

Report to the Uniform Code Council (UCC)

AS2 Conformance Validation

Final Status

Test Round AS2-2Q01

August 6, 2001

Report to the Uniform Code Council (UCC)

AS2 Conformance Validation

Final Status

Test Round AS2-2Q01

Prepared By:

DRUMMOND GROUP, INC.

www.drummondgroup.com

Test Participants






DRUMMOND GROUP, Inc. is pleased to announce that the following participants in the AS2 Conformance Validation & Interoperability Test 2Q01 have completed all requirements and passed tests (*see Final Test Results*) between each product demonstrating interoperability and conformance to the AS2 document.

To fully understand what completing the test means in the use of the products in production, please read this document carefully.

Workarounds were necessary in some cases and we continue to find immaturity in the interoperability between PKI toolkits. Both of these issues are documented below (*see Interoperability Caveats*).

Sincerely,

Rik Drummond
CEO Drummond Group Inc.

 <p>bTrade Transaction Delivery Networks</p> <p>www.btrade.com</p> <p>Product Name: TDAccess and TDPeer using EDIINT engine, vs 2.0</p>	 <p>Cyclone Commerce</p> <p>www.cyclonecommerce.com</p> <p>Product Name: Cyclone Interchange, vs 4.1</p>
 <p>GXS GE Global eXchange Services</p> <p>www.gxs.ge.com</p> <p>Product Name: GXS AS2 Adapter 2.0</p>	 <p>IPNet Solutions</p> <p>www.ipnetsolutions.com</p> <p>Product Name: IPNet eBizness™ Transact, vs 3.4</p>
 <p>iSoft</p> <p>www.isoft.com</p> <p>Product Name: iSoft Peer-to-Peer Agent, vs 3.0</p>	 <p>SeeBeyond</p> <p>www.seebeyond.com</p> <p>Product Name: SeeBeyond e*Gate Integrator, vs 4.5</p>
 <p>TBSI</p> <p>www.as400ftp.com</p> <p>Product Name: ZMOD Exchange, vs 2.1</p>	 <p>webMethods, Inc.</p> <p>www.webmethods.com</p> <p>Product Name: webMethods Integration Platform vs 4.0</p>

Abstract

This is the second round of testing for [IETF AS2](#). AS2 (Applicability Statement 2) is the draft specification standard (RFC Standards Track) by which vendor applications communicate EDI (EDIFACT or X12), binary or XML data over the Internet. AS2 is an expansion of the AS1 specification (which specifies EDI data transmission over SMTP) to provide for EDI data transmission over HTTP.

The purpose of the test is to provide a venue for vendors to test and correct their software systems in a non-competitive environment. To accomplish this, the systems are put through a series of three Test-Steps, each containing three Test-Groups (or some subset thereof). The Test-Groups are designed to discover and correct weaknesses with vendor systems and to build upon previous tests until full compliance is realized. The first Test-Group is composed of exchanging Security Certificates (*certs*) and basic EDI and XML data. This test is focused primarily on EDI transactions. The data is mostly X12 or EDIFACT but also includes a large (10MB) XML file. The data ranges from small data files to very large (50MB) files. Once it has been established that participants can transfer data successfully, the participants are then asked (in the second Test-Group) to pass the data securely in all combinations of digitally signing and encrypting the data. Finally (in the third Test-Group), the signed/encrypted data is passed with requests for receipts or MDNs (Message Disposition Notification reports) both unsigned and digitally signed MDNs, including non-Repudiation of Receipt (NRR) hashing. As an addendum, some purposely-erroneous data is also exchanged to test the reaction of each system under error conditions.

This test is not just focused on determining who can pass the test, although that is the end result, but on encouraging and promoting interoperability between the participating vendors. By slowly building upon previous tests, proving conformance to the standards, working with other vendors to ensure interoperable features and allowing participants to correct or debug their code during the process, the end result is an entire community of interoperable products. As this paper reports, all the vendors passed the final test (a subset of the entire test suite encompassing the most complex test, which also test simpler tests).

The entire test was conducted over an eleven-week period, the First test-step (debug) was completed in eight weeks. We then took a one-week break to give participants time to recompile code (remove debug features) and reinstall. The Second test-step (Dry-Run) then took one week – to prepare participants for the final test. The Final test-step was conducted the following week.

