# Final Report

# AS4 Interoperability Test

# 2022


## August 15, 2022

Prepared & Administered By:
DRUMMOND GROUP
http://www.drummondgroup.com/

# Table of Contents

# Cover Letter

Drummond Group is pleased to announce that the participants listed in this report have completed all requirements and passed the test requirements for the Drummond Certified™ AS4 Interoperability Certification 2022 Test Event. This was the nineth Drummond Group facilitated AS4 test event to require full-matrix interoperability between each product. Full-matrix testing certifies that all of the products work with each other over the different conformance profiles for which they have tested. This report provides the description of how these products were tested, the technical requirements and test cases required of them, listing of important consensus items made and insight into product configuration setup used to achieve interoperability. The Overview of the Test Event section highlights the scope of this report and provides hyperlinks to the key sections of the document.

Please note that an AS4 interoperability certification indicates the interoperability of a specific product-with-version, such as AS4 Product X v2.0, within a specific group of other products-with-version for a given test round, such as AS4 2022. Products certified in this test event may not be interoperable with the products-with-version from future certification test rounds unless these products are retested.

Interoperability test rounds must be periodically repeated to verify that as product names, versions or releases change they remain interoperable. Therefore, passing an interoperability test round does not guarantee perpetual interoperability. The relevance of an AS4 test round certification within real world deployment diminishes with time. New products enter the market and existing products change with revisions and updates. Given such changes in the product test group, an interoperability certification does not guarantee perpetual interoperability within real world deployment, and interoperability test events must be repeated to include new products, unchanged existing products, and existing products with new versions. From a market perspective, interoperability lasts for 6-12 months.

InSitu™, a patented test automation tool, was introduced in this test event and will play a critical role in future test events to allow for automated testing of all test cases and elimination of the manpower requirements for coordinating testing, permitting participants to focus on debugging their codebase.

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

Sincerely,

Aaron Gomez
Standards Certification
Drummond Group

# Disclaimer

Drummond Group conducts interoperability and conformance testing in a neutral test environment for various companies and organizations ("Participant"). At the end of the testing process, Drummond Group may list the name of the Participant in the final test report along with an indication that the Participant passed the test. The fact that the name of the Participant appears in the Final Report is not an endorsement of the Participant nor its products or services, and Drummond Group therefore makes no warranties, either expressly or implied, regarding any facet of the business conducted by the Participant or their product.

# Test Participants

| | |
|---|---|
| **Axway**<br><br>http://www.axway.com<br><br>**Product Name:** **Axway B2Bi 2.6 / Activator 6.1** | **CData Software**<br><br>*https://www.cdata.com/arc*<br><br>**Product Name:** **CData Arc v2022** |
| **Cleo**<br><br>*http://www.cleo.com*<br><br>**Product Name:** **Cleo Integration Cloud: Private Cloud Edition (Version 5.8 of Cleo LexiCom®, Cleo VLTrader® and Cleo Harmony®)** | **OpenText**<br><br>*https://businessnetwork.opentext.com/b2b-gateway/*<br><br>**Product Name:** **BizLink v16.6** |

# Definitions

***Interoperability*** -- A product is deemed interoperable with all other products in the Interoperability Test Round if and only if it demonstrates in a full-matrix manner the pair-wise exchange of data covering the *Test Criteria* between all products in the Interoperability Test Round. A product is either totally interoperable or it is not interoperable. Waivers or exceptions are not given in demonstrating interoperability for the *Test Criteria* unless the entire *Product Test Group* and Drummond Group agree.

***Interoperable products*** – The group of products, from the *Product Test Group*, which successfully completed the *Test Criteria*, in a full-duplex manner with every other *Product Test Group* participant in an Interoperability Test Round without any errors in the Final Test Phase. Interoperable products receive a Drummond Certified ™ Seal.

***Product Test Group*** – A group of products involved in an interoperability or conformant Test Round.

***Product, product-with-version, or product-with-version-with-release*** – are interchangeable and are defined for the purpose of a Test Round as a product name, followed by a product version, followed by a single digit release. The assumption is that version and release syntax is as: "VV.Rx…x," where VV is the version numeral designator, R is the single digit release numeral designator and x is the sub-release multiple digit numeral designator. Drummond Group assumes that any digits of less significance than the R place do not indicate code changes on the product-with-version-with-release tested in the Test Round. A vendor must list a product as product name, followed by version digits followed by a decimal point followed by a single release designator digit before the Test Round is complete.

***Test case*** – The test criteria is a set of individual test cases, often 10 to 50 which the product test group exchange among themselves to verify conformance and interoperability.

***Test Criteria*** – A set of individual tests, based on one or more standard specifications, that is used to verify that a product is conformant to the specification(s) or that a set of Product-with-versions are interoperable under the *Test Criteria*.

# Interoperability Test Summary

### Overview of Test Event

The AS4 2022 Interoperability test event consisted of four participanting products: Axway, CData Software, Cleo and OpenText. These products successfully achieved Drummond Certified™ Interoperable status for the AS4 2022 test event. They performed full-matrix testing over different AS4 conformance profiles without error or code changes during the AS4 2022 Certification Run during the first week of August 2022 to demonstrate their interoperability. The time preceding the Certification Run was set aside for execution of all the required test cases for the purpose of debugging interoperability issues, and preparing for the Certification Run. The list of products and the conformance profile(s) for which they were certified can be found in the Final Test Results section of this report.

