Date: June 2019

Information Exchange Framework Reference Architecture

Beta 2

OMG Document Number: ptc/2019-06-07

Release Date:

Standard document URL: http://www.omg.org/spec/IEFRA/

Additional Document:

ptc/2018-05-20 - IEF RA Supplementary Information.docx

Machine Consumable File(s):

ptc/2019-02-17 - IEF ISMB messages XSDs.zip

ptc/2019-02-29 - IEF PPS Policy XSDs.zip

Comments on the content of this document are welcome and should be directed to issues@omg.org . You may view the pending issues for this specification from the OMG revision issues web page http://www.omg.org/issues/.

USE OF SPECIFICATION - TERMS, CONDITIONS & NOTICES

The material in this document details an Object Management Group specification in accordance with the terms, conditions and notices set forth below. This document does not represent a commitment to implement any portion of this specification in any company's products. The information contained in this document is subject to change without notice.

LICENSES

The companies listed above have granted to the Object Management Group, Inc. (OMG) a nonexclusive, royalty-free, paid up, worldwide license to copy and distribute this document and to modify this document and distribute copies of the modified version. Each of the copyright holders listed above has agreed that no person shall be deemed to have infringed the copyright in the included material of any such copyright holder by reason of having used the specification set forth herein or having conformed any computer software to the specification.

Subject to all of the terms and conditions below, the owners of the copyright in this specification hereby grant you a fully-paid up, non-exclusive, nontransferable, perpetual, worldwide license (without the right to sublicense), to use this specification to create and distribute software and special purpose specifications that are based upon this specification, and to use, copy, and distribute this specification as provided under the Copyright Act; provided that: (1) both the copyright notice identified above and this permission notice appear on any copies of this specification; (2) the use of the specifications is for informational purposes and will not be copied or posted on any network computer or broadcast in any media and will not be otherwise resold or transferred for commercial purposes; and (3) no modifications are made to this specification. This limited permission automatically terminates without notice if you breach any of these terms or conditions. Upon termination, you will destroy immediately any copies of the specifications in your possession or control.

PATENTS

The attention of adopters is directed to the possibility that compliance with or adoption of OMG specifications may require use of an invention covered by patent rights. OMG shall not be responsible for identifying patents for which a license may be required by any OMG specification, or for conducting legal inquiries into the legal validity or scope of those patents that are brought to its attention. OMG specifications are prospective and advisory only. Prospective users are responsible for protecting themselves against liability for infringement of patents.

GENERAL USE RESTRICTIONS

Any unauthorized use of this specification may violate copyright laws, trademark laws, and communications regulations and statutes. This document contains information which is protected by copyright. All Rights Reserved. No part of this work covered by copyright herein may be reproduced or used in any form or by any means--graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems--without permission of the copyright owner.

NON-ASSERTION COVENANT FOR OMG FORMAL SPECIFICATION

Obligation. The OMG Obligated Parties who have incurred a Participation Obligation or a Contribution Obligation to this OMG Formal Specification (See OMG IP Policy Statement for the definitions of capitalized terms) provide the following Non-Assertion Covenant for the benefit of any Implementer of this OMG Formal Specification. Solely upon implementation of this OMG Formal Specification, you and your Affiliates, as “Implementers” accept and are bound by this agreement and provide the same Covenant covering your Essential Claims to all other Implementers of this OMG Formal Specification as if you were an OMG Obligated Party. If you do not agree, do not implement this OMG Formal Specification. *For the avoidance of doubt, you may download this OMG Formal Specification for inspection only without granting reciprocal rights.*

Non-Assert. OMG Member promises irrevocably not to assert its Essential Claims against anyone for making, using, selling, offering for sale, importing, or distributing Covered Implementations of the OMG Formal Specification identified above.

Personal Promise. This is a personal promise; by receiving any benefits of this Covenant, Implementer acknowledges that the benefits are received directly from OMG Member.

Suspension. No rights, grants, or promises are made under this Covenant as to any party that files, maintains, or voluntarily joins any lawsuit asserting that an implementation of the OMG Formal Specification infringes any Essential Claims. (This provision does not apply to a counterclaim or countersuit to a suit for infringement of Essential Claims.)

Continuation. This Covenant is intended to bind any future owner, assignee, or exclusive licensee who is given the right to enforce any Essential Claims against third parties, provided that all OMG Member obligations under this Paragraph are satisfied if the OMG Member notifies the transferee or assignee (or the transferee or assignee is otherwise made aware) of the obligations under this Covenant with respect to any patent that the OMG Member knows contains Essential Claims.