The Conformance Test

What is the AS2 Compliance Validation Test?

The process for each **Test Round** has three, interrelated Test-Steps. Each **Test-Step** is composed of three Test-Groups (or some subset of these). Each **Test-Group** has a series of **Tests**.

Test Round AS2-4Q00 (4th Quarter 2000)

Test Round AS2-2Q01 (15 May 2001)

Test-Step-1 (Debug Step)

Complete all tests from each Test-Group to allow Code Check & Debug. There are three Test-Groups comprising tests A through L.

Test-Step-2 (Dry Run)

Software installed from scratch following “written” install procedure as it appears in the Product-with-version Installation Manual.

Run some subset of the tests (at least one test from each Test-Group – e.g. Tests A, E, F.3, H.4)

Test-Step-3 (Final Conformance Validation)

Final Verification and official conformance test. One or Two Day Event (no product fixes or code debug activities will be allowed). Successful test completion demonstrates Interoperability to UCC/DGI satisfaction.

Test Round AS2-xQ02 (Date TBD)

*There will be other rounds of testing annually (or semi-annually as required). Continued UCC sanctioned AS2 Interoperability **Conformance Validation** will require each Product-with-Version to participate in future Testing Rounds to retain their conformance rating.*

Each subsequent Test-Round is composed of the same three Test-Steps as described above.

Test-Steps

Test-Step-1 is designed to help companies implement interoperable products by conducting a series of product verification tests. (Test-Step-1 testing is composed of three Test-Groups. Test-Group-1 is MIME-only tests, Test-Group-2 is S/MIME tests and Test-Group-3 is Signed-Receipt tests.) In Test-Step-1, product developers learn and adjust their products as they go. Test-Step-1 testing is conducted periodically – usually at least once a year or when three or more new products become available to establish a new Test-Group.

Test-Groups

Test-Group-1 is composed of Tests A through E. Once a company has completed Test-Group-1, their product-with-version is deemed interoperable with those of their Test-Group-1 Test-Group. Test-Group-1 must be completed before advancing to Test-Group-2.

Test-Group-2 and *Test-Group-3* consists of completing Tests E.1 through E.4 and Tests F through I with all other products that have previously been verified. Again, this must be completed to UCC/DGI's satisfaction. Once this has been accomplished, UCC will verify the product as *AS2 Conformance Validated*. If the code of a previously tested product is changed, the product-with-version must go back through Test-Group-2 & Test-Group-3 testing to ensure interoperability with all the other products.

After a product has completed Test-Group-2 & Test-Group-3 testing in a manner that is satisfactory to UCC, the product-with-version must be commercialized and released within 90 days of completion of Test-Step-3. If the product-with-version is not released within 90 days, *Conformance Validation* will be withdrawn and the product-with-version will be required to complete Test-Group-2 & Test-Group-3 testing again.

Conformance Validation

UCC *Conformance Validation* of a product-with-version will be issued when the following have been completed to UCC/DGI's satisfaction: The product-with-version has passed Test-Steps-3 as defined in this document. This means that each product must exchange information as described in Tests A through I between all products-with-version that have previously passed Test-Step-3 tests.

Test Specifics

To see the text of each of the final tests, please refer to the [Appendix](#).

Interoperability Caveats

Testing Conditions

Interoperability is highly dependant upon test conditions and specifications. Altering or exceeding the conditions under which the test is performed may significantly alter the interoperability results. The three primary impediments to interoperability are Firewalls, Proxy Servers and Certificate Configurations. The former two must be configured locally to allow access to and from the Internet. Certificates should conform to X.509 standards and any system should ignore extensions not understood. However, since most vendor products incorporate security toolkits, it is not entirely within the control of the AS2 software vendors to support all possible certificate fields or extensions. For this reason, certificates should be kept as simple as possible, with one field or Set per Sequence (*see description below*). Creativity is not encouraged when building certs.