There are three conformance profiles for AS4 implementations, as defined within the AS4 1.0 specification by OASIS. In order to be certified, each vendor was required to perform full-matrix testing. Full-matrix testing requires each participant to test with every other participant for all applicable test criteria. The list of which test cases were required can be found in the section of this report summarizing the test cases and conformance profiles.

To assist in the interoperability of these products in real-world deployments, specific details required for achieving interoperability can be found in the Interoperability Issues section. This section explains how the products were configured and key consensus items made to ensure their interoperability.

Finally, this report contains sections describing the trading partner requirements and technical requirements given to the participants in order to complete full-matrix interoperability testing, as well as a section summarizing the Drummond Group Interoperability and Compliance Process.

## Final Test Results

AS4 1.0 specifies three conformance profiles and the specific features that are either required or optional for each profile. The details of each profile are provided in [AS4-Profile], and the conformance profiles are listed here:

- MC – Minimal Client
- LC – Light Client
- ebH – ebHandler

This test event focused on the ebH Conformance Profile. Please note that the ebH Conformance Profile test cases are a superset of the other profiles. Ideally speaking, if a product supports the ebH Profile, it can support the Minimal Client and Light Client profile as well. In test events where products support only the LC or MC Conformance Profiles, testing of these profiles would be required. Testing of the MC and LC Conformance Profiles was not required in this test event as all participants demonstrated support for the superset ebH Conformance Profile successfully.

## About AS4 v1.0

AS4 v1.0 is an open standard developed by the ebXML Messaging Services Technical Committee at OASIS as a profile to the ebMS 3.0 specification. While ebMS 3.0 represents a leap forward in reducing the complexity of Web Services B2B messaging, the specification still contains numerous options and comprehensive alternatives for addressing a variety of scenarios for exchanging data over a Web Services platform. The AS4 profile of the ebMS 3.0 specification has been developed in order to bring continuity to the principles and simplicity that made AS2 a successful messaging protocol, while adding better compliance to Web Services standards, and features such as message pulling capability and a built-in receipt mechanism. Using ebMS 3.0 as a base, a subset of functionality is defined along with implementation guidelines adopted based on the "just-enough" design principles and AS2 functional requirements to trim down ebMS 3.0 into a more simplified and AS2-like specification for Web Services B2B messaging. In addition to addressing EDIINT requirements, a Minimal Client conformance profile is provided that addresses lower-end exchange requirements. This document defines the AS4 profile as a combination of conformance profiles that concern an implementation capability, and of a usage profile that concerns how to use this implementation. Several variants are defined for the AS4 conformance profile - the AS4 ebHandler profile, the AS4 Light Client profile and the AS4 Minimal Client profile which reflect different endpoint capabilities. Drummond Group AS4 Interoperability testing focuses on the ebHandler profile.

# Test Case and Conformance Profile Summary

### Test Case and Conformance Mode Summary: Overview

The certification event contained test cases which covered the conformance profiles defined by the AS4 1.0 specification. All conformance profiles were exclusive to the other profiles and each was tested by the participants. Each test case was part of one or more conformance profiles.

### Test Cases and Test Criteria

The test criteria and the subsequent test cases cover all the conformance profiles for this test event and were the same test cases that were part of the first round of AS4 interoperability testing in 4Q13, vetted by that initial test group of participants both at the outset of that initial test event, and during that test event. The actual test case descriptions for this test event can be found in the Appendix.

### AS4 Defined Conformance Profiles

AS4 1.0 specifies three conformance profiles and the specific features that are either required or optional for each profile. The details of each profile are provided in [AS4-Profile], and the conformance profiles are listed here:

- MC – Minimal Client
- LC – Light Client
- ebH – ebHandler

The Minimal Client and Light Client profiles were not tested in this event. Implementations of the ebHandler Conformance Profile were tested in this event. The ebH profile encompasses all test cases that the MC and LC Conformance profiles are required to support.

### Test Cases Associated with Conformance Profiles

In order to achieve certification in one or more of the AS4 Conformance Profiles, the associated required test cases must be completed with all test participants with aligning profiles. The specific pairing among participants are given at the beginning of the certification event. A conformance profile may not require completion of all the test steps in the associated test cases. For instance, normally in a full-matrix test, each trading partner executes each test case as both the Initiator and the Responder. Light Client implementations can only perform the test cases as the Initiating party.

| Conformance Mode | Test Cases |
|---|---|
| AS4 Light Client | B.1.1, B.1.2, C.1.1, C.1.2, D.1.1, D.1.2, D.2.1, D.2.2, E.1.1, E.1.2, E.2.3, E.4.1, E.4.2, E.4.3 |
| AS4 ebHandler | B.1.1, B.1.2, C.1.1, C.1.2, D.1.1, D.1.2, D.2.1, D.2.2, E.1.1, E.1.2, E.2.3, E.4.1, E.4.2, E.4.3 |

The above tests were identified as representative of the overall Basic Profile test suite and are composed of the most complex required features. These tests comprise the AS4 1.0 Certification Run Test Suite and were executed as part of the Final Test.

