

## **AS2 Interoperability**

### **Issues Report**

Provides a list of issues resolved over the course of multiple AS2 Interoperability Tests through AS2-3Q09

**November 24, 2009**

Prepared & Facilitated by:  
Drummond Group Inc.  
[www.drummondgroup.com](http://www.drummondgroup.com)

## Table of Contents

Interoperability Issues .....	3
Interoperability Issues Resolved or Affirmed AS2-3Q09 .....	3
Interoperability Issues Resolved or Affirmed AS2-1Q09 .....	6
Interoperability Issues Resolved or Affirmed AS2-3Q08 .....	8
Interoperability Issues Resolved or Affirmed AS2-1Q08 .....	9
Interoperability Issues Resolved or Affirmed AS2-3Q07 .....	10
Interoperability Issues Resolved or Affirmed AS2-1Q07 .....	11
Interoperability Issues Resolved or Affirmed AS2-3Q06 .....	12
Interoperability Issues Resolved or Affirmed AS2-1Q06 .....	13
Interoperability Issues Resolved or Affirmed from previous Test Rounds .....	14
About Drummond Group Inc. ....	17

## Interoperability Issues

During the course of interoperability tests, interoperability issues were discovered or questioned and then resolved through the debugging stage of the test. All products from a given test comply with the corresponding resolved issues. These issues are listed below to assist in resolving any supply-chain trading problem which may occur between products-with-version from this test and AS2 products-with-version from outside the test, including backward versions of these test products.

### Interoperability Issues Resolved or Affirmed AS2-3Q09

1. MDN Receipts from some participant were not parsed successfully because the MDN Receipt was split into two packets along the CRLF CRLF split between headers and body as follows:

Packet 1: Headers, CR, LF, CR  
Packet 2: LF Body

2. Several participants were sending TIFF files with content-types of application/octet-stream instead of image/tiff
3. One participant was checking the content-type of the MDN with case sensitivity, which lead to error-messages that no MDN has been received.
4. One participant was checking for the existence of a Message-ID in the MDN's which is optional with MDN's. The participant, acting as originator, was returning an ERROR 500 when a recipient posted an ASYNC MDN to them.
5. One participant was setting the content-length-header in the HTTP-response of an AS2-request with synchronous MDN's to two bytes too low leading to the effect that the trailing CR/NL after the last MIME-boundary was missing. All except one participant was ignoring this. Sending the correct length solved the problem.
6. Four products had problem processing AS2 message with an AS2 Identifier which had exclamation mark at end as assigned to one participant. For Example:  

```
AS2 Identifier:  a b c !
```
7. Participant was starting to parse the HTTP-response the moment a CR was received and didn't wait until the NL was transferred but was interpreting it as part of the response-body. Together with the above explained effect that the trailing CRNL was missing, the read of the body stopped with one '-' missing, leading to the (in that case correct) error-message that there was no valid MIME-boundary at the end.
8. Participant was failing FN test cases because it expected the filename in the Content-ID where it should have been using the attribute filename in Content-Disposition
9. One participant was sending messages and MDN's with extra CRFL at end. They were also sending messages and MDN's with Base64 encoding containing LF instead of CRFL. Participant resolved by removing the extraneous CRLF's and to their base64 encoding (CRLF at end of lines).