Although higher levels of security are available, it was deemed sufficient and prudent to perform the tests using the following certificate attributes:

Client Certificates

- 128/1024 bit encryption
- Triple DES
- SHA1 (preferred) or MD5

Server Certificates

- SSL port 443 (unless otherwise specified)
- Server Side Authentication only
- No Basic Authentication (not necessary although supported in most cases)

Data Types

Testing consisted of transporting a variety of test data types, EDI-X12, EDIFACT and XML of a variety of sizes. The following MIME Content-Types were used:

```
Content-Type: application/EDI-X12
Content-Type: application/EDIFACT
Content-Type: application/XML
Content-Type: application/PKCS7-signature
Content-Type: application/PKCS7-mime
Content-Type: message/disposition-notification
Content-Type: multipart/signed
Content-Type: multipart/report
```

File Size

Most tests were performed with moderate sized EDI-X12 and EDIFACT files. Although very large file size was tested (50MB) this scenario did not go through full matrix testing (all participants to all other participants) because of Internet constraints between participant servers. Each participant demonstrated the ability to send and receive a signed/encrypted

file of this size and return an MDN after decryption and verification of the sender's signature. Additional 10MB tests were performed in full matrix fashion both with XML and EDI type data files.

Folded Headers

There have been problems detected when sending folded headers (headers continued on multiple lines) to systems on Microsoft Internet Information Server (IIS). Microsoft does not currently have any plans to correct this problem in the current version of IIS so the software tested includes workarounds for this problem. They have indicated this may be corrected in a future release.

Several of the participants are planning MVS (mainframe) implementations which will require the use of folded headers (line lengths need to be shorter than 76 characters). This will prevent interoperability between these mainframe implementations and systems running on IIS.

As a work-around, all participants removed folded headers for products in this test.

Certificate Header Field Population

Some systems have experienced difficulties with some certificates -- specifically those built with multiple attributes indicating the ongoing immaturity of interoperability between PKI software. A certificate contains a series of **Sequences**. Each **Sequence** may contain a series of **Sets**. While all toolkits retain the Sequence order in security operations, some may reorder Sets within a given Sequence. While this is not specifically incorrect, it may cause problems interoperating with toolkits which expect the order to be respected.

The work-around for this problem is to put only one **Set** per **Sequence**. The RSA toolkit allows this using the New-Line-Flag option.

Certificate Types

Certificate exchange/installation is typically the most difficult type of problem encountered during installation in the *real-world*. We tested certificates as thoroughly as possible. Certificates were generated multiple times (per participant) by Verisign, Entrust and a variety of other public Certificate Authorities. Some individual participants generated their own certificates (those whose systems had this capability -- not required). Certificates were tested as common certs while some participants chose to use separate certs for signing and encrypting. Certificates were created with a variety of fields filled in, with and without *urls*, and with/without a trusted root (self-signed certificates). While there is no way to test all possible certificate scenarios, a wide variety of situations were used.