Licenses. In lieu of this Covenant, upon request by any Implementer OMG Member will provide a written license in accordance with its obligations under the OMG IPR Policy.

No Other Rights. The rights granted are only those expressly stated in this Covenant; no other rights of any kind are granted by implication, waiver, estoppel, or otherwise.

Disclaimers. This Covenant does not imply that any OMG Member has any Essential Claims, nor that it covers all intellectual property rights in the OMG Formal Specification or rights held by any third party. Nothing in this Covenant requires any party to undertake a patent search.

No Warranties. THE OMG FORMAL SPECIFICATION IS PROVIDED "AS IS". The entire risk of using or implementing the OMG Formal Specification is assumed by the Implementer and user; OMG and OMG Members expressly disclaim any warranties (express, implied, or otherwise), including implied warranties of merchantability, non-infringement, fitness for a particular purpose, or title, relating to the OMG Formal Specification.

DISCLAIMER OF WARRANTY

WHILE THIS PUBLICATION IS BELIEVED TO BE ACCURATE, IT IS PROVIDED "AS IS" AND MAY CONTAIN ERRORS OR MISPRINTS. THE OBJECT MANAGEMENT GROUP AND THE COMPANIES LISTED ABOVE MAKE NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS PUBLICATION, INCLUDING BUT NOT LIMITED TO ANY WARRANTY OF TITLE OR OWNERSHIP, IMPLIED WARRANTY OF MERCHANTABILITY OR WARRANTY OF FITNESS FOR A PARTICULAR PURPOSE OR USE. IN NO EVENT SHALL THE OBJECT MANAGEMENT GROUP OR ANY OF THE COMPANIES LISTED ABOVE BE LIABLE FOR ERRORS CONTAINED HEREIN OR FOR DIRECT, INDIRECT, INCIDENTAL, SPECIAL, CONSEQUENTIAL, RELIANCE OR COVER DAMAGES, INCLUDING LOSS OF PROFITS, REVENUE, DATA OR USE, INCURRED BY ANY USER OR ANY THIRD PARTY IN CONNECTION WITH THE FURNISHING, PERFORMANCE, OR USE OF THIS MATERIAL, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

The entire risk as to the quality and performance of software developed using this specification is borne by you. This disclaimer of warranty constitutes an essential part of the license granted to you to use this specification.

RESTRICTED RIGHTS LEGEND

Use, duplication or disclosure by the U.S. Government is subject to the restrictions set forth in subparagraph (c) (1) (ii) of The Rights in Technical Data and Computer Software Clause at DFARS 252.227-7013 or in subparagraph (c)(1) and (2) of the Commercial Computer Software - Restricted Rights clauses at 48 C.F.R. 52.227-19 or as specified in 48 C.F.R. 227-7202-2 of the DoD F.A.R. Supplement and its successors, or as specified in 48 C.F.R. 12.212 of the Federal Acquisition Regulations and its successors, as applicable. The specification copyright owners are as indicated above and may be contacted through the Object Management Group, 109 Highland Avenue, Needham, MA 02494, U.S.A.

TRADEMARKS

IMM®, MDA®, Model Driven Architecture®, UML®, UML Cube logo®, OMG Logo®, CORBA® and XMI® are registered trademarks of the Object Management Group, Inc., and Object Management Group™, OMG™ , Unified Modeling Language™, Model Driven Architecture Logo™, Model Driven Architecture Diagram™, CORBA logos™, XMI Logo™, CWM™, CWM Logo™, IIOP™ , MOF™ , OMG Interface Definition Language (IDL)™ , and OMG SysML™ are trademarks of the Object Management Group. All other products or company names mentioned are used for identification purposes only, and may be trademarks of their respective owners.

COMPLIANCE

The copyright holders listed above acknowledge that the Object Management Group (acting itself or through its designees) is and shall at all times be the sole entity that may authorize developers, suppliers and sellers of computer software to use certification marks, trademarks or other special designations to indicate compliance with these materials.

Software developed under the terms of this license may claim compliance or conformance with this specification if and only if the software compliance is of a nature fully matching the applicable compliance points as stated in the specification. Software developed only partially matching the applicable compliance points may claim only that the software was based on this specification, but may not claim compliance or conformance with this specification. In the event that testing suites are implemented or approved by Object Management Group, Inc., software developed using this specification may claim compliance or conformance with the specification only if the software satisfactorily completes the testing suites.