10. Sending participant was receiving "failure to receive HTTP response" when sending to one participant. The sending participant increased an internal connection limit and the tests ran successfully
11. Participant had a problem sending Test Case I test to a participant. The recipient was timing out in the DMZ waiting for their message. Receiving participant changed a timeout value internally from 5 minutes to 30 minutes and the test succeeded
12. Participant was sending messages with non-conformant date format in their headers which caused jetty exceptions on receiving AS2 server. Sender changed the format of their date to conform to RFC
13. Sending participant was failing all outbound AS2 messages despite Recipient getting message, processing and sending back good MDN's. Sender complained about Received-Content-MIC not having a space before the digest algorithm id. Sender agreed to relax check that sender should be flexible in allowing for no space there. Researching specifications did not indicate space was required.
14. Default basic authentication caused problems with one participant. When sending to that participant, the sending participant received an access denial, "HTTP/1.0 401 [ISS.0084.9001] Invalid credentials" WWW-Authenticate: Basic realm="xxx" Connection: Close, Authorization required message. The sender was doing a 'basic authentication' step just before sending the AS2 message; the sender changed their configuration to remove basic authentication for that one recipient.
15. Participant reported failures related to non-empty MIME epilogue in signed AS2 Tests involving signed payloads from originator. The problem was that the originator was generating a non-empty MIME epilogue (another CRLF after the final MIME separator and its CRLF) in the signed message. The resolution was for originator to change their code to stop generating the non-empty MIME epilogue (the HTTP RFC clearly disallows them).
16. This issue revolved around tests with encrypted payloads (with no compression or signing). The problem was that originator was base64-encoding the payload at the MIME layer before encrypting it and recipient was not removing this encoding. The resolution was for recipient to change their code to perform the proper decoding after decryption.
17. This issue revolved around LF versus CRLF in signed AS2. Tests involving signed payloads. The problem was that originator was generating lines ending only in LF in the base64-encoded signature. The resolution was for originator to change their code to generate CRLF for these lines instead of LF.
18. This issue revolved around no space after the key word "processed" as in ;processed, in the MDN disposition. The problem was that the MDN from the recipient contained no space between ";" and "processed" in the MDN disposition (causing originator to treat it as a negative MDN). The resolution was for originator to change their code to treat this MDN as a positive MDN by allowing a no spaced after the colon and before key word "processed".
19. This issue revolved around a tab character included in the Disposition-Notification-Options header. The problem was that the Disposition-Notification-Options header sent by the originator contained a tab character after the semi-colon which caused the

recipient to reject the AS2 message. The resolution was for recipient to change their code to accept tabs in this header.

20. A participant was rejecting a sender's AS2 message with the error: " 500: HTTP/1.1 500 Cannot convert date: Tue Sep 29 10:18:18 EDT 2009". This date format is the Java default format, i.e., "Tue Sep 29 11:21:09 EDT 2009". The sender changed their date format to conform to RFC 822, for example to "Tue, 29 Sep 2009 11:21:09 EDT", which recipient then accept as valid.
21. This issue revolves around SSL connectivity between different SSL toolkits. While doing a Client Handshake whenever session-resume is requested by the Client and the version of Client Hello does not match the version of previous handshake, an SSLException is generated. This is happening with one participant's SSL Client using Java 6 APIs with two other participants SSL Servers: one using IAİK and the other Entrust Toolkit for Java (which currently only supports JDK 1.4 and 1.5)

The issue was resolved by the client participant using the Java 6 APIs setting explicitly the session timeout and session cache size. This solved the failed test cases with JDK 1.6 being in use, so there is was no need for the client participant to be restricted to using JDK 1.5. The setting of above settings is described in <http://java.sun.com/javase/6/docs/api/javax/net/ssl/SSLSessionContext.html>

The timeout was set to 1 second and cache size was set to 1 to make sure that there is no session-caching happening. This is not the final solution as the JDK 1.6 participant is working with their toolkit vendors to resolve the interoperability issue with the other toolkits. In summary, the Sun Java API should not send TLSv1.2-messages and mark them as TLSv1.0.

## Interoperability Issues Resolved or Affirmed AS2-1Q09

1. Participant was not accepting uppercase 'SHA1' for specifying the digest algorithm in the AS2-received-content-MIC-field and in the signed-receipt-micalg as they believed that it must be lowercase 'sha1'. This value is case insensitive and participant changed code to allow any case.
2. Participant was specifying micalg=SHA instead of micalg=SHA1 in the content-type header of signed messages. Participant resolved by modifying their security package.
3. Participant expected Message-ID in MDN's however they are optional and not required. Participant modified code so as not to require Message-ID's.
4. Payloads with indefinite lengths should padded at the end with EOC (x'00'), and nothing else. One participant was padding with zeroes (x'00') but was followed by a linefeed character. This caused the test case to fail. Participant modified code to remove the final linefeed character (x'0A').
5. Participant was incorrectly using the From field of the AS2 MDN instead of the AS2-From as the value of the final-recipient.
6. Participant was sending an AS2 message with bad date format and one recipient responded with a 500 error response code with the error "Cannot convert date: Tue,07Apr200912:35:27GMT." and then shutdown the connection. Participant resolved by adding spaces to the date value where appropriate.
7. One Participant resolved security retrieval information when # character assigned to Participants AS2 Identifier.
8. Participant was not including Disposition Notification field as required by the test case and thus the test case failed because MDN was never returned.
9. Sending "expect : 100 continue" caused issued with other participant http server. Participant disabled sending "expect: 100 continue"
10. Participant was not adding quotes to the AS2-From, AS2-To MDN field values as expected when these values contained spaces.
11. Participant removed linefeed at end of encrypted data which caused other participant parser to fail, and at the end of multipart (e.g. mdn) data which caused other partner to fail when verifying signature.
12. There was an incompatibility between curl (as a client) and JSSE (as a server) in https. This causes failure because of TLS Extension "Ticket Session" sent by curl and not understood (TLS 1.1) or rejected silently (TLS 1.0) by JSSE. An upgrade to both sides was the solution (curl 7.9.14/openssl 0.9.8j and Java 1.6.0u12).

