOST, Content-Length, and Content-Type. The supported Content-Type is MIME multipart/form-data.

MIME Body

Required HETS-specific body elements for 270 requests and X12 responses using MIME are defined in *Table 5: Required Body Elements for 270 Requests Using MIME* and *Table 6: Required Body Elements for X12 Responses Using MIME*.

Table 5: Required Body Elements for 270 Requests Using MIME

Element Name	Description
PayloadType	X12_270_Request_005010X279A1
ProcessingMode	RealTime
PayloadID	Refer to Section 4.4.2 of Phase II CORE 270: Connectivity Rule for structural guidelines for CORE envelope metadata.

[2] Effective April 23, 2022. MIME Submitters should not utilize this URL before that date.

Element Name	Description
TimeStamp	Format is YYYY-MM-DDTHH:MMSSZ. Refer to http://www.w3.org/TR/xmlschema11-2/ for more information.
SenderID	This is a Submitter-defined alphanumeric field. The value must be ten characters in length. Recommended value is your HETS 270/271 MIME Submitter ID plus trailing zeros for a total of ten characters.
ReceiverID	CMS
CORERuleVersion	2.2.0
Payload	X12 request. The X12 request must be submitted as part of the MIME request and not as an attachment. If an attachment is received, the transaction will be rejected. The request does not need to be enclosed within a CDATA tag. See Appendix A of the HETS Companion Guide located on the HETS Help site for an example of the data.

Table 6: Required Body Elements for X12 Responses Using MIME

Element Name	Description
PayloadType	X12_271_Response_005010X279A1, X12_TA1_Response_00501X231A1, X12_999_Response_005010X231A1
ProcessingMode	RealTime
PayloadID	Refer to Section 4.4.2 of Phase II CORE 270: Connectivity Rule for structural guidelines for CORE envelope metadata.
TimeStamp	Format is YYYY-MM-DDTHH:MMSSZ. Refer to http://www.w3.org/TR/xmlschema11-2/ for more information.
SenderID	CMS
ReceiverID	This field must be 10 characters in length. The same as the 270 Sender ID.
CORERuleVersion	2.2.0
Payload	X12 response

4.2 MIME Examples

Examples of a real-time MIME request and response can be found in Sections 4.2.1.1 and 4.2.1.2 of the CORE Phase II Connectivity Rule (link to that Rule available in Section 3 of this document).

MIME Data Requirements for Header and Body:

Refer to *Table 7: MIME Request Message Structure* in this document for the HETS-specific body elements.

MIME Request and Response Examples:

Table 7: MIME Request Message Structure and Table 8: MIME Response Message Structure provide examples of a 270 request and a 271 response using HTTP MIME Multipart. The following examples are for illustrative purposes only. The 270 request must be submitted as part of the MIME request and not as an attachment. If an attachment is received, the transaction will be rejected. The request does not need to be enclosed within a CDATA tag. All the variable data will be unique per transaction and should not be copied verbatim and sent to HETS.

Table 7: MIME Request Message Structure

MIME Structure Element	Content
MIME Header	POST https://mime.hetsp-haa.cms.gov HTTP/1.1 Connection: keep-alive Content-Length: 1392 Content-Type: multipart/form-data; boundary=COSZiva9NdnYzPXUEGy-tLBO8n4-czud Host: mime.hetsp-haa.cms.gov User-Agent: Apache-HttpClient/4.2.1 (java 1.5)
MIME Body	--COSZiva9NdnYzPXUEGy-tLBO8n4-czud Content-disposition: form-data; name="PayloadType" X12_270_Request_005010X279A1 --COSZiva9NdnYzPXUEGy-tLBO8n4-czud Content-disposition: form-data; name="ProcessingMode" RealTime --COSZiva9NdnYzPXUEGy-tLBO8n4-czud Content-disposition: form-data; name="PayloadID" d5cf23d4-240d-1d9e-b7d5-ab0f8185296b --COSZiva9NdnYzPXUEGy-tLBO8n4-czud Content-disposition: form-data; name="TimeStamp" 2020-02-25T19:50:40.611Z --COSZiva9NdnYzPXUEGy-tLBO8n4-czud Content-disposition: form-data; name="SenderID" HETS00001 --COSZiva9NdnYzPXUEGy-tLBO8n4-czud Content-disposition: form-data; name="ReceiverID" CMS --COSZiva9NdnYzPXUEGy-tLBO8n4-czud Content-disposition: form-data; name="CORERuleVersion"