OMG’s Issue Reporting Procedure

All OMG specifications are subject to continuous review and improvement. As part of this process we encourage readers to report any ambiguities, inconsistencies, or inaccuracies they may find by completing the Issue Reporting Form listed on the main web page http://www.omg.org, under Documents, Report a Bug/Issue (http://www.omg.org/report_issue.)

Table of Contents

List of Figures

List of Tables

Preface

About the Object Management Group

Founded in 1989, the Object Management Group, Inc. (OMG) is an open membership, not-for-profit computer industry standards consortium that produces and maintains computer industry specifications for interoperable, portable, and reusable enterprise applications in distributed, heterogeneous environments. Membership includes Information Technology vendors, end users, government agencies, and academia.

OMG member companies write, adopt, and maintain its specifications following a mature, open process. OMG's specifications implement the Model Driven Architecture® (MDA®), maximizing Return on Investment (ROI) through a full-lifecycle approach to enterprise integration that covers multiple operating systems, programming languages, middleware and networking infrastructures, and software development environments. OMG's specifications include: UML® (Unified Modeling Language™); CORBA® (Common Object Request Broker Architecture); CWM™ (Common Warehouse Metamodel); and industry-specific standards for dozens of vertical markets.

More information on the OMG is available at http://www.omg.org/.

OMG Specifications

As noted, OMG specifications address middleware, modeling and vertical domain frameworks. All OMG Specifications are available from the OMG website at:

http://www.omg.org/spec

Specifications are organized by the following categories:

Business Modeling Specifications

Middleware Specifications

- **CORBA/IIOP**
- **Data Distribution Services**
- **Specialized CORBA**

IDL/Language Mapping Specifications

Modeling and Metadata Specifications

- **UML, MOF, CWM, XMI**
- **UML Profile**

Modernization Specifications

Platform Independent Model (PIM), Platform Specific Model (PSM), Interface Specifications

- **CORBAServices**
- **CORBAFacilities**

OMG Domain Specifications

CORBA Embedded Intelligence Specifications

CORBA Security Specifications

All of OMG's formal specifications may be downloaded without charge from our website. (Products implementing OMG specifications are available from individual suppliers.) Copies of specifications, available in PostScript and PDF format, may be obtained from the Specifications Catalog cited above or by contacting the Object Management Group, Inc. at:

OMG Headquarters
109 Highland Ave,
Needham, MA 02494 USA
Tel: +1-781-444-0404
Fax: +1-781-444-0320
Email: *pubs@omg.org*

Certain OMG specifications are also available as ISO standards. Please consult **http://www.iso.org**

Typographical Conventions

The type styles shown below are used in this document to distinguish programming statements from ordinary English. However, these conventions are not used in tables or clause headings where no distinction is necessary.

Times/Times New Roman - 10 pt.: Standard body text.

Helvetica/Arial - 10 pt. Bold: OMG Interface Definition Language (OMG IDL) and syntax elements.

Courier - 9 pt. Bold: Programming language elements.

Helvetica/Arial – 9 pt.: XML Schema

Helvetica/Arial - 10 pt: Exceptions.

NOTE: Terms that appear in italics are defined in the glossary. Italic text also represents the name of a document, specification, or other publication.

Issues

The reader is encouraged to report any technical or editing issues/problems with this specification to http://www.omg.org/report_issue.htm.

1 Information Exchange Framework Reference Architecture

1.1 Scope

The Information Exchange Framework (IEF) is an OMG initiative to develop a family of specifications for policy-driven, data-centric information sharing and safeguarding (ISS) services. These services target the automation of key policy decision and enforcement points to enable responsible information sharing supporting real-time situational awareness across a broad range of operational scenarios. The IEF Reference Architecture (RA) guides the overall IEF effort, broadens general understanding of domain requirements, and guides the development of ISS solutions.

The IEF RA is primarily targeting operational environments that require the ability and capacity to share information within and beyond organizational boundaries (public and private sectors) and are challenged by rapid, unpredictable changes in operational contexts (e.g., threat, risk, roles & responsibilities, scale, scope, and severity). These include:

- Military (coalition and Civilian-Military) operations;
- National Security;
- Public Safety;
- Crisis Management;
- Border Security;
- Emergency Management;
- Peace Keeping; and
- Humanitarian Assistance.