RFC 2246 that TLS 1.0 with extensions should be supported:

"Forward compatibility note: In the interests of forward compatibility, it is permitted for a client hello message to include extra data after the compression methods. This data must be included in the handshake hashes, but must otherwise be ignored. This is the only handshake message

for which this is legal; for all other messages, the amount of data in the message must match the description of the message precisely."

13. One participant was using a key size of RSA 4096 bits in their public key but other participant does not support that key size. Per the AS2 test plan, "Testing will assume 128 bit and 1024/2048 bit keys and 24 byte 3DES unless consensus for larger keys is reached with all participants". Participant changed change certificate to agreed key size.
14. Participant was returned "Signature verification fails when verifying signature on compressed payload" for test case G and J. Participant modified default content-transfer-encoding on the body part when there is none specified. Participant was using Bouncy Castle as their S/MIME library. If content-transfer-encoding is not specified on the body part, Bouncy Castle uses 7-bit encoding as a default value. Participant modified code to set "binary" as the default content-transfer-encoding. NOTE: This issue also occurred as issue No.3 as reported for AS2-1Q08" in this "AS2 Interoperability Issues Report".
15. Participant was not providing the Content-Length header in the MDN returned. This caused the message digest to be calculated incorrectly and thus fail. Participant modified code to add Content-Length.
16. Participant was not returning the proper Disposition for MDN Conformance test case K.3. Participant could getting a processing error:

[2009-04-17 08:32:46,137 Process\[PID: 646, Desc.: No description\] ERROR \[...\] indefinite length primitive encoding encountered](#)

Participants newer version of their S/MIME toolkit was causing the issue as newer version threw different exception: ([IllegalStateException](#)). Participant changed code to catch new exception code. Proper MDN response should have been:

[Disposition: automatic-action/MDN-sent-automatically; processed/error: decompression-failed](#)

17. First participant sent a Bravo CEM request to a second participant and their Bravo cert was in a "pending" state. Second participant then sent a Bravo CEM request to first participant before accepting their Bravo cert request. First participant immediately accepted Bravo cert (and as per the spec, as soon as a partner accepts the new signing cert, they must be prepared to accept messages signed with the new certificate). Second participant then sent a response to their Bravo CEM request signed with new Bravo cert (since first participant had accepted it) but the MDN returned from that request contained an authentication error since first participant had not yet updated the second participants profile with the new Bravo certificate.