Final Test Results

Completed

STATUS	A	C	F.3	G.1	G.2	G.3	G.5	H.1	H.2	H.3	STATUS
A <i>bTrade</i>	bcdefghi BCDEFGHI	bcdefghi BCDEFGHI	e C	bcdefghi BCDEFGHI	bcdefghi BCDEFGHI	bcdefghi BCDEFGHI	bcdefghi BCDEFGHI	bcdefghi BCDEFGHI	bcdefghi BCDEFGHI	bcdefghi BCDEFGHI	DONE
B <i>Cyclone Commerce</i>	acdefghi ACDEFGHI	acdefghi ACDEFGHI	a C	acdefghi ACDEFGHI	acdefghi ACDEFGHI	acdefghi ACDEFGHI	acdefghi ACDEFGHI	acdefghi ACDEFGHI	acdefghi ACDEFGHI	acdefghi ACDEFGHI	DONE
C <i>GeGXS</i>	abdefghi ABDEFGHI	abdefghi ABDEFGHI	b D	abdefghi ABDEFGHI	abdefghi ABDEFGHI	abdefghi ABDEFGHI	abdefghi ABDEFGHI	abdefghi ABDEFGHI	abdefghi ABDEFGHI	abdefghi ABDEFGHI	DONE
D <i>IPNet Solutions</i>	abc fghi ABC FGHI	abc fghi ABC FGHI	c F	abc fghi ABC FGHI	abc fghi ABC FGHI	abc fghi ABC FGHI	abc fghi ABC FGHI	abc fghi ABC FGHI	abc fghi ABC FGHI	abc fghi ABC FGHI	DONE
E <i>IPNet Solutions (1)</i>	abc fghi ABC FGHI	abc fghi ABC FGHI	l A	abc fghi ABC FGHI	abc fghi ABC FGHI	abc fghi ABC FGHI	abc fghi ABC FGHI	abc fghi ABC FGHI	abc fghi ABC FGHI	abc fghi ABC FGHI	DONE
F <i>iSoft</i>	abcdeghi ABCDEGHI	abcdeghi ABCDEGHI	d G	abcdeghi ABCDEGHI	abcdeghi ABCDEGHI	abcdeghi ABCDEGHI	abcdeghi ABCDEGHI	abcdeghi ABCDEGHI	abcdeghi ABCDEGHI	abcdeghi ABCDEGHI	DONE
G <i>SeeBeyond</i>	abcdefhi ABCDEFHI	abcdefhi ABCDEFHI	f E	abcdefhi ABCDEFHI	abcdefhi ABCDEFHI	abcdefhi ABCDEFHI	abcdefhi ABCDEFHI	abcdefhi ABCDEFHI	abcdefhi ABCDEFHI	abcdefhi ABCDEFHI	DONE
H <i>Trailblazer</i>	abcdefgi ABCDEFGI	abcdefgi ABCDEFGI	c I	abcdefgi ABCDEFGI	abcdefgi ABCDEFGI	abcdefgi ABCDEFGI	abcdefgi ABCDEFGI	abcdefgi ABCDEFGI	abcdefgi ABCDEFGI	abcdefgi ABCDEFGI	DONE
I <i>Web Methods</i>	abcdefgh ABCDEFGH	abcdefgh ABCDEFGH	h E	abcdefgh ABCDEFGH	abcdefgh ABCDEFGH	abcdefgh ABCDEFGH	abcdefgh ABCDEFGH	abcdefgh ABCDEFGH	abcdefgh ABCDEFGH	abcdefgh ABCDEFGH	DONE

Note: Caps mean sent, Lower case mean received

Uppercase indicates successful send. (e.g. if bTrade has a *D* under test A this indicates a successful send to IPNet. Lower case indicates successful receive. All participants submit a cumulative status sheet after each set of tests (see below) and the results are summarized on the table above.

Test results were gathered directly from the participants. Only success was reported. Since each Test involved a full duplex (both directions i.e. a send and a reply) for each participant and each participant captured and reported receiving and responding to each test, all tests were reported in duplicate. If a participant believed they sent data and received receipts correctly, that result must be correlated with the corresponding report from the participant at the receiving end of that transaction. In this way, we gather full interoperability testing results from all participants and to all participants (full matrix). This gives a positive interoperability result on each test data and for each testing scenario.

While not every possible situation may be tested, a large portion of the expected real-world scenarios are represented.

The tests represented by this test are an expansion over the previous test set. Larger files were tested in this Test Round and an additional Test Group was added – Asynchronous Replies.

AS2 - EDI Interoperability Testing Status Report – Final Test-step

S = Sent v = verified by receiver **Example:** Sm/Rm = Sent/mdn / Received/mdn

R = Received, p = processed OK and sent back verification, m = sent back appropriate MDN

for: **Cyclone** E = Error detected - fix and resend; b= bounced mail

as of: **8/4/01** m = mdn sent/received back

Status = DONE

Test	A	C	F.3	G.1	G.2	G.3	G.5	H.1	H.2	H.3	Status
bTrade.com	SvRv	SvRv	Rm	SmRm	SmRm	SmRm	SmRm	SmRm	SmRm	SmRm	Done
Cyclone Commerce	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
GeGXS	SvRv	SvRv	Sm	SmRm	SmRm	SmRm	SmRm	SmRm	SmRm	SmRm	Done
IPNet Solutions Inc.	SvRv	SvRv	N/A	SmRm	SmRm	SmRm	SmRm	SmRm	SmRm	SmRm	Done
IPNet Solutions Inc. (1)	SvRv	SvRv	N/A	SmRm	SmRm	SmRm	SmRm	SmRm	SmRm	SmRm	Done
iSoft	SvRv	SvRv	N/A	SmRm	SmRm	SmRm	SmRm	SmRm	SmRm	SmRm	Done
SeeBeyond	SvRv	SvRv	N/A	SmRm	SmRm	SmRm	SmRm	SmRm	SmRm	SmRm	Done
Trailblazer Systems	SvRv	SvRv	N/A	SmRm	SmRm	SmRm	SmRm	SmRm	SmRm	SmRm	Done
webMethods Inc.	SvRv	SvRv	N/A	SmRm	SmRm	SmRm	SmRm	SmRm	SmRm	SmRm	Done