MIME Structure Element	Content
	2.2.0 --COSZiva9NdnYzPXUEGy-tLBO8n4-czud Content-disposition: form-data; name="Payload" ***The 270 request will appear here beginning with the ISA segment and ending with the IEA segment as shown in the example from Appendix A of the HETS Companion Guide located on the HETSHelp site - https://www.cms.gov/Research-Statistics-Data-and-Systems/CMS-Information-Technology/HETSHelp/index *** --COSZiva9NdnYzPXUEGy-tLBO8n4-czud--

Table 8: MIME Response Message Structure

MIME Structure Element	Content
MIME Header	HTTP/1.1 200 OK Server: Apache-Coyote/1.1 Content-Type: multipart/form-data; boundary=7aaeaf96-1e54-4567-a8d0- e93de77cd66a Content-Length: 1567 Date: Mon, 27 Jan 2020 15:45:25 GMT
MIME Body	--7aaeaf96-1e54-4567-a8d0-e93de77cd66a Content-disposition: form-data; name="PayloadType" Content-type: text/plain X12_TA1_Response_005010X279A1 --7aaeaf96-1e54-4567-a8d0-e93de77cd66a Content-disposition: form-data; name="ProcessingMode" Content-type: text/plain RealTime --7aaeaf96-1e54-4567-a8d0-e93de77cd66a Content-disposition: form-data; name="PayloadID" Content-type: text/plain d5cf23d4-240d-1d9e-b7d5-ab0f8185296b --7aaeaf96-1e54-4567-a8d0-e93de77cd66a Content-disposition: form-data; name="TimeStamp" Content-type: text/plain 2020-02-25T19:50:40.611Z --7aaeaf96-1e54-4567-a8d0-e93de77cd66a Content-disposition: form-data; name="SenderID" Content-type: text/plain

MIME Structure Element	Content
MIME Body	CMS --7aaeaf96-1e54-4567-a8d0-e93de77cd66a Content-disposition: form-data; name="ReceiverID" Content-type: text/plain HETS000001 --7aaeaf96-1e54-4567-a8d0-e93de77cd66a Content-disposition: form-data; name="CORERuleVersion" Content-type: text/plain 2.2.0 --7aaeaf96-1e54-4567-a8d0-e93de77cd66a Content-disposition: form-data; name="Payload" Content-type: text/plain ***The 271 response will appear here beginning with the ISA segment and ending with the IEA segment as shown in the example from Appendix B of the HETS Companion Guide located on the HETSHelp site https://www.cms.gov/Research-Statistics-Data-and-Systems/CMS-Information-Technology/HETSHelp/index *** --7aaeaf96-1e54-4567-a8d0-e93de77cd66a Content-disposition: form-data; name="ErrorCode" Content-type: text/plain Success --7aaeaf96-1e54-4567-a8d0-e93de77cd66a Content-disposition: form-data; name="ErrorMessage" Content-type: text/plain --7aaeaf96-1e54-4567-a8d0-e93de77cd66a—

5. Common Error Processing for SOAP & MIME

The HETS 270/271 application will process SOAP and MIME transactions and return errors as described in this section.

5.1 HTTP Status & Error Codes

The processing and error codes for the HTTP layer are defined in the HTTP specifications.

The intended use of these status and error codes in processing transactions is specified in Table 4.3.3.1 of Phase II CORE 270: Connectivity Rule, referenced in section 3.

5.2 CORE Envelope Processing Status & Error Codes

Table 9: Envelope Process Status and Errors describes envelope processing status and error codes specific to the HETS 270/271 application for SOAP and MIME transactions.

Table 9: Envelope Process Status and Errors

Element Name	Description
<FieldName>Illegal	Illegal value provided for <FieldName>.
<FieldName>Required	The field <FieldName> is required but was not provided.
VersionMismatch	The CORERuleVersion sent is not acceptable to the Receiver.
Success	The envelope was processed successfully.

5.3 SOAP Specific Processing Errors

Table 10: SOAP Specific Processing Errors describes examples of SOAP processing errors.

Table 10: SOAP Specific Processing Errors

Element Name	Description
UnAuthorized	The signature could not be verified.

5.4 SOAP & MIME Transaction (X12) Error Processing

Refer to the HETS Companion Guide for additional information on the transaction processing errors that will be returned as a SOAP message or MIME Multipart/form-data containing the related response.