Interoperability is determined by each product-with-version successfully sending and receiving each test case with the other participants according to their implemented conformance profile. A test case is successful when the expected result is achieved according to the message specifications.

All products-with-version listed on this test report successfully sent and received test cases B.1.1, B.1.2, C.1.1, C.1.2, D.1.1, D.1.2, D.2.1, D.2.2, E.1.1, E.1.2, E.2.3, E.4.1, E.4.2, E.4.3 with each and every other participant according to their implemented conformance profile.

It should also be noted that no warranty of product interoperability is implied over and above the publishing of the results of the Test Round as completed by all vendors during the specified time period of testing.

### Large Messages

The F.1.1 and F.1.2 Large Message Tests are required test cases and were tested as part of the Certification Run of this AS4 Interoperability Test Event. Axway, CData Software, Cleo and OpenText successfully executed these tests during all test phases in this AS4 Interoperability Test Event. The F.1.1 and F.1.2 large message tests are straightforward tests of a product-with-version's ability to push and/or pull large messages (50 megabyte). The test is not intended as a stress test or as a performance test.

### Multiple Payload Messages

The F.2.1 and F.2.2 Multiple Payload Tests are required test cases and were tested as part of the Certification Run of this AS4 Interoperability Test Event. Axway, CData Software, Cleo and OpenText successfully executed these tests during all test phases in this AS4 Interoperability Test Event.

### Reception Awareness Duplication Detection and Retries

The Reception Awareness Duplicate Detection and Retry tests were not fully defined and were not tested as part of this AS4 Interoperability Test Event. It is expected that the definition of these test cases will be refined and included in future Test Rounds.

### Error Testing

Test cases for error testing, handling, and reporting were not fully defined and were not tested as part of this AS4 Interoperability Test Event. It is expected that the definition of these test cases will be refined and included in future Test Rounds.

### Optional Tests

AS4 functions and features that were designated as “Optional” were not tested as part of this AS4 Interoperability Test Event. They include bundling a receipt on a callback with another PullRequest (E.2.2), message pull with alternative client authentication with receipt on a callback channel (E.3.1), and message pull using username/password tokens on an MPC sub-channel (E.3.2).

# Interoperability Issues

During AS4 interoperability test rounds, issues arise that required consensus to achieve interoperability. Some of these items are outside the scope of the AS4 1.0 and are related to underlying technical specifications, and some of these issues address AS4 1.0 features which have been interpreted differently by different readers.

### AS4 Consensus Items

To be published in an external document.

# Test Requirements

In order to be part of the certified interoperable products-with-versions, each participant must both successfully send and receive all tests cases in the Basic Profile with each and every other participant according to the conformance profile implemented by the product.

### Trading Partner Requirements

All participants were required to establish trading partner relationships with each other according to the P-Mode parameter definitions for each test case. All participants were remote from each other, and all test messages were exchanged over the public Internet. Participants were responsible for distributing their network information and configuring their firewalls to allow all other participants access to their product-with-version.

Each participant provided their security certificates (including SSL server and client certificates) to the other participants for storage in their trusted store. Each certificate conformed to the X.509 standards but varied with respect to the fields used in the certificates. All participants generated their own self-signed certificates. Some participants chose to use a single certificate for all purposes, including SSL Server Authentication, SSL Client Authentication, Digital Signature and XML Encryption.

Additionally, all participants generated username/password credentials for each of the other participants for test cases that required a username/password instead of a digital certification for message authentication.

Drummond Group provided test payloads and user identification aliases.

### Technical Requirements – Basic Profile

Each ebHandler participant successfully sent and received all tests cases in the Basic Profile with each and every other ebHandler participant. Light Client participants can only initiate the test cases with ebHandler Responders. Light Client participants cannot exchange messages with other Light Client participants.

The Basic Profile test cases cover the core requirements of AS4 1.0 and include some optional features of AS4 1.0 that are widely implemented and or desired by end users. These requirements are described directly below.

The effect is that all the products-with-version are proven interoperable over a feature-rich, industry horizontal profile and demonstrates that the products-with-version can cover the technical requirements listed below. For additional technical information regarding AS4 1.0 requirements, please see the specification at:

http://docs.oasisopen.org/ebxml-msg/ebms/v3.0/profiles/AS4-profile/v1.0/os/AS4-profile-v1.0-os.html

### Message Packaging

AS4 supports XML payload packaging within the SOAP message body or leverages SOAP with Attachments (SwA) to define an extensible message package for non-XML payloads. Message headers for routing, partner identification, message identification, time stamping, security token credentials, digital signature, encryption, compression, and other quality of service features are also supported. The message package is also capable of encapsulating one or more business documents or other binary data as payloads. Participant products-with-version must be capable of formatting SwA messages in the manner described by the specification.

### Digital Signature

AS4 leverages XMLDigitalSignature to provide proof of content-integrity, authentication of senders and receivers and Non Repudiation of Receipt. An AS4 signature is a signature over the entire message which may include one or more payloads.

### XML Encryption

AS4 allows for the use of a persistent encryption mechanism that can be applied to payloads within a message. Persistent encryption can be leveraged as an additional layer of security for Internet based messaging; essentially part or all of a message payload may be encrypted in a manner that allows only the intended recipient to decrypt the message. Participants successfully interoperated with a combination of XMLEncryption with DigitalSignature.