## Interoperability Issues Resolved or Affirmed AS2-3Q08

1. Participant was not providing filename in the Content-Disposition for Filename Preservation test cases.
2. Participant was sending MDN's with folded headers and reconfigured to not send folded headers in MDN's.
3. Participant was calculating incorrect MIC digest value on test cases F, H, I, and FNP-F. The sender was not including "multipart" in the Content-Type.
4. Participant was assigned AS2 Identifier with angled brackets (< >) as part of the identifier however one participant did not support processing angled brackets; angled brackets are allowed and participant added support for allowing angled brackets.
5. Participant was not calculating correctly the MIC on test cases with Multiple Attachments for test cases MA-D, MA-E.
6. Participant was unable to properly decompress a compressed message.
7. Participant was failing test cases because the data is using indefinite length encoding. When indefinite length encoding is used, the data was getting terminated by end of content bytes, which is 00 00. In the data, these bytes are found before actual end of data; hence the parser was not reading the data beyond the end of content bytes. (Test Cases: D, G, J, VFNP-D, VFNP-G and VFNP-J)
8. For MA test cases, participant was missing the required "Type" parm in the multi-part/related content-type header. Content-Type: multipart/related however it is required (RFC 2387). Also, participant content-type value of the first attachment which was an XML file had a content-type of multipart-related/plain it should have been application/xml. Finally, the second attachment did not have a Content-Type header.
9. Participant was sending Content-Disposition headers with additional double-quotes around the filename attribute, for FNP MA test cases, for example:

```
Content-Disposition: attachment; name="\"T234-ma_test_data_1.xml\"";  
filename="\"T234-ma_test_data_1.xml\""
```

10. Participant was not including Content-Type header for the second attachment of the MA test cases. Recipient was requiring a Content-Type and modified code to default to "text/plain" per the specifications; however attachment was actually a PDF file. Sending participant modified code to add Content-Type header. Below is example of missing Content-Type;

```
--boundarySA==  
Content-Id: <mime-part-1>  
Content-Disposition: attachment; name="z_ma_test_data_2.pdf";  
filename="z_ma_test_data_2.pdf"
```

## Interoperability Issues Resolved or Affirmed AS2-1Q08

1. Participant was receiving 400 bad requests when sending Async MDN's to two participants. The problem was that the initiating participant was sending the "length" twice i.e., in the transport header and also in MDN headers. Hence the receiving partner replied with a HTTP response code 400 - bad request".
2. One participant was sending folded headers in MDN's. Based on earlier consensus items, folded headers are not allowed because some Microsoft IIS does not process folded headers. Also, the Date header was incorrectly inserted before the protocol and boundary parameters.

Wrong:

```
Content-Type: multipart/signed; micalg=sha1;
Date: Wed, 02 Apr 2008 12:55:01 UTC
    protocol="application/pkcs7-signature";
    boundary="-----_Part_361_1392923398.1207140899485"
```

Correct:

```
Content-Type: multipart/signed; micalg=sha1; protocol="application/pkcs7-
signature"; boundary="-----_Part_361_1392923398.1207140899485"
```

3. Bouncy Castle 1.38 assumes that if Content Transfer Encoding is not specified in the body part that the encoding is 7-bit instead of binary.

Participant was getting:

```
"org.bouncycastle.cms.CMSException: invalid signature format in message: content hash
found in signed attributes different"
```

The Recipient participant modified the default to be "binary", for example:

```
BodyPart bodyPart = mime.getBodyPart(0);
if (bodyPart.getHeader("content-transfer-encoding") == null) {
    signedParser = new SMIMESignedParser(receivedMsg, "binary");
} else {
    signedParser = new SMIMESignedParser(receivedMsg);
}
....
```

4. Participant was receiving Content-Length with leading zeros but the participant did not have support for leading zeros.

Example:

Content-Length: 00000000000977

RFC2616 says the content length is formatted as 1\*DIGIT. This means that at least 1 digit must be present and it can repeat. The value of DIGIT is 0-9. Therefore the following values 00000000000977 and 977 would satisfy the rule and should be considered equivalent.

See: 14.13 Content-Length

<http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html#sec14.13>

See: BNF notation

<http://www.apps.ietf.org/rfc/rfc2234.html>

5. One Participant's outer Content-Type header was missing the mime type (and protocol) element. Receiving participant was reporting an error: "Unknown MIC algorithm: null".
6. One participant during persistent HTTP connections was not resetting the Async MDN flag, hence they were sending HTTP 200 response with html message body (in addition to regular MDN) for test cases requesting Sync MDN after Async MDN test case was executed.
7. One participant assumed that the Message-ID in Async MDN's was required however it is optional.
8. Participant was acknowledging receiving AS2 message with a 200 OK, but not closing the connection until after the Async MDN Receipt was returned.