The HETS Companion Guide is available in the downloads section of the CMS HETS Help website.

6. X-Forwarded-For HTTP Header Requirement

Required Action
To enhance the protection of Personally Identifiable Information (PII) and Protected Health Information (PHI), improve transaction monitoring capabilities, and strengthen security protocols, the Centers for Medicare & Medicaid Services (CMS) requires SOAP/MIME Submitters to include the originating IP address along with the network hops that a HETS 270 eligibility request traverses prior to receipt by the HETS system.

Implementation Requirements
When transmitting eligibility transactions, Submitters must utilize the standard X-Forwarded-For HTTP header element to include the network IP addresses from the point of origin through receipt by the HETS system for each request.

Submitter Responsibilities
To ensure compliance with the HETS RoB, organizations must include the X-Forwarded-For HTTP header element in each 270 SOAP/MIME request sent to HETS, incorporating IP address details beginning with the Originating IP and documenting the network hops the transaction has traversed.

If the 270 request originates outside your organization, you should coordinate with your providers and any third-party organizations to ensure they include the Network IP details in the X-Forwarded-For header to maintain complete traceability of the request to its origin. If a request originates from or traverses an offshore worksite, the TPA should include at least one IP address for that worksite. The accompanying graphic illustrates the request process aligned with CMS business objectives.