### Error Handling

AS4 leverages SOAP Fault semantics for low level SOAP-related errors, and specifies higher level “AS4/ebMS error lists” that can be comprised of a list of warnings and/or errors that occur at the AS4 transport level. For example, a SOAP syntax error will generally result in a SOAP Fault error reply, while a message where TimeToLive has expired will result in an AS4/ebMS defined error list reply stating that the message has expired. Error Handling and Reporting was not tested as part of this AS4 Interoperability Test Event.

### Synchronous and Asynchronous messaging

AS4 supports both synchronous (response) and asynchronous (callback) message patterns. The type of message pattern is defined per test case. This allows AS4 to be highly transfer protocol neutral and to be used in business scenarios where immediate reply is required and in business scenarios where delayed replies are common due to queuing operations, load balancing, system outages or other technical or business reasons.

### Synchronous and Asynchronous Acknowledgments of Receipt

Acknowledgments validate the receipt and persistent storage of a message. Synchronous acknowledgments provide a confirmation of receipt in a message returned over the same session and the same transfer protocol as the original message. Asynchronous acknowledgments are sent back to the originator over a separate session.

Acknowledgments are tested in both synchronous and asynchronous styles, both signed and unsigned. A signed acknowledgment includes hash digests of the original message allowing for true Non Repudiation of Receipt.

### Secure Transport Protocols

Both HTTP & HTTP/s transports were tested.

### Payloads

The AS4 message package provides for multiple payloads. Effectively, more than one business document can be sent in a single message. In some cases, the secondary documents may be binary files such as pictures and are often referred to as attachments; conceptually similar to email attachments.

Tests of single payloads were required to be executed, and these tests included Digital Signature and HTTP/s transport.

These payloads were used throughout the testing:

- Medium sized HIPAA compliant X12 document, approximately 18 kBytes provided by HCCO
- Small EDIFACT EDI document, approximately 2 kBytes
- Small XML document, approximately 600 bytes
- Large XML document, approximately 41 kBytes
- Medium sized XML automotive Parts Order BOD, approximately 4 kBytes
- Large XML automotive Parts Invoice BOD, approximately 1 megabyte
- Very large X12 EDI file, approximately 50 megabytes
- Medium sized binary jpeg file, approximately 11 kBytes

### Large Messages

AS4 provides the ability to transport any data type including large files. As a message service standard gains wider deployment in the market, invariably end users demand the ability to send very large messages. Large message test cases for both push and pull with a 50 megabyte EDI payload were defined and tested in all phases of the AS4 Interoperability Test Event. This test is intended to

prove the ability to send and receive large messages, and is not intended as a performance or stress test.

### GZIP Based Compression

The use of gzip based compression where the payload itself is composed of compressed data using the MIME type application/gzip is supported by AS4. Test cases for a signed, encrypted, and zipped message push were tested in all phases of the AS4 Interoperability Test Event.

# Overview of the Interoperability Compliance Process®

Interoperability of B2B products for the Internet is essential for the long-term acceptance and growth of electronic commerce. To foster interoperability, Drummond Group facilitates interoperability and conformance tests. This section contains a description of the test process involved with creating and listing interoperable products.

### Drummond Group In-the-Queue Test Round

In-the-Queue Test Rounds are designed to allow participants—with products new to Drummond Group interoperability testing, or previously certified products that have made significant product changes or undergone version changes, or missed the most recent test round—to both test and debug their products with the Drummond Group Test Server.

The Drummond Group Test Server is a collection of products-with-version from the previous Interoperability Test Round. These products were provided by the vendors on a voluntary basis. The Drummond Group Test Server allows products new to the interoperability process to be debugged in a quicker manner by testing with proven products-with-version.

Through the In-the-Queue Test Rounds, participants will see their products-with-version become conformant to the AS4 standard and interoperable with the Drummond Group Test Server products. Products which successfully complete In the Queue Test Rounds are considered compliant to the respective standard and will be listed on the www.drummondgroup.com website as “In the Queue," but will not be given product Interoperability Status on the www.drummondgroup.com website.

Successful test completion also qualifies that particular product to participate in the next Drummond Group Interoperability Test round, but does NOT guarantee successful completion of the full Interoperability Test Round. Drummond Group makes no warranties or guarantees that products passing In the Queue Test Rounds will pass the Interoperability Certification Tests.

## Drummond Group Interoperability Test Round

Products-with-version from the previous AS4 1.0 Interoperability Test Round and products-with-version from the In-the-Queue tests come together in a vendor-neutral and non-competitive environment to test with each other in order to become interoperable with each other. In an Interoperability Test Round, each product-with-version must successfully test with each other in order to be certified as interoperable.

The Drummond Group Interoperability Test Round verifies conformance to a standard and then verifies that members of the Product Test Group are interoperable among themselves. Interoperability is an all or nothing within the Product Test Group over the Test Criteria. A product is either interoperable with all other products in the Test Group or it is not.

Products-with-version which demonstrate complete interoperability among the passing members of the Product Test Group are given a Drummond Certified™ Seal and are listed with Interoperability Status on the www.drummondgroup.com website. Interoperability Test Rounds are periodically repeated to verify that as product names, versions or releases change, the products continue to remain interoperable.