## **Interoperability Issues Resolved or Affirmed AS2-3Q07**

1. One participant was using the character "#" in PrintableString field in their certificate but it is not allowed.
2. One participant's certificate was using Extended KeyUsage extension marked as critical but this field is not supported as critical.
3. One participant's SSL certificate was invalid due to invalid encoding of AuthorityKeyIdentifierExtension.
4. One participant was calculating incorrect MIC on base64 encoded MIME part of AS2 message
5. One participant enabled base64 encoding at transport level but some participants failed to process AS2 message. Earlier consensus determined that base64 encoding at transport level is not allowed.

6. One participant was not adding quotes to MDN AS2-To field value in as required for AS2 identifiers which include spaces.
7. One participant reported incorrect Disposition for K.1 Test Case. Reported processed” only, no reason given and evaluated positive when it should have been evaluated negative.
8. One participant reported incorrect Disposition value of for K.2 Test Case: automatic-action/MDN-sent-automatically; processed/error: decryption-failed;/processed/error: decryption-failed;
9. One participant reported incorrect Disposition on K.3 Test Case. Reported as “Processed”, should have reported: Processed/Error: decompression-failed
10. HTTPs SSL handshake issue resolved by increasing response timeout.
11. One participant introduced new SSL layer – issues reported and resolved.
12. One participant could not support MA payloads if one of the attachments was not XML or EDI. MA payloads contained were PDF and TIFF.
13. One participant was calculating incorrect MIC for MA AS2 message.
14. One participant was not terminating properly the MIME inner boundary on MA AS2 messages.
15. One participant was enveloping valid MIME content with an extraneous boundary marker prior to encryption, for encrypted-only MA AS2 messages.

## **Interoperability Issues Resolved or Affirmed AS2-1Q07**

1. One participant issued certificates with a Country Code using three characters (USA) vs. two (US). Two characters are required, three characters caused Interop issues with some participants.
2. One participant issued certificates using IAString vs. the correct DirectoryString type for organizational unit name field which caused Interop issues.
3. AS2 Identifiers containing embedded spaces MUST be enclosed in double quotes. When the AS2 Identifier is not enclosed in double quotes, the agreed rule is to parse up to the first blank space, however this caused Interop issues. The solution was to enclose the AS2 Identifier with double quotes.
4. The ability to calculate a message integrity check (MIC) on the received message and return it to the sender of the message inside the signed receipt (MDN) is a basic requirement of AS2. The MIC should be calculated over the signed data. One participant was uncompressing the signed data first, then calculating the MIC, thus generating an incorrect value and causing Interop issues. The solution was to calculate the MIC over the signed payload.
5. Several participants were sending folded headers in the MDNs. However, an earlier Consensus item stated that headers should not be folded. The participants unfolded their headers to resolve Interop issues.

6. One participant was encoding their data as 8bit (used in SMTP), however, binary transfer encoding is the default over HTTP. The participant switched to binary encoding to resolve Interop issues as a result of using 8bit encoding.
7. The Message-ID in the MDN is not required. One participant was failing test cases because of the misunderstanding that the Message-ID was required in the MDN's. This continues to be a source for misinterpretation. The Original-Message-ID, however, is required in the MDNs.
8. AS2-From and AS2-To do not have to be UPPER CASE. One participant was failing test cases because they misunderstood that the AS2 portion must be UPPERCASE. MIME/HTTP headers are not case sensitive.

## **Interoperability Issues Resolved or Affirmed AS2-3Q06**

1. Certificates and security toolkit related errors continued to be observed in this test round. Certificates using unusual fields or extensions could create problems within supply-chains. Not all possible certificate fields or extensions were tested against every AS2 product's toolkit, and potential issues could still exist due to certain certificate fields and extensions. Also, it is becoming apparent that not every version of security toolkits is interoperable with every other version.
2. MDN were rejected by at least one participant, based on misunderstanding that Message-ID header is not required. The Message-ID is optional, and the Original-Message-ID is required.
3. Certificates with serialNumber equal to zero could not be processed by at least one participant's security toolkit. Therefore, the consensus from previous Interop's was amended to further constrain serial number to be non-negative integers, greater than, but not including zero.
4. Folded-Headers once again caused Interop issues in this round. Folded-Headers, although allowed in the standards, cause Interop issues with some participants, and thus are not allowed. Several participants were using folded headers, and at least one participant could not accept MDN messages with folded headers (that is a CR LF in the string value of a header).
5. Participant was missing a CRLF for the MIME boundary, or not following CR with a LF, or adding an extra CR LF, causing receiving applications to fail the test. See RFC 2045-2049.
6. Participant was missing the report-type, as in: Content-Type: multipart/report; report-type=disposition-notification; for in the MDN header, however, it was required.
7. Although not required, the To: and From: headers, if used, should follow the MIME header formatting rules. At least one participant was not enclosing the values with quotes when they were required. Same issue appeared for AS2-To and AS2-From headers.
8. Base64 encoding the entire HTTP body was being used. Note however, that Content Transfer Encoding (CTE) of MIME body parts within the AS2 message is allowed. Consensus was arrived that if the MIME bodies were already encrypted and or