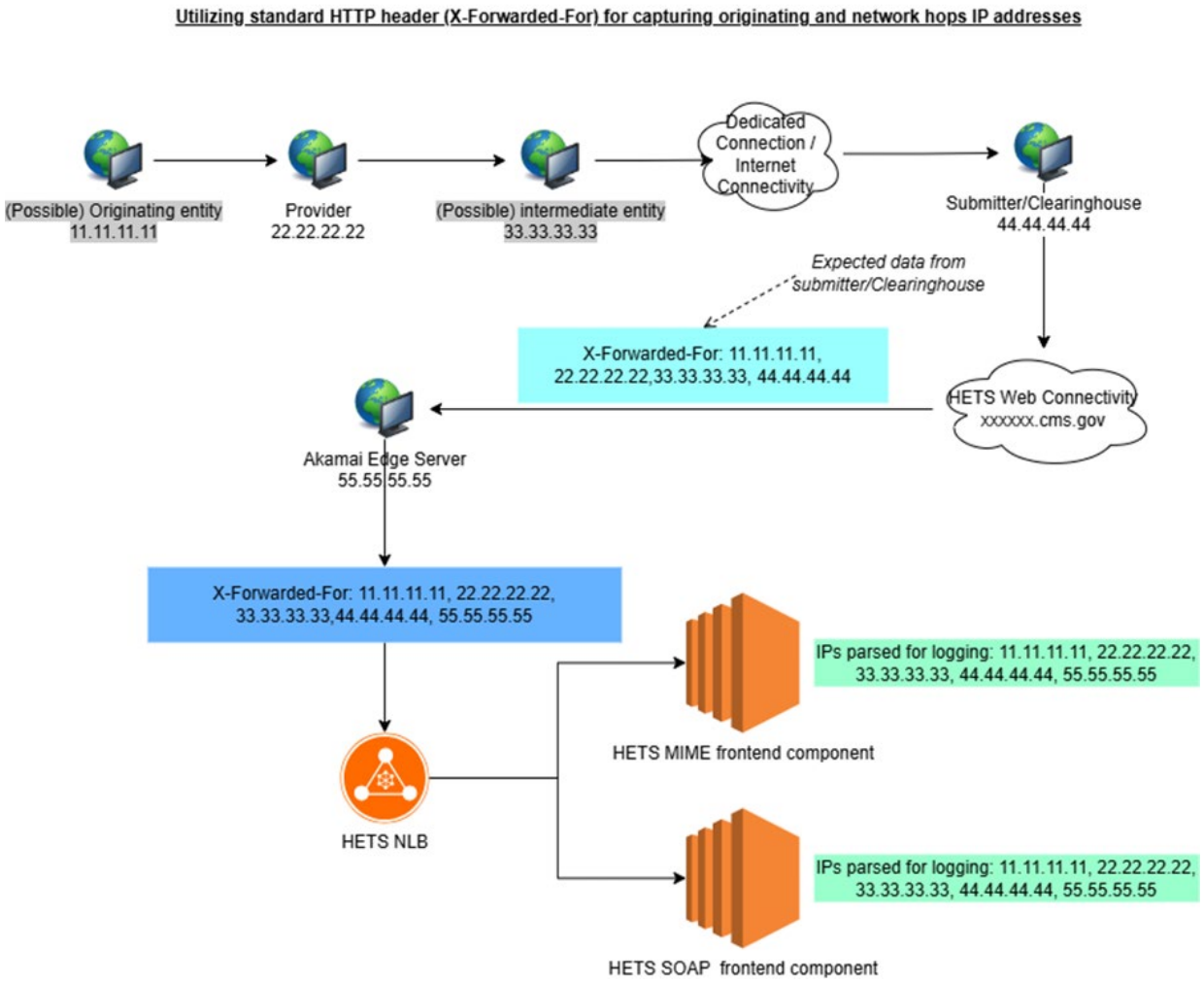

Figure 3: HTTP Header (X-Forwarded-For) Capturing Originating and Network Hops IP Addresses

7. General Onboarding Checklist

If the Trading Partner is a new HETS Submitter, they must first follow the traditional enrollment processes on the "How to Get Connected – HETS 270/271" page of the HETS Help website and complete the Trading Partner Agreement. It will take approximately two weeks to complete this process. If the Trading Partner already has a HETS Submitter ID (SID) or has just completed the traditional enrollment process, the following steps serve as a general guide to onboarding for SOAP/MIME submissions. It will take approximately two weeks to complete this process.

Table 11: General Onboarding Checklist

Checkbox	General Onboarding Checklist Items
☐	When the Trading Partner contacts MCARE to request access to SOAP/MIME, they must have already purchased an X.509 Digital Certificate and be prepared to provide the following information: • Organizational Legal Business Name • Organization Submitter ID (SID) if previously assigned • Organization connecting IP address(es) that will be linked to the certificate • X.509 Digital Certificate Issuer Name • X.509 Digital Certificate Type • X.509 Digital Certificate Serial Number
☐	The Trading Partner should email the X.509 Digital Certificate to MCARE in (.PEM) format to MCARE at MCARE@cms.hhs.gov. The Trading Partner should NOT include the private key when sending the digital certificate.
☐	MCARE will review the digital certificate. If there are any issues or errors, MCARE will notify the Trading Partner and assist in resolving them.
☐	Upon validation of the Digital Certificate, MCARE will work with the HETS team to provide access to the Trading Partner.
☐	Once access has been provided, MCARE will inform the Trading Partner and work with them to verify that transactions can be sent successfully.
☐	After successfully implementing HETS via SOAP or MIME (i.e., sending a good 270 request and receiving a proper 271 response), the Trading Partner's Submitter ID status will be moved from 'Test' to 'Production'. The Trading Partner may then send regular Medicare eligibility traffic to HETS.