## InSitu™ Test System

Drummond Group has created a system for the automation of interoperability testing called InSitu™. InSitu is a patented and innovative testing tool developed for conducting automated interoperability testing that allows multiple products to coordinate the sending and receiving of test cases without human intervention. Manpower requirements for coordinating testing have been eliminated, allowing participants to focus on debugging their codebase.

The InSitu test system was introduced in this test event to allow participants to familiarize themselves with the automation process and to understand the tasks required to begin integrating the use of InSitu into their systems to enable automated testing. During this Test Event, tests continued to be run manually and the InSitu Test System was primarily used for sharing configuration information and for reporting test case evaluations. All products-with-version will be required to have the use of InSitu fully integrated into their systems in future test events.

# References


[AS4-Profile] *AS4 Profile of ebMS 3.0 Version 1.0*. 23 January 2013. OASIS Standard. http://docs.oasisopen.org/ebxml-msg/ebms/v3.0/profiles/AS4-profile/v1.0/os/AS4-profile-v1.0-os.html

[ebMS3-Core] *OASIS ebXML Messaging Services Version 3.0: Part 1, Core Features.* 01 October 2007. OASIS Standard. http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/core/os/ebms_core-3.0-spec-os.html

# Appendix

## Test Case Matrix Summary

### AS4 IOP Test Case Matrix

Version 2.0

| | | | | CPs* | | |
|---|---|---|---|---|---|---|
| | Test Case | Description | R/O | MC | LC | ebH |
| Connectivity | B.1.1 | Minimal Push | R | x | x | x |
| | B.1.2 | Minimal Push SSL | R | x | x | x |
| | B.2.1 | Minimal Pull | R | x | x | x |
| | B.2.2 | Minimal Pull SSL | R | x | x | x |
| Debug Phase | C.1.1 | Minimal Pull w/Basic MPC AuthZ | R | x | x | x |
| | C.1.2 | Minimal Pull w/Basic MPC AuthZ SSL | R | x | x | x |
| | D.1.1 | Push w/Password AuthN | R | | x | x |
| | D.1.2 | Push w/Password AuthN SWA | R | | x | x |
| | D.2.1 | Push w/XMLDsig AuthN | R | | x | x |
| | D.2.2 | Push w/XMLDSig AuthN SWA | R | | x | x |
| | D.3.1 | Pull w/XMLDSig AuthZ | R | | x | x |
| | D.3.2 | Pull w/XMLDSig AuthZ SWA | R | | x | x |
| | E.1.1 | Push w/Password AuthN & Receipt Response | R | | x | x |
| | E.1.2 | Push w/XMLDSig AuthN & Receipt Response | R | | x | x |
| | E.2.1 | Pull w/XMLDSig AuthZ & Receipt Callback | R | | x | x |
| | E.2.2 | Pull w/XMLDSig AuthZ & Bundled Receipt Callback | O | | x | x |
| | E.2.3 | Pull w/Basic MPC AuthZ & Receipt Callback | R | | x | x |
| | E.3.1 | Pull Using Alternate MPC AuthZ SSL | O | | x | x |
| | E.3.2 | Pull from MPC Sub-channel | O | | x | x |
| | E.4.1 | Push w/XMLDSig AuthN & Receipt Callback SSL | R | | | x |
| | E.4.2 | Push w/XMLDSig AuthN & XMLENC & Receipt Response | R | | | x |
| | E.4.3 | Push w/XMLDSig AuthN & XMLENC & Receipt Callback | R | | | x |
| | E.4.4 | Push w/XMLDSig AuthN & XMLENC & Receipt Response (SWA) | R | | | x |
| | E.4.5 | Push w/XMLDSig AuthN & XMLENC & Receipt Callback (SWA) | R | | | x |
| | E.4.6 | Push w/XMLDSig AuthN & XMLENC & Zipped & Receipt Response | R | | | x |
| | E.4.7 | Push w/XMLDSig AuthN & XMLENC & Zipped & Receipt Callback | R | | | x |
| | F.1.1 | Push Large Message | R | | x | x |
| | F.1.2 | Pull Large Message | R | | x | x |
| | F.2.1 | Push Multiple Payloads | R | | x | x |
| | F.2.2 | Pull Multiple Payloads | R | | x | x |

## Test Case Descriptions and Details

| Test Case | B.1.1 |
|---|---|
| **Description** | Simple message push to test connectivity over HTTP |
| **Required/Optional** | Required |
| **Payload** | filename=smallxmlPO.xml; SOAP Body |
| **User Message Security** | None |
| **Receipt** | None |
| **MPCs & Pull AuthZ** | None |
| **MEP & Transport Bindings** | Push over HTTP |

| Test Case | B.1.2 |
|---|---|
| **Description** | Simple message push to test connectivity over HTTPS |
| **Required/Optional** | Required |
| **Payload** | filename=smallxmlPO.xml; SOAP Body |
| **User Message Security** | None |
| **Receipt** | None |
| **MPCs & Pull AuthZ** | None |
| **MEP & Transport Bindings** | Push over HTTPS |

