



Intelligence Community Guidance Document

Whitelist Guidance for ISM

Version 2019-MAR

December 1, 2022

Distribution Notice:

This document has been approved for Public Release and is available for use without restriction.

Table of Contents

Chapter 1 - Introduction	1
1.1 - Purpose	1
1.2 - Scope	1
1.3 - Enterprise Need	1
1.4 - Conventions	1
1.4.1 - XML Namespaces	2
1.5 - Dependencies	2
1.5.1 - Specification Dependencies	2
1.5.2 - Inverse Dependencies	4
Chapter 2 - Development Guidance	5
2.1 - Understanding Whitelisting	5
2.2 - Testing Documents with Whitelist	5
2.3 - Whitelist Example Files	6
Chapter 3 - Definitions, Interfaces, and Constraints	7
3.1 - Data Validation Constraint Rules	7
3.1.1 - Additional Constraints	7
3.1.1.1 - DES Constraints	7
3.1.2 - Constraint Rules	7
Appendix A - Feature Summary	8
A.1 - Whitelist Feature Summary	8
Appendix B - Change History	9
B.1 - V2019-MAR Initial Release Summary	9
Appendix C - List of Abbreviations	10
Appendix D - Bibliography	11
Appendix E - Points of Contact	12

List of Figures

Figure 1 - Related Specifications	4
---	---

List of Tables

Table 1 - XML Namepaces	2
Table 2 - Dependencies	2
Table 3 - Feature Summary Legend	8
Table 4 - Whitelist Feature comparison	8
Table 5 - Version Identifier History	9
Table 6 - Data Encoding Specification V2019-MAR Initial Release Summary	9

Chapter 1 - Introduction

1.1 - Purpose

This *Whitelist Guidance for ISM* (Whitelist.XML) provides guidance on the use of a whitelist to prevent the ingestion of unauthorized documents by testing *XML Data Encoding Specification for Information Security Markings* (ISM.XML^[2]) markings. This implementation uses Schematron^[3] to determine if a given document meets the constraints given by a configuration file. Each system is expected to have one or more configuration files defining which ISM.XML^[2] values can be handled. The system may have additional configuration files to check which documents a downstream system is authorized to receive.

1.2 - Scope

The *Intelligence Community Technical Specification Framework* (IC-SF.XML^[1]) defines the basic conceptual structure and outlines the core philosophy of Intelligence Community (IC) technical specifications. For convenience, a copy of this framework is included in every package.

This information guidance document addresses general concepts of using Schematron^[3] to implement a whitelist capability for enforcing allowable security markings.

This specification is applicable to the IC and information produced by, stored, or shared within the IC. This document may have relevance outside the scope of intelligence; however, prior to applying outside of this defined scope, the Data Encoding Specification (DES) should be closely scrutinized and differences separately documented and assessed for applicability.

1.3 - Enterprise Need

The Intelligence Community Chief Information Officer (IC CIO) funds and oversees a number of critical enabling projects to allow interagency access control, automated exchanges, and appropriate protection of shared intelligence, information sharing within the national intelligence enterprise will increasingly rely on information assurance metadata including information security markings, enterprise data headers, and determination of an individual's need-to-know. A successful information sharing enterprise depends on the ability of the data creator and/or providers to specify the means by which need-to-know can be established in a manner to facilitate discovery and access via automated means.

This document provides general and prescriptive guidance for the use of whitelisting to validate that a system is authorized to handle the ISM.XML^[2] markings on a given Extensible Markup Language (XML) document.

1.4 - Conventions

Certain technical and presentation conventions are used in the creation of the IC technical specifications to ensure readability and understanding. For details, please see the "Specification Conventions" chapter in the IC-SF.XML^[1].

1.4.1 - XML Namespaces

Namespaces referenced in this document and the prefixes used to represent them are listed in the following table. The namespace prefix of any XML Qualified Name used in any example in this document should be interpreted using the information below.