Missions and operations in each of these operational domains typically have the requirement to share sensitive (private, confidential and classified) information with other agencies, other levels of government, private and international partners. This is coupled with the requirement to:

- Address planned and non-planned missions and operations;
- Rapidly deploy and integrate a coalition/partner ISS capability;
- Rapidly adapt ISS patterns to changing operational conditions and contexts:
    - Commander’s intent (target outcomes);
    - Coalition configuration, organizational structure, roles and responsibilities;
    - Threats, risk and severity;
    - Operational stage;
    - Plans and orders; and
    - Communication capacity.

Although these environments are the primary focus of the IEF specifications, most of the defined features could equally support the transactional domains of a broad range of public and private sector organizations that require:

1. The ability to exchange information in a secure and trusted manner with clients, partners and subcontractors;
2. The ability to selectively share elements of an information holding with individuals and agencies in conformance with legislation regulation, and policy, e.g.:

    a. Healthcare: Electronic Health Records (EHR);
    b. Finance: Banking and Insurance records;
    c. Justice: Criminal Case Files;
    d. Government Programs; and
    e. Customs and Immigration.

3. The ability to log and audit the exchange of information holdings.

The IEF initiative will, through adoption or development of a series of open standards, define:

1. A service integration layer that integrates user specified security components into a Policy-driven Data-centric Information Sharing and Safeguarding capability;
2. Integration layer components including:
    a. Policy Administration Points (PAP);
    b. Policy Enforcement Points (PEP);
    c. Policy Decision Points (PDP);
    d. Policy-based Packaging Services (PPS);
    e. Secure Messaging Bus (SMB); and
    f. Security Service Gateway (SSG).
3. IEF Policy Vocabularies, including:
    a. Decision Request and Response Vocabulary;
    b. Packaging Policy Vocabulary (PPV);
    c. Decision Policy Vocabulary (DPV); and
4. Policy Development and Management Environment.

1.2 Organization of this Reference Architecture

This Reference Architecture includes sixteen (16) Clauses and five (5) Annexes:

- Clause 1: Provides an overview of the Reference Architecture.
- Clause 2: Defines the compliance points for the IEF Reference Architecture (RA).
- Clause 3: Identifies *Normative References* for this Reference Architecture.
- Clause 4: Identifies *Terms and their Definitions* used in various parts of the Reference Architecture. This Clause does not include concepts and properties comprising the IEF RA.
- Clause 5: Identifies any special *Symbols/Acronyms* used in the development of this reference architecture.
- Clause 6: Provides *Additional Information* about this reference architecture.
- Clause 7: Provides an Overview of the IEF Reference Architecture.
- Clause 8: Policy Administration Point (PAP).
- Clause 9: Policy Decision Point (PDP).
- Clause 10: Policy Enforcement Points (PEP), including:
    - Messaging-PEP;

    - File-Share-PEP;
    - Email-PEP; and
    - Instant Messaging-PEP.
- Clause 11: Policy-based Packaging and Processing Services (PPS).
- Clause 12: The Security Services Gateway (ISSG).
- Clause 13: Cryptographic Transformation Service (CTS).
- Clause 14: Secure Messaging Bus (ISMB).
- Clause 15: Trusted (Tamper Resistant) Logging Service (TLS).
- Clause 16: ISMB Messages and data patterns.
- Annex A: ISMB Message XSDs
- Annex B: Policy Model XSDs (Informational).
- Annex C: Enumerations (Informational).
- Annex D: Sequence Diagrams (Informational).
- Annex E: Glossary (Informational).

1.3 Motivation

Numerous after-action and news reports on events such as SARS, the 2005 London Underground bombing, the 1998 Ice Storm (Eastern Canada and Northern New York State), Haiti, Afghanistan, Katrina, and 9/11 events have documented the challenges faced by even the most technologically advanced agencies to effectively and efficiently interoperate with partners. Equally prevalent are the reports documenting the growing need for agencies to increase the quantity and quality of information they share with partners when responding to an emergency or crisis situation; e.g.:

> *"Today, information sharing is critical to almost every institution. There is no more critical need for information sharing than during an international crisis, when international coalitions dynamically form. In the event of a crisis, whether it is humanitarian relief, natural disaster, combat operations, or terrorist incidents, international coalitions have an immediate need for information. These coalitions are formed with international cooperation, where each participating country offers whatever resources it can muster to support the given crisis. These situations can occur suddenly, simultaneously, and without warning. Often times, participants are coalition partners in one crisis and adversaries in another, raising difficult security issues with respect to information sharing."*[1]

Each participating agency requires the ability to rapidly establish pre-planned or *ad hoc* ISS capabilities to enable:

- Shared situational awareness;
- Collaboration (e.g., operational planning and intelligence);
- Coordination; and
- Command-and-Control.