| Test Case | B.2.1 |
|---|---|
| **Description** | Simple message pull to test connectivity over HTTP |
| **Required/Optional** | Required |
| **Payload** | filename=smallxmlPO.xml; SOAP Body |
| **User Message Security** | None |
| **Receipt** | None |
| **MPCs & Pull AuthZ** | AuthZ |
| **MEP & Transport Bindings** | Pull over HTTP |

| Test Case | B.2.2 |
|---|---|
| **Description** | Simple message pull to test connectivity over HTTPS |
| **Required/Optional** | Required |
| **Payload** | filename=smallxmlPO.xml; SOAP Body |
| **User Message Security** | None |
| **Receipt** | None |
| **MPCs & Pull AuthZ** | AuthZ |
| **MEP & Transport Bindings** | Pull over HTTPS |

| Test Case | C.1.1 |
|---|---|
| **Description** | Simple message pull to test username/password authorization on a non-default MPC over HTTP |
| **Required/Optional** | Required |
| **Payload** | filename=smallxmlPO.xml; SOAP Body |
| **User Message Security** | None |
| **Receipt** | None |
| **MPCs & Pull AuthZ** | Non-default MPC; Username/Password AuthZ |
| **MEP & Transport Bindings** | Pull over HTTP |

| Test Case | C.1.2 |
|---|---|
| **Description** | Simple message pull to test username/password authorization on the non-default MPC over HTTPS |
| **Required/Optional** | Required |
| **Payload** | filename=smallxmlPO.xml; SOAP Body |
| **User Message Security** | None |
| **Receipt** | None |
| **MPCs & Pull AuthZ** | Non-default MPC; Username/Password AuthZ |
| **MEP & Transport Bindings** | Pull over HTTPS |

| Test Case | D.1.1 |
|---|---|
| **Description** | Message push using a Username/Password security token over HTTP |
| **Required/Optional** | Required |
| **Payload** | filename=smallxmlPO.xml; SOAP Body |
| **User Message Security** | WSSE Username/Password token |
| **Receipt** | None |
| **MPCs & Pull AuthZ** | None |
| **MEP & Transport Bindings** | Push over HTTP |

| Test Case | D.1.2 |
|---|---|
| **Description** | Message push using SwA and a Username/Password security token over HTTP |
| **Required/Optional** | Required |
| **Payload** | filename=edifact.edi; SOAP w/Attachments |
| **User Message Security** | WSSE Username/Password token |
| **Receipt** | None |
| **MPCs & Pull AuthZ** | None |
| **MEP & Transport Bindings** | Push over HTTP |

| Test Case | D.2.1 |
|---|---|
| **Description** | Signed message push using a X.509 digital signature security token over HTTP |
| **Required/Optional** | Required |
| **Payload** | filename=smallxmlPO.xml; SOAP Body |
| **User Message Security** | WSSE X.509 digital certificate using XML Dsig |
| **Receipt** | None |
| **MPCs & Pull AuthZ** | None |
| **MEP & Transport Bindings** | Push over HTTP |

| Test Case | D.2.2 |
|---|---|
| **Description** | Signed message push using SwA and a X.509 digital signature security token over HTTP |
| **Required/Optional** | Required |
| **Payload** | filename=edifact.edi; SOAP w/Attachments |
| **User Message Security** | WSSE X.509 digital certificate using XML Dsig |
| **Receipt** | None |
| **MPCs & Pull AuthZ** | None |
| **MEP & Transport Bindings** | Push over HTTP |

| Test Case | D.3.1 |
|---|---|
| **Description** | Message pull to test X.509 digital signature authorization on a non-default MPC over HTTP |
| **Required/Optional** | Required |
| **Payload** | filename=smallxmlPO.xml; SOAP Body |
| **User Message Security** | WSSE X.509 digital certificate using XML Dsig |
| **Receipt** | None |
| **MPCs & Pull AuthZ** | Non-default MPC; X.509 digital certificate using XML Dsig as MPC AuthZ |
| **MEP & Transport Bindings** | Pull over HTTP |

| Test Case | D.3.2 |
|---|---|
| **Description** | Message pull using SwA to test X.509 digital signature authorization on a non-default MPC over HTTP |
| **Required/Optional** | Required |
| **Payload** | filename=edifact.edi; SOAP w/Attachments |
| **User Message Security** | WSSE X.509 digital certificate using XML Dsig |
| **Receipt** | None |
| **MPCs & Pull AuthZ** | Non-default MPC; X.509 digital certificate using XML Dsig as MPC AuthZ |
| **MEP & Transport Bindings** | Pull over HTTP |

| Test Case | E.1.1 |
|---|---|
| **Description** | Message push using a Username/Password security token over HTTP with receipt on the backchannel and reception awareness turned on. |
| **Required/Optional** | Required |
| **Payload** | filename=smallxmlPO.xml; SOAP Body |
| **User Message Security** | WSSE Username/Password token |
| **Receipt** | Synchronous Receipt w/o NRR |
| **MPCs & Pull AuthZ** | None |
| **MEP & Transport Bindings** | Push over HTTP |