Table 1 - XML Namespaces

Prefix	URI
ism	urn:us:gov:ic:ism
ntk	urn:us:gov:ic:ntk
xsd	http://www.w3.org/2001/XMLSchema

1.5 - Dependencies

Specifications often rely on other specifications, components or artifacts, either directly or indirectly. For specific definitions of dependency terminology used throughout this section, please see the “Dependency Definitions” chapter in the IC-SF.XML^[1].

1.5.1 - Specification Dependencies

This technical specification directly depends on the technical specifications, documentation, and implementations listed in [Table 2](#). The dependencies listed below are directly referenced in this specification (e.g., Schema, Schematron), and are normative or informative as indicated.

The subsequent figure, [Figure 1](#), is an informative graphical representation of all of the IC CIO specifications related to this specification. The graphic depicts dependencies. However, the representations may not match an exact schema import tree or dependency diagram that an analysis of the Schema, Schematron or other documents would yield. For example, the graphic only shows a given specification once even though it may actually be imported by many specifications or be a direct dependency. All IC CIO specifications listed in [Table 2](#) will be shown in [Figure 1](#); however not all IC CIO specifications listed in [Figure 1](#) may appear in [Table 2](#). [Figure 1](#) is to aid users in gaining a general understanding of all dependencies whether direct or transitive.

Table 2 - Dependencies

Name	Dependency Description
<i>XML Data Encoding Specification for Information Security Marking Metadata</i> (ISM.XML.V2019-MAR+ ^[2])	This specification does not depend on a specific version of ISM.XML ^[2] ; versions later than version v2019-MAR MAY be used. The minimum version was based on technical dependencies.

Name	Dependency Description
<i>Intelligence Community Specification Framework</i> (IC-SF.XML.V2019-MAR+ ^[1])	<p>This specification does not depend on a specific version of IC-SF.XML^[1]; versions later than version 2019-MAR MAY be used, however, the newest version of IC-SF.XML SHOULD be used as IC-SF.XML is expected to always replace its preceding version. The minimum version was based on technical dependencies on IC-SF.XML; IC-SF.XML is the basic structure of and philosophy behind intelligence community technical specifications.</p>
Schematron ^[3]	<p>Schematron — International Organization for Standardization (ISO)/International Electrotechnical Commission (IEC) 19757-3:2006 — is a rule-based document schema definition language. In this specification Schematron is a formal language used to express normative business rules, so this reference is normative.</p> <p>The Schematron rules are normative in the sense that they convey criteria that a document MUST adhere to, exactly as English may be used to convey normative criteria. It is not necessary for implementers to use the specific Schematron encoding in this specification. Implementers MAY use any encodings, tools, or languages desired to implement validation schemes for conformance to this specification.</p> <p>Note: The Schematron rules in this specification use Transformations (XSLT) 2.0^[5] query binding.</p>
<p>XSLT 2.0^[5] implementation of Schematron^[3] by Rick Jelliffe (2010-04-14)</p> <p>Note: The only available identifying descriptors for this implementation are the implementer's name and date of release. This implementation may be found at the following Uniform Resource Locator (URL): http://code.google.com/p/schematron/.</p>	<p>The International Organization for Standardization does not create nor endorse reference implementations of its standards. For the purposes of this specification the <i>behavior</i> of the implementation created by Mr. Jelliffe is normative.</p> <p>Implementers MAY use any encodings, tools, or languages desired to implement validation schemes for conformance to this specification. To conform to this specification, a validator MUST find a document valid <i>if and only if</i> the Schematron implementation by Mr. Jelliffe would find the document valid according to the Schematron rules in this specification.</p>