Operational users tend to emphasize the need to share information and maximize the volume, variety and quality of information discoverable and accessible by authorized users and partners. They recognize information is vital to the

[1] Charles E. Phillips, Jr. et al, SACMAT '02 Proceedings of the seventh ACM symposium on Access control models and technologies, Pages 87-96, http://dl.acm.org/citation.cfm?doid=507711.507726

formation, quality, and timeliness of decisions, as well as the creation of decision advantage (/decision superiority). Conversely, Security and Privacy Officers, representing data owners, stewards, and custodians, apply and emphasize need-to-know practices and principles, which ensure that only users with the appropriate credentials, authorizations and need are provided access to designated information elements. These need-to-know practices result in the development and deployment of multiple self-contained enclaves based on security level and warning terms, or "caveats", such as Eyes Only, Canadian-US, and NATO. These enclaves are logically and physically separated in a manner that isolates policies, applications, platforms, networks, infrastructures, and information stores. In addition to being expensive to develop, maintain, and deploy, these enclaves are silos that are often detrimental to information provision, i.e., the realization of shared situational awareness, collaboration, coordination, and decision making.

The recent shift from a "need-to-know" to a "requirement-to-share" information management and network cultures has introduced additional risks to information system environments. An increasing number of participants are being authorized to access security enclaves. Once a participant is authorized to enter an enclave, they are provided access to a wide range of information elements. Conventional access control solutions do not typically provide sufficient fidelity and flexibility in the application of policy/rules. They do not apply policies/rules to the actual content of individual information elements or provide defense-in-depth, i.e., layering safeguards based on the value of key data elements (e.g., security and privacy tags) within the instances of the information element. Recent security incidents illustrate the limitations of conventional access control solutions and their inability to exert sufficient control over and protect critical information assets. The following incidents illustrate current limitations:

As part of the response to the 9/11 attacks, the US determined that increased information sharing between departments and their infrastructure was necessary to prevent future terrorist activity. The perception of departments' information stores as hardened silos was seen as a barrier to effective security response. As a result, a new culture of openness was in effect at the Sensitive Compartmented Information Facility (SCIF). In this environment, Bradley Manning was able to use ***unrestricted and uncontrolled access to information*** *to disclose large amounts of sensitive data.*[2]

Edward Snowden's ***role as a systems administrator*** *provided easy access to classified National Security Agency documents sitting in a file-sharing location on the spy agency's intranet portal. As a contracted NSA systems administrator with top-secret Sensitive Compartmented Information (SCI) clearance, Snowden could access the intranet site and move especially sensitive documents to a more secure location without triggering security incident alarms.*[3]

SLt. Delisle has admitted to selling secret information to the Russians over a 4 ½-year period jeopardizing Canada's ability to protect itself, as well as its standing with key partner nations. Canadian officials ***concede they do not know precisely what SLt. Delisle gave the Russians*** *between 2007 and 2011. They're drawing inferences from material they intercepted just before arresting him.*[4]

While the two priorities—sharing and safeguarding—are often seen as mutually exclusive, in reality, they are mutually reinforcing. Information systems that strengthen protection and the fidelity of controls for sensitive information help build trust within the user and stakeholder communities. This trust will provide data owners and custodians with the confidence to:

- Increase operational effectiveness;

[2] http://www.telegraph.co.uk/news/worldnews/wikileaks/10210236/WikiLeaks-five-things-we-learned-from-the-Bradley-Manning-case.html
[3] http://www.computerworld.com/s/article/9242493/Snowden_s_role_provided_perfect_cover_for_NSA_data_theft
[4] http://www.theglobeandmail.com/news/national/convicted-spy-delisle-sold-csis-names-to-russians-court-told/article8030374/

- Improve information sharing capability;
- Increase information safeguarding capability;
- Increase the availability, deployment and repurposing of common/shared services and infrastructure;
- Reduce operational costs;
- Reduce management costs; and
- Reduce acquisition costs.