Appendix A: HETS Web Services Security Policy

The following text is an example of the XML Schema.

```
<?xml version="1.0" encoding="utf-8"?>
<wsp:Policy xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"
  xmlns:sp="http://docs.oasis-open.org/ws-sx/ws- securitypolicy/200702"
  xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-
1.0.xsd">
  <wsp:ExactlyOne>
    <wsp:All>
      <wsp:Policy wsu:Id="transport-ssl-client-cert">
        <sp:TransportBinding>
          <wsp:Policy>
            <sp:TransportToken>
              <wsp:Policy>
                <sp:HttpsToken>
                  <wsp:Policy>
                    <sp:RequireClientCertificate/>
                  </wsp:Policy>
                </sp:HttpsToken>
                <sp:HttpsToken RequestClientCertificate="true"/>
              </wsp:Policy>
            </sp:TransportToken>
            <sp:AlgorithmSuite>
              <wsp:Policy>
                <wsp:ExactlyOne>
                  <sp:Basic256Sha256/>
                </wsp:ExactlyOne>
              </wsp:Policy>
            </sp:AlgorithmSuite>
            <sp:IncludeTimestamp/>
          </wsp:Policy>
        </sp:TransportBinding>
      </wsp:Policy>
```

```
<sp:AsymmetricBinding>
 <wsp:Policy>
  <sp:RecipientSignatureToken>
   <wsp:Policy>
    <sp:X509Token
     sp:IncludeToken="http://docs.oasis-open.org/ws-sx/ws-
securitypolicy/200702/IncludeToken/AlwaysToInitiator">
     <wsp:Policy>
      <sp:WssX509V3Token10/>
     </wsp:Policy>
    </sp:X509Token>
   </wsp:Policy>
  </sp:RecipientSignatureToken>
  <sp:InitiatorSignatureToken>
   <wsp:Policy>
    <sp:X509Token
     sp:IncludeToken="http://docs.oasis-open.org/ws-sx/ws-
securitypolicy/200702/IncludeToken/AlwaysToRecipient">
     <wsp:Policy>
      <sp:WssX509V3Token10/>
     </wsp:Policy>
    </sp:X509Token>
   </wsp:Policy>
  </sp:InitiatorSignatureToken>
  <sp:AlgorithmSuite>
   <wsp:Policy>
    <sp:Basic256Sha256/>
   </wsp:Policy>
  </sp:AlgorithmSuite>
  <sp:Layout>
   <wsp:Policy>
    <sp:strict/>
   </wsp:Policy>
  </sp:Layout>
```

```
        <sp:IncludeTimestamp/>
      </wsp:Policy>
    </sp:AsymmetricBinding>
    <sp:EndorsingSupportingTokens
      xmlns:sp="http://docs.oasis-open.org/ws-sx/ws-securitypolicy/200702">
      <wsp:Policy>
        <sp:X509Token
          sp:IncludeToken="http://docs.oasis-open.org/ws-sx/ws-
securitypolicy/200702/IncludeToken/AlwaysToRecipient">
          <wsp:Policy>
            <sp:WssX509V3Token10/>
          </wsp:Policy>
        </sp:X509Token>
      </wsp:Policy>
      <wsp:Policy>
        <sp:X509Token
          sp:IncludeToken="http://docs.oasis-open.org/ws-sx/ws-
securitypolicy/200702/IncludeToken/AlwaysToInitiator">
          <wsp:Policy>
            <sp:WssX509V3Token10/>
          </wsp:Policy>
        </sp:X509Token>
      </wsp:Policy>
    </sp:EndorsingSupportingTokens>
    <wsp:Policy wsu:Id="request_parts">
      <sp:SignedElements>
<sp:XPath>/*[namespace-uri()='http://www.w3.org/2003/05/soap-envelope' and local-
name()='Envelope']/*[namespace-uri()='http://www.w3.org/2003/05/soap-envelope' and local-
name()='Header']/*[namespace-uri()='http://docs.oasis-open.org/wss/2004/01/oasis-200401-
wss-wssecurity-secext-1.0.xsd' and local-name()='Security']/*[namespace-
uri()='http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd' and
local-name()='Timestamp']</sp:XPath>
<sp:XPath>/*[namespace-uri()='http://www.w3.org/2003/05/soap-envelope' and local-
name()='Envelope']/*[namespace-uri()='http://www.w3.org/2003/05/soap-envelope' and local-
name()='Body']/*[namespace-uri()='http://www.caqh.org/SOAP/WSDL/CORERule2.2.0.xsd' and
local-name()='COREEnvelopeRealTimeRequest']/Payload</sp:XPath>
      </sp:SignedElements>
```

```
    </wsp:Policy>
    <wsp:Policy wsu:Id="response_parts">
      <wsp:ExactlyOne>
        <wsp:All>
 <sp:SignedElements>
  <sp:XPath>/*[namespace-uri()='http://www.w3.org/2003/05/soap-envelope'    and
local-name()='Envelope']/*[namespace-uri()='http://www.w3.org/2003/05/soap-envelope' and
local-name()='Header']/*[namespace-uri()='http://docs.oasis-open.org/wss/2004/01/oasis-
200401-wss-wssecurity-secext-1.0.xsd' and local-name()='Security']/*[namespace-
uri()='http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd' and
local-name()='Timestamp']</sp:XPath>
          <sp:XPath>/*[namespace-uri()='http://www.w3.org/2003/05/soap-envelope'    and
local-name()='Envelope'] /*[namespace-uri()='http://www.w3.org/2003/05/soap-envelope' and
local-name()='Body']/*[namespace-
uri()='http://www.caqh.org/SOAP/WSDL/CORERule2.2.0.xsd' and local-
name()='COREEnvelopeRealTimeResponse']/Payload</sp:XPath>
          </sp:SignedElements>
        </wsp:All>
        <wsp:All>
          <sp:SignedElements>
 <sp:XPath>/*[namespace-uri()='http://www.w3.org/2003/05/soap-envelope'    and
local-name()='Envelope']/*[namespace-uri()='http://www.w3.org/2003/05/soap-envelope' and
local-name()='Header']/*[namespace-uri()='http://docs.oasis-open.org/wss/2004/01/oasis-
200401-wss-wssecurity-secext-1.0.xsd' and local-name()='Security']/*[namespace-
uri()='http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd' and
local-name()='Timestamp']</sp:XPath>
          </sp:SignedElements>
        </wsp:All>
       </wsp:ExactlyOne>
    </wsp:Policy>
    <wsaw:UsingAddressing xmlns:wsaw="http://www.w3.org/2005/08/addressing"/>
   </wsp:All>
  </wsp:ExactlyOne>
</wsp:Policy>
```