Test Description

- A** Certificate Exchange
- C** Exchange 10MB XML file (Large XML Interchange in an RFC1767 Bodypart) (Using XML data from Test-Data # 5)
- F.3** Exchange 50MB edi file - Signed/Encrypted with unsigned MDN
- G.1** Exchange EDIFACT edi file - Signed/HTTP with signed MDN
- G.2** Exchange X12 edi file - Signed/HTTPS with signed MDN
- G.3** Exchange X12 edi file - Encrypted/HTTP with signed MDN
- G.5** Exchange 10MB X12 edi file Signed/Encrypted with signed MDN
- H.1** Exchange EDIFACT edi file - Encrypted/HTTP with Asynchronous signed MDN over a separate HTTP session
- H.2** Exchange X12 edi file - Signed/HTTPS with Asynchronous signed MDN over a separate HTTPS session
- H.3** Exchange X12 edi file - Signed/Encrypted/HTTP with Asynchronous signed MDN over email

**Note: This is a sample report sheet for Cyclone, i.e. no data in the Cyclone row.*

About DRUMMOND GROUP, INC.

The Drummond Group Inc. works with software vendors, vertical industry and the standards community to drive adoption for standards by facilitating vertical industry pilots, interoperability conformance testing and building competitive supply chain strategies. Founded in 1990, the vendor-neutral group represents best-of-breed in the industry on linking horizontal infrastructure technologies, standards and interoperability issues with the needs of vertical industry such as retail, grocery, healthcare, transportation, government and automotive.

For further information, please contact Beth Morrow at Beth@drummondgroup.com

Appendix

Test A: Certificate Exchange

Test Description: Exchange of certificates by eMail in "certificate-only PKCS#7 in S/MIME" between all test members. This means that S/MIME certificate-only messages are to be enclosed in application/PKCS#7-MIME.

MIME headers.

Originator

- A.1 With URL
- A.2 Without URL
- A.3 Self Certified
- A.4 Certification Authority Certified

Recipient

- Interpret URL
- Does not require URL
- Interprets Certificate correctly
- Interprets Certificate correctly

Test Setup: The URL, which matches the source URL EDI translator address, *SHOULD* be carried in the certificate. However, the receiver *SHOULD NOT* expect that the certificate would contain a URL. The complete certification chain *MUST* be included in all certificates. All certificate verifications *MUST* "chain to root". In the case of a self-signed certificate, the certificate is signed by itself. Note the sender may send one or more combinations of the test above. The receiver should be able to interpret A.1/A.3, A.1/A.4, A.2/A.3 and A.2/A.4 combinations. Additionally, the certificate hash should match the hash recomputed by the receiver.

Data Used: N/A

Test C: Large XML Interchange in a RFC1767 Bodypart

Test Description: This test expands on Test B by adding a large XML file to the MIME envelope.

Test Setup:

Data Used: The XML data from Test-Data #5.

Expected Results: The contents of the MIME message and XML file are the same for the original and received copies.

Test F: Receipt Request /Unsigned Receipt Returned

Test Description: The initiator creates a normal exchange as in Test B. The presence of the header field “MDN-request-header” (“Disposition-Notification-To:” *url*) in the message indicates to the receiving system that an unsigned Message Disposition Notice (MDN) is expected to be received by the exchange initiator.

Note: This is several tests using different combinations of signed and encrypted messages.