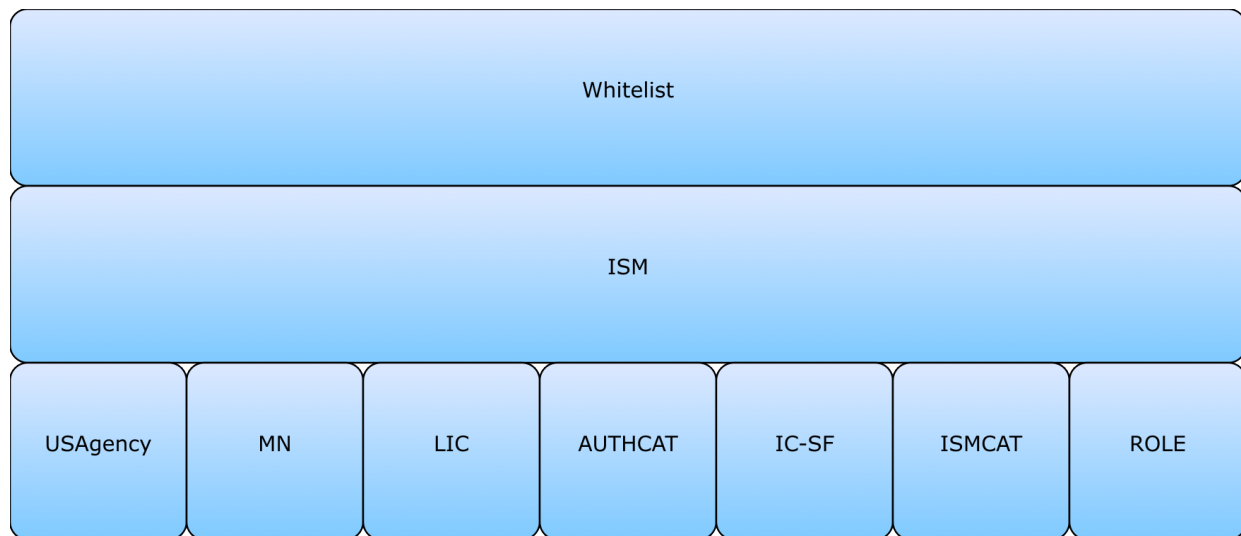


Figure 1 : Related Specifications

1.5.2 - Inverse Dependencies

Generally, it is only necessary to think of the *dependencies* in the dependency tree. However, with the specification versions being decoupled, it is also important to consider the *inverse dependencies*, for compatibility with newer versions of a given specification. The changes introduced to a given specification can sometimes make it incompatible with current versions of its inverse dependencies (specifications that uses the given specification).

This specification is not used by other specifications released by the IC CIO, and therefore does not contain an Inverse Dependency Diagram.

Chapter 2 - Development Guidance

For information on the structure and content of the specifications, please see the “Specification Overview” chapter in the IC-SF^[1] framework document. This chapter is intended to expand upon the common information that the framework specifies providing specific development guidance that is specific to the implementation of this guidance document.

2.1 - Understanding Whitelisting

The IC Whitelist rules support whitelisting, whitelisting with minimum required values, and blacklisting.

Whitelisting is the process of identifying attributes that are recognized and supported; if an attribute on a document does not exist in the whitelist configuration file, then the document will fail business rule validation. The whitelist configurations are implemented with schematron^[3]. Whitelisting is preferred to blacklisting because it can protect a system against ingesting documents with new markings that could potentially result in a data spill. Some ISM.XML^[2] attributes contain a large number of values (e.g. lists of countries). For these attributes, rather than creating a long whitelist entry listing each value, the IC Whitelist supports the use of a wildcard character (“*”) to accept any value.

Whitelisting with minimum required values operates on the same principal except that it will test a document to ensure that it contains a minimum set of values for a particular element. For example, a whitelist for a Five-Eyes Enterprise (5EE) system may want to ensure that the **@ism:releasableTo** attribute contains at least all five partner countries. The 5EE system could allow any country in **@ism:releasableTo**, as long as the 5 member countries are included. It is also possible to use the whitelist with a wild card and minimum value list.

Blacklisting is the opposite of whitelisting; any value will be accepted unless it appears in the blacklist. This approach is generally less desirable than a whitelist because it will protect against known conditions but not against unanticipated values. In the interest of avoiding long entries in the configuration file, blacklisting has been allowed for certain attributes that list countries and organization tetragraphs. For instance, **@ism:displayOnly** can be handled with a blacklist if the system is not accredited to handle North Atlantic Treaty Organization (NATO) data, but has no additional restrictions.