Appendix B: Frequently Asked Questions

Table 12: Frequently Asked Questions

Question Number	Question	Answer
1	Do I need my own digital certificate to exchange 270/271 via SOAP and MIME?	Yes. The User ID and Password authentication method are not supported by HETS. Instead, a Trading Partner must procure a digital certificate and configure their system to connect to HETS.
2	Are there specific Digital Certificates that can be used to access HETS?	Sections 2.1.1 through 2.1.2 contain information regarding digital certificate issuance.
3	What specific connectivity configurations must I complete for a successful SOAP connection?	Trading Partners using SOAP are encouraged to ensure the following: The SOAP communication protocol requires Trading Partners to send their certificate and digitally sign the payload and timestamp using their private key. This allows HETS to validate the contents of the received message and when it was sent. The “wsu:ID” attribute is contained in both the timestamp and payload nodes. They should both match the “<Reference URI.” That their perimeter equipment IP range or subnet has been provided to MCARE for configuration within the CMS firewall. Their application uses Public Key Infrastructure (PKI) and configures the Trading Partner keystore with the correct client certificate to sign SOAP messages. The trust store is correctly configured with the CMS certificate.
4	Do I need a Virtual Private Network (VPN) over the internet for connection to the HETS 270/271 Application?	A VPN connection to CMS is not required for connectivity to the HETS 270/271 Application.
5	What is the difference between SOAP and MIME transactions, specific to the HETS 270/271 Application?	From the Trading Partner’s perspective, the HETS 270/271 Application has two different URLs for sending these transactions. The processing for both MIME and SOAP is the same.
6	How do I go about developing my SOAP or MIME client?	HETS does not require any specific tool for client-side implementation. The Trading Partners are free to choose various Commercial-off-the-Shelf (COTS) products or custom code to create the SOAP & MIME requests.

Question Number	Question	Answer
7	How do I wrap a 270 transaction for submission?	For SOAP transactions, the Trading Partners must ensure that the 270 transaction is contained in the payload tag and the “CDATA” tag is present. For MIME transactions, the Trading Partners must ensure that the 270 transaction is contained within the MIME boundary of the payload. MIME does not use CDATA tags, and it should not be present.
8	Can I send more than one 270 in a single SOAP or MIME request?	No. Only one 270 should be submitted per SOAP or MIME request. The HETS 270/271 Application does not support batch.
9	Can I send my transactions as SOAP or MIME attachments?	No. The 270 transactions should be sent as part of the payload tag in SOAP requests. For MIME requests, they should be sent in-line, as part of the payload element.
10	Do I need to use a User ID/Password when establishing a connection to HETS to submit SOAP or MIME transactions?	No. The HETS 270/271 Application connection authentication requirements are based only on digital certificates.
11	Does the SID used in the SOAP message body need to match the X12 SID?	Trading Partners should ensure that the Submitter IDs match. However, the HETS 270/271 Application uses only the SID embedded in the X12 270 transaction for authorization.
12	How can we ensure the digital certificate doesn’t get activated until MCARE validates and authorizes the Submitter?	The certificate will be active the day it is issued to the Trading Partner. However, MCARE will ensure that access to the firewall is allowed only after the certificate verification step is complete.
13	What happens when an organization is revoked by its CA?	The Certificate Revocation Lists for each CA will be loaded into the production environment infrastructure, and Trading Partners that attempt to submit with a revoked digital certificate will be denied access through the CMS firewall.
14	How will the Trading Partner get the WS-Policy, also known as the Web Services Security Policy?	The Trading Partners should receive a copy of the WS-Policy document during the onboarding process. See Section 6.