| Test Case | E.1.2 |
|---|---|
| **Description** | Signed message push using a X.509 digital signature security token over HTTP with receipt on the backchannel and reception awareness turned on and NRR. |
| **Required/Optional** | Required |
| **Payload** | filename=smallxmlPO.xml; SOAP Body |
| **User Message Security** | WSSE X.509 digital certificate using XML Dsig |
| **Receipt** | Synchronous Receipt with NRR |
| **MPCs & Pull AuthZ** | None |
| **MEP & Transport Bindings** | Push over HTTP |

| Test Case | E.2.1 |
|---|---|
| **Description** | Message pull to test X.509 digital signature authorization on a non-default MPC over HTTP with receipt on the callback channel and reception awareness turned on and NRR |
| **Required/Optional** | Required |
| **Payload** | filename=smallxmlPO.xml; SOAP Body |
| **User Message Security** | WSSE X.509 digital certificate using XML Dsig |
| **Receipt** | Asynchronous Receipt with NRR |
| **MPCs & Pull AuthZ** | Non-default MPC; X.509 digital certificate using XML Dsig as MPC AuthZ |
| **MEP & Transport Bindings** | Pull over HTTP |

| Test Case | E.2.2 |
|---|---|
| **Description** | Message pull to test X.509 digital signature authorization on a non-default MPC over HTTP with a bundled receipt on the callback channel and reception awareness turned on and NRR. Receipt is bundled on a callback with a PullRequest for another UserMessage. |
| **Required/Optional** | Optional / Not Tested |
| **Payload** | filename=smallxmlPO.xml; SOAP Body |
| **User Message Security** | WSSE X.509 digital certificate using XML Dsig |
| **Receipt** | Bundled asynchronous Receipt with NRR |
| **MPCs & Pull AuthZ** | Non-default MPC; X.509 digital certificate using XML Dsig as MPC AuthZ |
| **MEP & Transport Bindings** | Pull over HTTP |

| Test Case | E.2.3 |
|---|---|
| **Description** | Message pull to test Basic username/password authorization on the default MPC over HTTP with receipt on the callback channel and reception awareness turned on and NRR |
| **Required/Optional** | Required |
| **Payload** | filename=smallxmlPO.xml; SOAP Body |
| **User Message Security** | WSSE X.509 digital certificate using XML Dsig |
| **Receipt** | Asynchronous Receipt with NRR |
| **MPCs & Pull AuthZ** | Default MPC channel; username/password as MPC AuthZ |
| **MEP & Transport Bindings** | Pull over HTTP |

| Test Case | E.3.1 |
|---|---|
| **Description** | Message pull to test alternate SSL client authorization on a non-default MPC over HTTPS with receipt on the callback channel |
| **Required/Optional** | Optional / Not tested |
| **Payload** | filename=smallxmlPO.xml; SOAP Body |
| **User Message Security** | WSSE X.509 digital certificate using XML Dsig |
| **Receipt** | Asynchronous Receipt w/o NRR |
| **MPCs & Pull AuthZ** | Non-default MPC; SSL Client Authentication as MPC AuthZ |
| **MEP & Transport Bindings** | Pull over HTTPS |

| Test Case | E.3.2 |
|---|---|
| **Description** | Message pull to test username/password authorization on a MPC sub-channel over HTTP with receipt on the callback channel and reception awareness turned on and NRR |
| **Required/Optional** | Optional / Not tested |
| **Payload** | filename=smallxmlPO.xml; SOAP Body |
| **User Message Security** | None |
| **Receipt** | Asynchronous Receipt with NRR |
| **MPCs & Pull AuthZ** | MPC sub-channel; username/password as MPC AuthZ |
| **MEP & Transport Bindings** | Pull over HTTP |

| Test Case | E.4.1 |
|---|---|
| **Description** | Signed message push using a X.509 digital signature security token over HTTPS with receipt on the callback channel and reception awareness turned on and NRR. |
| **Required/Optional** | Required |
| **Payload** | filename=smallxmlPO.xml; SOAP Body |
| **User Message Security** | WSSE X.509 digital certificate using XML Dsig |
| **Receipt** | Asynchronous Receipt with NRR |
| **MPCs & Pull AuthZ** | None |
| **MEP & Transport Bindings** | Push over HTTPS |

| Test Case | E.4.2 |
|---|---|
| **Description** | Signed and encrypted message push using a X.509 digital signature security token over HTTP with receipt on the backchannel and reception awareness turned on and NRR. |
| **Required/Optional** | Required |
| **Payload** | filename=smallxmlPO.xml; SOAP Body |
| **User Message Security** | WSSE X.509 digital certificate using XML Dsig and XML Encryption |
| **Receipt** | Synchronous Receipt with NRR |
| **MPCs & Pull AuthZ** | None |
| **MEP & Transport Bindings** | Push over HTTP |

| Test Case | E.4.3 |
|---|---|
| **Description** | Signed and encrypted message push using a X.509 digital signature security token over HTTP with receipt on the callback channel and reception awareness turned on and NRR. |
| **Required/Optional** | Required |
| **Payload** | filename=edifact.edi; SOAP Body |
| **User Message Security** | WSSE X.509 digital certificate using XML Dsig and XML Encryption |
| **Receipt** | Asynchronous Receipt with NRR |
| **MPCs & Pull AuthZ** | None |
| **MEP & Transport Bindings** | Push over HTTP |