1.4 This Reference Architecture

The IEF is a framework for delivering defense-in-depth solutions that address a broad range of information sharing and safeguarding requirements. This reference architecture describes the integration patterns for, the functional requirements for and interactions between a set policy-driven data-centric services that include:

- Data Centric Policy Enforcement Points (PEP);
- Data Centric Policy Decision Points (PDP);
- Policy Administrations Points (PAP);
- Policy-based Packaging and Processing Services (PPS);
- Cryptographic Transformation Services;
- Security Services Gateway; and
- Logging Services.

This reference architecture combined with the specifications for these services enable the development of information management and security measures targeting the responsible sharing of information in accordance with policy and commensurate with the sensitivity of the data being accessed or shared. The IEF Reference Architecture (this document) defines an integration layer for off-the-shelf products and services that will enable the enforcement of ISS policy at the data level rather than the networks, platforms, systems, and applications.

1.5 IEF Delivered Capabilities

The Information Exchange Framework (IEF) seeks to define a set of open, publicly available international standards for policy-driven data-centric information sharing and safeguarding capabilities. The composite capability will deliver layered defense-in-depth safeguards that enable responsible information sharing across a broad-based set of missions and operational requirements. Where practical the IEF will align and integrate existing methodology, tools, technologies, and protocol standards and specifications. Most important, the IEF will define a service layer that facilitates the integration of existing, user applications, systems, platforms, and infrastructure.

The IEF separates the development and maintenance of policy/rules from the specific systems and services (i.e., policy decision and enforcement points) used to enforce them. This separation will enable users to:

- Evolve ISS polices/rules independent of services and infrastructure;
- Re-host ISS policies to multiple operating environments;
- Activate/deactivate ISS policies that respond to changes in operational requirements;
- Deploy a common or shared infrastructure, based on off-the-shelf products and services that users can rapidly tailor to planned, or spontaneous operational requirements;
- Adapt to rapid changes in operational context:
    - Changing mandates, roles and responsibilities;
    - Changing mission and operational context;

- Evolving threats and risks;
- Evolving institutional policy;
- Advancements in technology; and
- Control development and life-cycle costs.

Providing an architecture-driven approach to policy development provides the opportunity to integrate ISS policy models (e.g., IEPPV) into the broader segment, operational and enterprise architectures. This integration into standard architecture frameworks (e.g., DODAF) and supporting tools will:

- Align ISS policy model related architectural artifacts (operational topologies and deployments, platforms, systems, interfaces and data and information elements);
- Develop traceability to policy instruments (e.g., legislation, regulation, Service Level Agreements (SLA), Memorandum of Understanding (MoU) and Operating Procedures); and
- Provide information needed to effectively and efficiently validate, verify, and certify operational configurations and deployments.

The integration of ISS policy development into architecture will promote retention of institutional knowledge and an overall reduction in life-cycle costs.

1.6 IEF Objectives

The IEF will provide the ability and capacity for people, processes, and systems to work together efficiently to ensure that the right information is available to the right people or system at the right time. This will require solutions that incorporate the following characteristics.

Data-centric	Provide policy decision and enforcement points that operate on the data content for each instance of the information and data elements being assessed for release.
Data Centric Defense-in-Depth	Provide security services that directly apply security policy to data and information elements based on the sensitivity of content of the individual data and information elements and the authorizations of the publisher and each recipient.
Dynamic Interoperability	Provide services that provide the ability to adapt to changes in context within the operational environment over time. These services must have the ability to ingest and enforce configuration changes that adapt information interchanges to operational need.
Flexibility, Adaptability, and Agility	Provide services that enable the rapid re-use and repurposing of policy and data patterns for both planned and spontaneous operations; and enable run-time changes to active policy environments.
Information Decision Advantage	Provide services that enable users to rapidly adapt the operation of information services to provision decision makers with the highest quality of information available in a manner that provides them a strategic, operations, or tactical advantage.
Integration Overlay	Provide services that operate as an overlay to existing systems and infrastructure and do not require extensive change to the user systems and infrastructure.
Policy-driven	Provide practices, tools and services that provide a fully traceable path from

	information sharing and safeguarding policy to the rules executed by decision and enforcement points in the information systems and services that execute and enforce them.
Responsible Information Sharing	Provide services that enable users to align and balance the requirements to share and safeguard information and maximize the discoverability and access to information by authorized users.
Reuse of Existing Standards and Specifications	Integrate existing and evolving specifications and standards to define and implement IEF components. (e.g., DDS and XACML)
Self-Defending	Provide services that safeguard its own data and information elements (e.g., policies, instructions and logs).
Time-aware	Provide services that ensure policy decisions and enforcement occur at the time of sharing and enabling responsiveness to changes in operational context (e.g., threat, risk, roles & responsibilities, and access rights).
Vendor Agnostic	Provide open specifications that vendors can use to develop off-the-shelf products and services that combine to provide complete ISS solutions.