<u>Originator</u>	<u>Recipient returns a MDN</u>
- F.1 Signed	Unsigned
- F.2 Encrypted	Unsigned
- F.3 Signed/ Encrypted	Unsigned

Test Setup: Use Test B as the basic data with the presence of the “Disposition-Notification-To:” to indicate the request of a receipt (unsigned-receipt). A MDN request SHOULD never be ignored between EDI – UA’s that support these recommendations.

Originator: MDN-request-header = “Disposition-Notification-To:” initiator’s EDI gateway address.

Recipient: Normal EDI processing set.

The message originator processing may include:

- Turn on “MDN expected” for this transaction
- Set time-based counter for expecting an MDN
- Identify lost or delayed messages based on no MDN
- Determine whether automated process can occur based on MDN message content

The original message receiver processing should include:

- Turn on “Send MDN”
- Verify the MDN format
- Set time-based counter for returning an MDN
- Return MDN with appropriate status (disposition)

Data Used: F.3 50MB EDI

Expected Results: The return of a MDN to the “EDI gateway address” (the address in the “Disposition-Notification-To:” *url*) with the following contents:

- MDN Bodypart 1:

The message sent on YYYY MMM DD at HH:MM:SS (EDT) -0400 to <*url*> with subject “AAAAAAAAAAAA” has <”been processed” | “been processed with warning” | “been processed with error” | “failed”>. This is not a guarantee that the EDI message has been completely processed or understood by the receiving translator.

Note: This section may be any type of text, the above is an example only.

- MDN Bodypart 2:

Return one of the following for the disposition field:

- Disposition: automatic-action/MDN-sent-automatically; processed
- Disposition: automatic-action/MDN-sent-automatically; processed/ error
- Disposition: automatic-action/MDN-sent-automatically; processed/ warning
- Disposition: automatic-action/MDN-sent-automatically; failed

Test G: Receipt Request /Signed Receipt Returned

Test Description: The initiator creates a normal exchange as in Test B. The presence of the header field “MDN-request-header” (“Disposition-Notification-To:” *url*) in the RFC822 message headers indicates to the receiving system that a Signed-Message Disposition Notice (MDN) is expected to be received by the exchange initiator. Both of these must be present for Test G.

Note: This is several tests using different combinations of signed and encrypted messages.

<u>Originator</u>	<u>Recipient returns a MDN</u>
- G.1 Signed/HTTP	Signed
- G.2 Signed/ HTTPS	Signed
- G.3 Encrypted	Signed
- G.4 Signed/Encrypted	Signed
- G.5 Signed/Encrypted Large File	Signed

Test Setup: Use Test B as the basic data with the presence of the

Disposition-Notification-To: *url*
Disposition-notification-options: signed-receipt-protocol=optional, pkcs7-signature;
signed-receipt-micalg=optional, sha1, md5

to indicate the request for a signed-receipt. A MDN request should never be ignored between EDI-UA’s that support these recommendations.

The message originator processing may include:

- Turn on “Signed-MDN expected” for this transaction
- Set time-based counter for expecting a Signed-MDN
- Reconcile received MDN and sent messages for the fields: Reporting-UA:, Original-Recipient:, Final-Recipient:, Original-Message-ID:, Disposition:, Signed-receipt-disposition:, and Received-content-MIC:
- Identify lost or delayed messages based on no MDN
- Determine whether automated process can occur based on MDN message content

The original message receiver processing should include:

- Turn on “Send signed-MDN”
- Verify the request format and the signature
- Set time-based counter for returning an MDN
- Return Signed-MDN with appropriate status (disposition codes)

Recipient: Normal EDI processing set.

Data Used:

G.1	EDIFACT
G.2	EDI-X12
G.3	EDI-X12
G.5	10MB EDI

Expected Results: The return of a signed-MDN to the “EDI gateway’s address” (the address in the “Disposition-Notification-To:” string) with the following contents:

- MDN Bodypart 1:

The message sent on YYYY MMM DD at HH:MM:SS (EDT) -0400 to <*url*> with subject “AAAAAAAAAAAA” has <” been processed” | “been processed with warning” | “been processed with error” | “failed”>. This is not a guarantee that the EDI message has been completely processed or understood by the receiving translator.

Note: This section may be any type of text; the above is an example only.