Question Number	Question	Answer
15	The Submitter is receiving “Error getting response; javax.net.ssl.SSLHandshakeException: Received fatal alert: handshake_failure”. What does that mean?	The 2-way SSL handshake process did not complete successfully. This is most likely due to either the Submitter not having configured 2-way SSL on their end, or to an invalid or revoked digital certificate being used.
16	What types of attachments can be included in a MIME transaction?	No attachments can be included in the MIME transaction. The 270 request must be encoded inline in the MIME message.

Appendix C: References

Table 13: References

Document	Hyperlink
CAQH CORE site for the CORE Connectivity & Operating Rules, SOAP Header, WSDL, and XML Schema details	CAQH CORE – Committee on Operating Rules for Information Exchange
CORE Mandated Operating Rules	https://www.caqh.org/core/operating-rules-mandate
HETS Help website, including the HDT User Guide and HETS Companion Guide	https://www.cms.gov/Research-Statistics-Data-and-Systems/CMS-Information-Technology/HETSHelp/index
HTTP Specifications	https://www.rfc-editor.org/rfc/rfc9110.html
MIME Header & Body Specifications	http://www.faqs.org/rfcs/rfc2388.html
SOAP Body	http://www.w3.org/TR/soap12-part1
Timestamp Element Format	http://www.w3.org/TR/xmlschema11-2/
XML Schemas	http://www.w3.org/TR/SOAP-dsig

Appendix D: Revision History

Table 14: Revision History

Version	Date	Revision/Change Description	Pages Affected
11.0	6/22/2026	TW document review, finalization, baselining, and remediation.	All
10.2	6/10/2026	Removed reference to batch and SID/NPI relationships	6
10.1	4/20/2026	Updated DigiCert Procurement Tips information.	6
10.0	4/16/2026	TW document review, finalization, baselining, and remediation.	All
9.2	4/3/2026	Updated DigiCert content information and procurement process. Updated General Checklist verbiage. Updated X-Forward information.	Multi
9.1	3/30/2026	TW document initial review.	All
9.0	7/23/2025	Removed the Entrust section and any reference to Entrust in the document. Removed the footnote in Section 2.1.1. It's obsolete since HDT only allows the EV RSA CA G2 at this point. Removed "Root CA: IdenTrust Commercial Root CA 1" from 2.1.3. Removed Acronyms and Glossary sections per DMAM's previous instruction in the decision log. Updated TOB to reflect changes.	Page 4, and all with numbering updates
8.0	5/12/2025	Updated section 2.1 to include IdentTrust information, including new section 2.1.3 IdenTrust	3, 4, 5
7.0	11/18/2024	Finalization of the document and remediation.	All
6.1	11/8/2024	Removed SOAP/MIME URLs	Multiple
6.0	11/6/2024	Updated document links, updated release date, finalized	Multiple
5.0	6/28/2024	Updated final Release Date	All
4.8	3/29/2024	Added Note to Table 3 SOAP Request Message Structure under SOAP Header Signature Value	18
4.8	1/9/2024	Updated content in Section 4.1 MIME Data Requirements, advising Submitters that incorrectly constructed MIME headers will receive an error	18
4.8	1/9/2024	Updated content in Section 3.1 SOAP Data Requirements advising Submitters that incorrectly constructed SOAP headers will receive an error	10
4.8	1/9/2024	Updated content in Section 2. Authentication and Authorization Handling advising Submitters that IP addresses must be added to the TPA	7
4.8	1/9/2024	Added a third bullet item in the introduction, reminding Submitters to complete certificate maintenance	5
4.3	11/16/2020	Updated URL references to CAQH CORE to reflect changes on the CAQH website	Multiple

4.2	08/24/2020	Updated CAQH CORE URLs throughout the document to reflect organizational changes to their website Removed previous section 2.1.3, which detailed information for Certification Authority (CA) Symantec. Symantec's certificate business was previously acquired by Digicert.	Multiple
4.1	03/31/2020	Tables 5, 8 and 9 – Update to reflect minor changes in processing in the HETS high availability environment	Multiple