1.7 IEF RA Assumptions

The following assumptions were made when writing this reference architecture for the Information Exchange Framework:

1.7.1 IEF Component Specifications

Separate specifications exist or will be developed for each of the IEF Components. It is these component specifications that will define the detailed operations of each component. Examples of existing specifications that address core IEF requirements include:

- PDP & Access Control Policy Language: XACML v3 Specification; and
- ISMB: DDS and XMPP specifications were used in separate implementations of the IEF to demonstrate and prove the concepts contained in this reference architecture.

Examples of components that will require the development of public specifications include:

- PPS: Requires a specification for services that ingest and enforce data policies that are developed using IEPPV.
- PAP: Requires a specification that defines how component configurations and policies can be managed and administered by the user.

1.7.2 Error & Complex Conditions

Users will define the IEF response to Error Conditions (e.g., user not authorized to access specified InformationElement, requisite Metadata is not available, or Cryptographic Keys not available). The response to these conditions has policy implications and must be addressed during implementation or integration. The reference architecture provides component descriptions and sequence diagrams that describe the core operation of the components. Information not specified:

- Component response to error conditions;
- Complex policy decisions including special handling or release instructions; and
- Complex or compound file requests, e.g.:
    - Process multiple files in a single instruction (e.g. “COPY *.* Destination” or “DEL *.doc); and

- Processing structured messages containing multiple embedded information elements (e.g., Digest, multiple information packages, and multiple information Payloads).

1.7.3 File Metadata

The reference architecture does not specify the supported user file formats, and whether or not these formats include the requisite metadata elements. This level of detail should be covered in the PEP specification. Although metadata (sensitivity marking) is critical to the approach, how a user's policies addresses the need is beyond the scope of this reference architecture. The architecture defines a Secure Asset Container (SAC) as a core feature for securely handling information assets.

1.8 IEF RA Design Principles

The Key principles of the IEF and its support Environment.

Policy Development / Support Environment:

Define strategies, practices and tools that:

- Enable rapid development, maintenance and deployment of ISS Policy;
- Support MDA approaches;
- Align and integrate ISS policy development with Enterprise Architecture (EA) best practices;
- Separate the definition of policies (/rules/instructions) from the services employed to enforce them; and
- Provide users with the architectural artifacts that support and enable governance, auditing (real-time monitoring & forensic), Modeling and simulation, retention of institutional knowledge and Life-cycle management.

Information Sharing and Safeguarding Policy:

Define a policy vocabulary that:

- Supports multiple policy language implementations (e.g., XACML, SAML, and RuleML);
- Supports modeling language profiles (e.g., UML);
- Supports multiple domain vocabularies; and
- Provides Data Centric Approaches to Information protection.

IEF Services:

Define Services that:

- Automate the enforcement of user defined information sharing and safeguarding at the data-level;
- Implement defense-in-depth to the data level; and
- Deliver operational flexibility, adaptability and agility.

General:

- Reuse existing standards where and whenever possible; and
- Vendor neutral specifications.

1.9 Adapting to change

Each IEF component is governed by policy and/or configuration files that can be modified or augmented by the Policy Administration Point (PAP) using the PAP-Command Message (See Annex A). The PAP provides an interface that enables an authorized user to manually change, or schedule changes to the configurations of the IEF components, and the policies being executed by the Policy Decision Points(s) and the Packaging and Processing Services operating in its domain.