- MDN Bodypart 2:

Return one of the following for the disposition field:

- Received-content-MIC: Q2h1Y2sgSW50XwdyaXRIQ, <SHA1 | MD5>
Disposition: automatic-action/MDN-sent-automatically; processed
- Disposition: automatic-action/MDN-sent-automatically; processed
- Disposition: automatic-action/MDN-sent-automatically; processed/error
- Disposition: automatic-action/MDN-sent-automatically; processed/warning
- Disposition: automatic-action/MDN-sent-automatically; failed

If the above error condition is blank or omitted there is assumed to be no error.

Notes:

- Signed-receipt-disposition: Each tester should pick a different value for the test
- The Received-content-MIC: should match the MIC of the original message in Base64 content transfer type encoding
- Received-content-MIC: The test default is Base64 transfer encoding
- The identity of the signature of the entire signed-MDN may not match the From: field of the message

Test H: Asynchronous Receipt Request /Signed Receipt Returned

Test Description: The initiator creates a normal exchange as in Test B. This test is the same as the signed MDN in test H with the addition of the asynchronous option. The presence of the Receipt-Delivery-Option indicates that an Asynchronous Receipt has been requested.

Note: This test is several tests using different URLs for the signed asynchronous messages.

<u>Originator</u>	<u>Type</u>	<u>Sample Return URL</u>
- H.1 HTTP	Encrypted	http://www.company.com/ediua
- H.2 HTTPS	Signed	https://www.company.com/ediua
- H.3 MAILTO	Signed/Encrypted	mailto:ediua-user@company.com

Test Setup: Use Test B as the basic data with the presence of the

Disposition-Notification-To: *url*
 Disposition-notification-options: signed-receipt-protocol=optional, pkcs7-signature;
 signed-receipt-micalg=optional, sha1, md5
 Receipt-Delivery-Option: *return-url*

to indicate the request for a signed-receipt. A MDN request SHOULD never be ignored between EDI-UA's that support these recommendations.

The message originator processing may include:

- Turn on "Signed-MDN expected" for this transaction
- Reconcile received MDN and sent messages for the fields: Reporting-UA:, Original-Recipient:, Final-Recipient:, Original-Message-ID:, Disposition:, Signed-receipt-disposition:, and Received-content-MIC:
- Determine whether automated process can occur based on MDN message content

The original message receiver processing should include:

- Turn on "Send signed-MDN"
- Verify the request format and the signature
- Set time-based counter for returning an MDN
- Return Signed-MDN with appropriate status (disposition codes)

Data Used: Same as tests G.1-G.3

Expected Results: The Asynchronous (HTTP Status code of 200 returned and session terminated – MDN send initiated later) return of a signed-MDN to the "EDI gateway's address" (the address in the "Receipt-Delivery-Option:" string) with the following contents:

- MDN Bodypart 1:

The message sent on YYYY MMM DD at HH:MM:SS (EDT) -0400 to <*url*> with subject "AAAAAAAAAAAA" has <"been processed" | "been processed with warning" | "been processed with error" | "failed">. This is not a guarantee that the EDI message has been completely processed or understood by the receiving translator.

Note: This section may be any type of text; the above is an example only.

- MDN Bodypart 2:

Return one of the following for the disposition field:

- Received-content-MIC: Q2hlY2sgSW50XwdyaXRlQ, <SHA1 | MD5>
Disposition: automatic-action/MDN-sent-automatically; processed
- Disposition: automatic-action/MDN-sent-automatically; processed
- Disposition: automatic-action/MDN-sent-automatically; processed/error
- Disposition: automatic-action/MDN-sent-automatically; processed/warning
- Disposition: automatic-action/MDN-sent-automatically; failed

If the above error condition is blank or omitted there is assumed to be no error.

Notes:

- The value in Original-Message-ID should match the original Message-ID header value. Each tester should pick a different value for the test
- Signed-receipt-disposition: Each tester should pick a different value for the test
- The Received-content-MIC: should match the MIC of the original message in Base64 content transfer type encoding
- Received-content-MIC: The test default is Base64 transfer encoding
- Whenever a receipt-delivery-options header exists, the return value found in this header supercedes any value in the Disposition-Notification header
- A MDN MUST NOT request a MDN