| Test Case | E.4.4 |
|---|---|
| **Description** | Signed and encrypted SwA message push using a X.509 digital signature security token over HTTP with receipt on the backchannel and reception awareness turned on and NRR. |
| **Required/Optional** | Required |
| **Payload** | filename=edifact.edi; SOAP w/Attachments |
| **User Message Security** | WSSE X.509 digital certificate using XML Dsig and XML Encryption |
| **Receipt** | Synchronous Receipt with NRR |
| **MPCs & Pull AuthZ** | None |
| **MEP & Transport Bindings** | Push over HTTP |

| Test Case | E.4.5 |
|---|---|
| **Description** | Signed and encrypted message push using a X.509 digital signature security token over HTTP with receipt on the callback channel and reception awareness turned on and NRR. |
| **Required/Optional** | Required |
| **Payload** | filename=edifact.edi; SOAP w/Attachments |
| **User Message Security** | WSSE X.509 digital certificate using XML Dsig and XML Encryption |
| **Receipt** | Asynchronous Receipt with NRR |
| **MPCs & Pull AuthZ** | None |
| **MEP & Transport Bindings** | Push over HTTP |

| Test Case | E.4.6 |
|---|---|
| **Description** | Signed and encrypted and compressed SwA message push using a X.509 digital signature security token over HTTP with receipt on the backchannel and reception awareness turned on and NRR. |
| **Required/Optional** | Required |
| **Payload** | filename=edifact.edi; SOAP w/Attachments |
| **User Message Security** | WSSE X.509 digital certificate using XML Dsig and XML Encryption |
| **Receipt** | Synchronous Receipt with NRR |
| **MPCs & Pull AuthZ** | None |
| **MEP & Transport Bindings** | Push over HTTP |

| Test Case | E.4.7 |
|---|---|
| **Description** | Signed and encrypted and compressed message push using a X.509 digital signature security token over HTTP with receipt on the callback channel and reception awareness turned on and NRR. |
| **Required/Optional** | Required |
| **Payload** | filename=edifact.edi; SOAP w/Attachments |
| **User Message Security** | WSSE X.509 digital certificate using XML Dsig and XML Encryption |
| **Receipt** | Asynchronous Receipt with NRR |
| **MPCs & Pull AuthZ** | None |
| **MEP & Transport Bindings** | Push over HTTP |

| Test Case | F.1.1 |
|---|---|
| **Description** | Signed large SwA message push using a X.509 digital signature security token over HTTP with receipt on the callback channel and reception awareness turned on and NRR. |
| **Required/Optional** | Required |
| **Payload** | filename=X12fiftymb.edi; SOAP w/Attachments |
| **User Message Security** | WSSE X.509 digital certificate using XML Dsig |
| **Receipt** | Asynchronous Receipt with NRR |
| **MPCs & Pull AuthZ** | None |
| **MEP & Transport Bindings** | Push over HTTP |

| Test Case | F.1.2 |
|---|---|
| **Description** | Large SwA message pull to test X.509 digital signature authorization on the non-default MPC over HTTP with receipt on the callback channel and reception awareness turned on and NRR |
| **Required/Optional** | Required |
| **Payload** | filename=X12fiftymb.edi; SOAP w/Attachments |
| **User Message Security** | WSSE X.509 digital certificate using XML Dsig |
| **Receipt** | Asynchronous Receipt with NRR |
| **MPCs & Pull AuthZ** | Non-default MPC; X.509 digital certificate using XML Dsig as MPC AuthZ |
| **MEP & Transport Bindings** | Pull over HTTP |

| Test Case | F.2.1 |
|---|---|
| **Description** | Signed multiple message payload push using a X.509 digital signature security token over HTTP with receipt on the callback channel and reception awareness turned on and NRR. |
| **Required/Optional** | Required |
| **Payload** | filename(s)=binaryPayload.jpg, edifact.edi. smallxmlPO.xml; SOAP w/Attachments |
| **User Message Security** | WSSE X.509 digital certificate using XML Dsig |
| **Receipt** | Asynchronous Receipt with NRR |
| **MPCs & Pull AuthZ** | None |
| **MEP & Transport Bindings** | Push over HTTP |

| Test Case | F.2.2 |
|---|---|
| **Description** | Multiple message payload pull to test X.509 digital signature authorization on the non-default MPC over HTTP with receipt on the callback channel and reception awareness turned on and NRR |
| **Required/Optional** | Required |
| **Payload** | filename(s)=binaryPayload.jpg, edifact.edi. smallxmlPO.xml; SOAP w/Attachments |
| **Message Security** | X.509 digital certificate using XML Dsig |
| **Receipt** | Asynchronous Receipt with NRR |
| **MPCs & Pull AuthZ** | Non-default MPC; X.509 digital certificate using XML Dsig as MPC AuthZ |
| **MEP & Transport Bindings** | Pull over HTTP |

# About Drummond Group

Drummond Group is the trusted interoperability test lab offering global testing services throughout the product life cycle. Auditing, QA, conformance testing, custom software test lab services, and consulting are offered in addition to interoperability testing. Founded in 1999, Drummond Group has tested over a thousand international software products used in vertical industries such as automotive, consumer product goods, healthcare, energy, financial services, government, petroleum, pharmaceutical and retail. For more information, please visit www.drummondgroup.com or email: info2@drummondgroup.com