Any document that fails whitelist validation should be quarantined and investigated. For incoming documents, the investigation should determine if the configuration file is not defined correctly, the document is not marked correctly, or if the submitting system has spilled. For outgoing documents, the investigation should determine if the configuration file is not defined correctly, the document is not marked correctly, or why the user or system was attempting to send data that the intended recipient is not authorized to receive.

2.2 - Testing Documents with Whitelist

The master Whitelist schematron^[3] file, “Whitelist_XML.sch” is configured to read whitelist definitions from a configuration file called **whitelist_config.xml**. The “whitelist_config.xml” file has been included in this package with notional values that were used to test functionality of initial release. The notional values should be replaced with values that are allowed in the system implementing this whitelist.

The "Whitelist_XML.sch" is configured to read the whitelist_config.xml from the same directory. If the implementing system requires the configuration file to be located in another directory, the relative path defined by the "configXML" variable in the "Whitelist_XML.sch" file should be updated to reflect the new file path.

2.3 - Whitelist Example Files

The Whitelist example folder contains two example configuration files and a set of XML files that are designed to either pass or fail based upon the configurations chosen to test. Some conditions require modifications to the configuration file in order to achieve the desired result. For instance, there is a passing example ("Whitelist-LIC-PASS-001.xml") and a failing example ("Whitelist-LIC-FAIL-001.xml") to test the Need-To-Know Metadata (NTK) Datashpere License AccessProfile. Since there is only one possible value to test, it is necessary to add or remove the value from the configuration file to get the tests to pass or fail. Each example file contains a comment at the top of the file indicating which configuration file is used for testing and any additional steps required to get the intended result.

Chapter 3 - Definitions, Interfaces, and Constraints

3.1 - Data Validation Constraint Rules

The Whitelist.XML schema defines the data elements, attributes, cardinalities and parent-child relationships for which the configuration XML instances must comply. Validation of these syntax aspects is an important first step in the validation process. An additional level of validation is needed to ensure that the content complies with the constraints as specified in applicable IC policy guidance and codified in these constraint rules. Traditional schema languages are generally unable to effectively represent these additional constraints. For more information, please see the “Data Validation Constraint Rules” chapter in the IC-SF.XML^[1] framework document.

3.1.1 - Additional Constraints

3.1.1.1 - DES Constraints

The DES version is specified through attributes on the root element. The schema constrains the values of these attributes. The **@DESVersion** attribute enables systems processing an instance document to be certain which set of constraint rules, schema, Controlled Vocabulary Enumeration (CVE), and business rules are intended by the author to be used.

3.1.2 - Constraint Rules

The detailed constraint rules for the Whitelist.XML schema can be found in a separate document inside the Documents/Whitelist directory, in the “Whitelist_Rules.pdf” file. This document is generated from the individual Schematron^[3] files to provide a single searchable document for all of the constraint rules encoded in Schematron^[3]. Obsolete rule numbers are listed in the “Whitelist_Rules.pdf” file.

Appendix A Feature Summary

The following table summarizes major features by version for this [Whitelist.XML^{\[4\]}](#) and all dependent specs.

Table 3 - Feature Summary Legend

Key	Description
F	Full (able to comply and verified by spec to some degree)
P	Partial (Able to comply but not verifiable)
N	Non-compliance (Can't comply)
N/A	Not Applicable. Feature is no longer required.
Cell Colors represent the same information as the Key value	

A.1. Whitelist Feature Summary

Table 4 - Whitelist Feature comparison

Required date	Feature	V2019-MAR
March 8, 2019	Defines the initial capabilities for implementing whitelist for ISM attributes and NTK access profiles. Defines which ISM attributes are allowed, which values for ISM attributes are allowed and which NTK access profiles are allowed. Allows the blacklisting of attributes that contain country lists. Allows the use of the '*' wild card to allow any value for a given attribute.	F

Appendix B Change History

The following table summarizes the version identifier history for this document.