compressed, CTE was neither necessary nor practical for performance reasons. Participants agreed to remove Base64 encoding over the entire HTTP body, which helped resolve Interop issues, and theoretically improved processing performance of messages. Performance metrics are not measured in Interop testing

9. It was agreed that HTTP/1.0 servers are required to close HTTP connections, and it is not the responsibility of HTTP clients. At least one participant was relying on a timeout for the connection to close on the server-side, or for the HTTP client to close the connection. When the HTTP /1.0 Server waited for the HTTP /1.1 Client to close the connection and the Client waited for the Server to close the connection but the Server did not close then the Client timed-out and perceived it as an aborted connection and flagged the test failed. The HTTP 1.0 Servers must close the connection based on the HTTP/1.0 specification.
10. Certificates needed to have two-character country code. This was in the list "Interoperability Issues Resolved or Affirmed from previous Test Rounds", but it occurred in this round as well.
11. A participant had a certificate organizational unit name specified as "R&D" and it was encoded as an IA5String. This is supposed to be a DirectoryString. The participant discovered that their certificate generation tool used to create their certificates was using the outdated IA5String encoding for some of the elements within the Subject and or Issuer name fields.
12. At least one participant found an issue with LF or CRLF being removed on the outgoing payload data which was causing payload mismatches (the sent payload did not match the received).

## **Interoperability Issues Resolved or Affirmed AS2-1Q06**

1. Certificates and security toolkit related errors continued to be observed in this test round. Certificates using unusual fields or extensions could create problems within supply-chains. Not all possible certificate fields or extensions were tested against every AS2 product's toolkit, and potential issues could still exist due to certain certificate fields and extensions.

For example, there was an issue with a participant SSL certificate was not properly formatted (for DER encoding). It caused another participant to reject it during the SSL handshake. The participant's certificate had an explicit default value in the version identifier. The certificate had a 0 in this field, when it should be version 1 (everyone else's certificates had version 1).

2. Also, it was discovered that security toolkit versions are not always interoperable from version to version, and this Interop revealed and helped resolve these incompatibility in the security toolkits. However, security toolkits were not exhaustively tested for interoperability.

3. The AS2 specification requires that human-readable portion of MDN must contain "Final-Recipient". Please see <http://www.ietf.org/rfc/rfc4130.txt> section 7.4.2. A participant was not sending "Final-Recipient" however it was required.
4. A participant was sending "folded headers" in the MDN's and this caused an error because at least one other participant did not process "folded headers". In previous Interop's, it was a consensus item to not fold the headers.

Example:

```
Content-Type: multipart/report; report-type=disposition-notification;  
boundary="----=_Part_1139974138134"
```

5. An AS2 server was attempting to connect to port 80 instead of 443 when the URL was provided without a port, for example: <https://hostname/> The correct port to connect to should be 443, (the default port for SSL when port is not specified). The default port for non-ssl is port 80. Please see: <http://rfc.net/rfc2616.html>
6. MDN conformance testing revealed that one participant's MDN disposition text says, "Disposition: automatic-action/MDN-sent-automatically; processed". It should have returned error text "processed/error: authentication-failed" or "processed/error: integrity-check-failed".