Version	Date	Revision/Change Description	Pages Affected
4.0	09/06/2019	Changes related to the HETS 270/271 High Availability transition include: Section 1 – Updated URL for HETS Companion Guide and HDT User Guide Section 2.1 – Reorganized this Section for clarity Section 2.2 – Removed TLS_RSA_WITH_3DES_EDE_CBC_SHA from the list of supported cipher suites Section 3 – Updated SOAP destination URL Section 3.1.1 – Updated to note canonicalization method algorithm requirements Table 4 – Updated SOAP destination URL and HETS Companion Guide reference URL. Also updated CanonicalizationMethod Algorithm in the SOAP Header Signature example. Table 5 – Updated URL for HETS Companion Guide. Also updated CanonicalizationMethod Algorithm in the SOAP Header Signature example. Table 6 – Updated URL for HETS Companion Guide Table 8 – Updated MIME destination URL and HETS Companion Guide reference URL Table 9 – Updated URL for HETS Companion Guide Section 5.4 – Updated URL for HETS Companion Guide Appendix B – Updated FAQ #16 to clarify that MIME attachments are not accepted Appendix C – Removed extraneous references Appendix D – Added acronyms COTS, PKI & VPN	1, 3, 4, 6, 7, 8, 11, 14, 16, 18, 20, 27, 29, 30
3.2	8/21/2017	Section 1 – Updated URL for HETS Companion Guide Table 4 – Updated URL for HETS Companion Guide Table 5 – Updated URL for HETS Companion Guide Table 6 – Updated URL for HETS Companion Guide Table 8 – Updated URL for HETS Companion Guide Table 9 – Updated URL for HETS Companion Guide Table 15 – Updated Symantec URLs	1, 9, 11, 12, 13, 14, 24
3.1	12/5/2016	Table 8 – Updated MIME Body Content Table 9 – Updated MIME Body Content	13, 14

3.0	3/24/16	Modified document for R2016Q300 Redesign Release. Changes Include: Updated Title Page to remove OIS and 508 compliant check Section 1 – Updated HETS Help website, removed references to HPG and replaced with HDT, added URL for HDT User Manual Section 2 – Removed references to TLS 1.1 and December 31, 2015 deadline to utilize TLS 1.2 and a SHA2-256 certificate as this deadline has passed Section 2.1 – Removed references to December 31, 2015 deadline to utilize TLS 1.2 and a SHA2-256 certificate as this deadline has passed Section 2.1.2 – Removed outdated Certification Authorities for Entrust Section 2.2 – Removed reference to January 1, 2016 requirement as this deadline has passed and replaced SHA1 with SHA2 Section 3 – Updated SOAP URL Section 3.1, Table 2 and 3 description – Updated 271 Responses to X12 Responses Section 3.1, Table 2 and 3 – Updated TimeStamp description Section 3.2, Table 4 – Updated HTTP Header Content with new SOAP URL, and SOAP Header Timestamp Content with generic value, PayloadID Content with an example, TimeStamp Content with generic value Section 4 – Updated MIME URL Section 4.1, Table 6 and 7 description – Updated 271 Responses to X12 Responses Section 4.1, Table 6 and 7 – Updated TimeStamp description Section 4.2 – Added note that examples are for illustrative purposes only Section 4.2, Table 8 – Updated MIME Header Content with new MIME URL, PayloadID Content with an example, TimeStamp Content with generic value Section 4.2, Table 9 – Updated MIME Header and Body Content Section 5.2, Table 10 - Updated VersionMismatch Description and removed InvalidPayload Section 5.2, Table 11 – Added UnAuthorized error and removed other errors Section 5.4 – Removed MIME Specific Processing Errors as they no longer apply Appendix B – Updated FAQ 12 and 16	Multiple

Version	Date	Revision/Change Description	Pages Affected
		Appendix C – Replaced HPG User Guide Reference to HDT User Guide Appendix D – Replaced HPG with HDT	
2.1	9/2/2015	Added Entrust L1K/L1M certificate list and added additional values which can return in the PayloadType field for MIME responses.	4, 12
2.0	06/19/15	Added clarification on requirement of SHA2-256 and Transport Layer Security (TLS) 1.2 requirement as well as clarified existing SOAP/MIME processing.	Multiple
1.1	02/24/2014	Added clarification that only those characters referenced in the Basic and the Extended Character Sets noted in the Appendix of the ASCX12 270/271 version 005010X279A1 TR3 including the 005010X279E1 Errata are acceptable within a HETS 270 inquiry.	5, 10 and 23
1.0	08/15/2013	Initial release.	All