Table 5 - Version Identifier History

Version	Date	Purpose
2019-MAR	March 8, 2019	Initial Release. For details, see Section B.1 - V2019-MAR Initial Release Summary

B.1 - V2019-MAR Initial Release Summary

Significant drivers for Version V2019-MAR include:

- Creation of Whitelist specification.

The following table summarizes the initial release in V2019-MAR.

Table 6 - Data Encoding Specification V2019-MAR Initial Release Summary

#	Change	Artifacts changed	Compatibility Notes
	Creation of Whitelist specification.(CR-2016-088)	Documentation Schema Schematron	Initial Release.

Appendix C List of Abbreviations

This appendix lists all the acronyms and abbreviations referenced in this encoding specification.

5EE	Five-Eyes Enterprise
CVE	Controlled Vocabulary Enumeration
DES	Data Encoding Specification
DNI	Director of National Intelligence
IC	Intelligence Community
IC CIO	Intelligence Community Chief Information Officer
IEC	International Electrotechnical Commission
ISO	International Organization for Standardization
NATO	North Atlantic Treaty Organization
NTK	Need-To-Know Metadata
URL	Uniform Resource Locator
XML	Extensible Markup Language
XSL	Extensible Stylesheet Language
XSLT	XSL Transformations

Appendix D Bibliography

[1] IC-SF.XML

Office of the Director of National Intelligence. *Intelligence Community Specification Framework (IC-SF.XML)*.

Available online Intelink-TS at: <https://go.ic.gov/pNFyuVg> (case sensitive – papa November Foxtrot yankee uniform Victor golf)

Available online Intelink-U at: <https://w3id.org/ic/standards/IC-SF>

Available online at: <https://w3id.org/ic/standards/public>

[2] ISM.XML

Office of the Director of National Intelligence. *XML Data Encoding Specification for Information Security Markings (ISM.XML)*.

Available online Intelink-TS at: <https://go.ic.gov/qoNICy7> (case sensitive – quebec oscar November India Charlie yankee 7)

Available online Intelink-U at: <https://w3id.org/ic/standards/ISM>

Available online at: <https://w3id.org/ic/standards/public>

[3] Schematron

International Organization for Standardization (ISO). *Information technology -- Document Schema Definition Language (DSDL) -- Part 3: Rule-based validation -- Schematron*. ISO/IEC 19757-3:2006.

ISO Spec Available online at: <http://standards.iso.org/ittf/PubliclyAvailableStandards/index.html>

StyleSheets for compiling Available online at: <http://code.google.com/p/schematron/>

[4] WHITELIST.XML

Office of the Director of National Intelligence. *Whitelist Guidance for ISM (WHITELIST.XML)*.

Available online Intelink-TS at: <https://go.ic.gov/g5ATj66> (case sensitive – golf 5 Alpha Tango juliet 6 6)

Available online Intelink-U at: <https://w3id.org/ic/standards/WHITELIST>

Available online at: <https://w3id.org/ic/standards/public>

[5] XSLT2

World Wide Web Consortium (W3C). *XSL Transformations (XSLT) Version 2.0*. W3C Recommendation 23 January 2007.

Available online at: <http://www.w3.org/TR/xslt20/>

Appendix E Points of Contact

The Intelligence Community Chief Information Officer (IC CIO) facilitates one or more collaboration and coordination forums charged with the adoption, modification, development, and governance of IC technical specifications of common concern. This technical specification was produced by the IC CIO and coordinated with these forums, approved by the IC CIO or a designated representative, and made available at the following Director of National Intelligence (DNI)-sponsored web sites.

Public Website: <https://w3id.org/ic/standards/public>

Intelshare: <https://w3id.org/ic/standards/data-specs>

Direct all inquiries about this IC technical specification, IC technical specification collaboration and coordination forums, or IC element representatives involved in those forums, to the IC CIO.

E-mail: ic-standards-support@odni.gov.