## Interoperability Issues Resolved or Affirmed from previous Test Rounds

1. Some products could not accept certain characters or certain strings of AS2 identifiers. Two specific issues were: 1) having a space (" ") at the third location, e.g. "AS 2", and 2) identifiers containing a comma (","). While these conflicts were very rare and not associated with every participant, supply-chain implementers of these products should avoid identifiers with this syntax and discuss with their AS2 vendor any potential AS2 Identifier issues.
2. Trailing long white spaces (LWS) at the end of HTTP headers is not permitted. Leading LWS is allowed within HTTP (RFC2616) but not clear if trailing LWS is or is not.
3. The value "RSA-SHA1" was used by some participants for the MIC algorithm of the digital signature. It is a valid value and should be considered equal to that of the more common "SHA1" value. "RSA-SHA1" is a legacy value from an earlier S/MIME implementation.
4. Field names in MDNs, such as Original-Message-ID, are case-insensitive. According to RFC2298, section 3.1.1, "field names are case-insensitive, so the names of notification fields may be spelled in any combination of upper and lower case letters." As well, it is permissible to have a white space character (" ") before the message-id value of the Original-Message-ID field in the MDN. Thus, the two examples below are considered identical:
5. Original-Message-ID:<123foo@example>
6. original-message-id: <123foo@example>

7. The Message-ID header is not required in MDNs.
8. Chunked encoding for HTTP 1.1 requests and responses is acceptable for AS2. Rules for implementing, supporting and understanding chunked encoding can be found in the HTTP 1.1 standard, RFC2616.
9. Some products require valid EDI/XML documents on inbound messages and will generate MDNs with errors if they are invalid. This includes both valid formatting and/or recognized identifiers.
10. Certificate serial numbers must not be negative, per RFC3280. While some AS2 systems accept negative serial numbers, other systems cannot accept negative values.
11. Certificates are uniquely identified through their Issuer name and their serial number. As with negative serial numbers, certain AS2 systems will reject duplicate certificates, but others can accept them.
12. Some products utilizing the open source OpenSSL experienced problems in SSL transactions. The cause was due to the sending of empty fragments in the transaction which caused some trading partner products to corrupt the inbound document. The solution was to modify configuration flags within OpenSSL.
13. HTTP Content-length header is not necessarily required on MDN. The HTTP standard specifies the use and requirement of this header, and the AS2 draft is being updated to refer back to the HTTP standard for the use of content-length.
14. MIME Folded headers continue to cause problems with several products due to their associated web server. Folded headers were not used during the test and should be avoided in actual implementation.
15. The use of quotation marks on AS2 System Identifiers should not be used for atomic names. Also, the use of quotation marks on AS2 System Identifiers must be consistent for both the payload messages as well as for the MDNs. That is, if quotation marks are used in the payload message, they also must be present in MDNs.
16. A 204 (No content) HTTP response would be acceptable in an HTTP response of an async MDN request. This should be accepted (assuming the response has no body). From the latest version (13) of the AS2 draft, section 7.6, notice the comment of the response being "in the 200 range." HTTP RFC2616 states that if a 204 is returned, there is to be no message body and the message is terminated by the first empty line after the header fields. So, the 204 will work as long as there are only HTTP headers in the response.
17. If certificates use the country attribute, the country attribute may only contain two characters. For example, "C=USA" is invalid and instead should be listed as "C=US".
18. Encrypted messages can contain multiple RecipientInfo structures within the CMS data, including one describing the originator. Refer to RFC 2630 Section 6 for more details.
19. Consensus was reached that AS2 messages with EDI payloads should identify the content-type either as application/EDI-X12 or application/EDIFACT and NOT application/EDI-CONSENT.
20. The Message-ID is not required in Asynch MDN's because the AS2 standard states it SHOULD be contained, that is, it is not required. Asynch MDN's should not be rejected

if MDN's do not contain Message-ID because it is not required. It is recommended that it be present. Please refer to the meanings of SHOULD and MUST.

## **About Drummond Group Inc.**

Drummond Group Inc. (DGI) is an independent, privately held company that works with software vendors, vertical industries and the standards community to drive adoption of open standards by conducting interoperability and conformance testing, publishing related strategic research and developing vertical industry strategies. Founded in 1999, DGI represents best-of-breed in the industry on linking horizontal infrastructure technologies, standards and interoperability issues with the needs of vertical industries such as retail, grocery, health care, transportation, government and automotive. For more information, please visit [www.drummondgroup.com](http://www.drummondgroup.com) or email: [info2@drummondgroup.com](mailto:info2@drummondgroup.com).