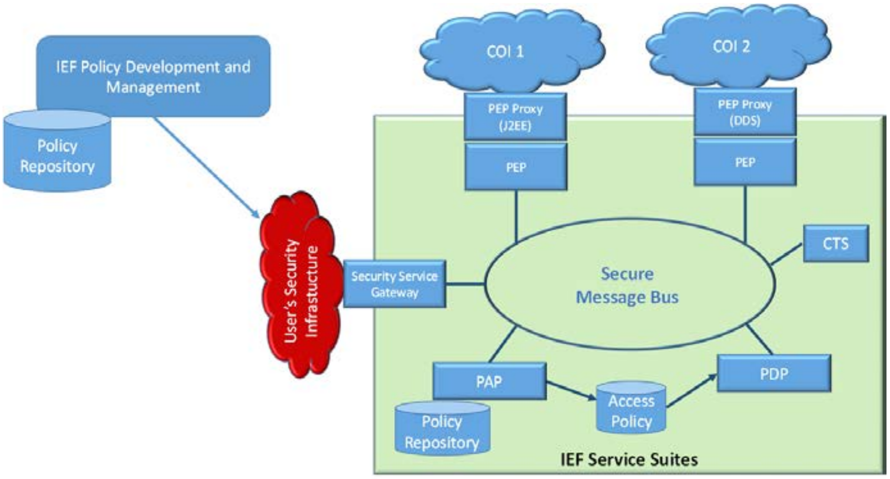


Figure 1 – Adapting to Change

In addition, PAP can be implemented with the ability to request policy sets and/or component configurations from the users' policy development or Policy Management environments. The PAP can use this capability to request policies or configurations that enable the IEF to adapt to changes in the operational environment. This capability can be augmented by fitting the PAP with a policy repository that stores policies and configurations to known changes in the operational environment (e.g., transitioning from mitigation, preparedness, response, and recovery phases of an emergency operation).

The IEF can be deployed as a standard software suite or virtual machine and tailored to the users' operational need using the flexibility offered in the robust policy and configuration options. In the cloud, IEF capability can be rapidly deployed and configured to mission and user needs.

1.10 Security Considerations

This clause draws together key security considerations described in more detail elsewhere in this RA. this document. It is included to assist the reader in understanding those security capabilities addressed by the IEF-RA and those that must be addressed by the user and/or their integrator.

The IEF is a framework (integration pattern) for delivering defense-in-depth for information sharing and safeguarding solutions. This reference architecture describes patterns of integration for securing email, file-share, instant-messaging (chat) and structured Messaging (e.g., XML and

JSON). These patterns are based on the integration of seven security services, including: PEP, PDP, PAP, PPS, CTS, SSG, and Logging.

This RA defines and extra level of defense for data, the architecture does not replace the need for the networks, platforms, systems, applications and other traditional security practices and methods. The RA relies on the users' implementation of identity, credential and access management (ICAM), cryptography, and other security services; weaknesses or deficiencies in these services will affect the effectiveness and performance of IEF capability.

The IEF is seeking to assure that the content of the information shared is appropriate for the recipient(s) authorizations, meaning:

- Information authorized to a recipient is not blocked by access control services that cannot differentiate between data and information elements with different levels of sensitivity or classification in the environments they are protecting. Many security solutions assume a single level of protection of all elements in their domain.
- Information is authorized for release because the access control services cannot identify sensitive or classified information not authorized for release is in the payload. Many access control services action metadata (tags and labels) in the header of the message or bound to the payload. Access control do not have the semantic capabilities to assess the content. The IEF offers the opportunity to integrate semantic services (e.g., PPS) into the ISS processes.
- Information aggregated by automated processes (e.g., application interfaces) can enforce policy that directs the marking (e.g.: tagging and labeling of individual and composite elements are marked appropriately in real-time, prior to release.

The IEF aligns well with Zero-Trust security approaches that assume that ever request to access to release information is assess vis-à-vis the recipients' authorizations and assurance the employed infrastructure is authorized to carry that information. This capability requires the user (data owner / steward) to:

- Assure that for human driven ISS services like email, file-share and text-messaging (chat) the ability to effectively tag and label information elements in accordance with policy. This capability is built into the IEF-PPS for structured messaging. This RA does not specifically address user services and thein integration of tagging and label services and interfaces.
- Define policies for the access, use, and sharing of information at a high level of fidelity; the level of fidelity in the policy implementation will dictate the level of capability provided.
- Assure that recipient attributes (authorizations) identify the types of information they can access in a specific role, location, or other user specified information assurance (IA) discriminators. This requires the user to integrate the IEF components within existing IA and security infrastructure through the IEF SSG.

- Tailor the PEP operation to exploit existing capabilities and information sharing services. The RA provides a representative sequence diagrams for this integration. Based on the capabilities of the user's own infrastructure and level of integration – greater of lesser capability can be developed.

The RA does not specify the requirements for security series beyond the seven (7) services comprising the IEF. It is up to the users' own tolerance for risk that will dictate boundary, networks, platforms, and applications, ICAM and cryptography that will dictate these requirements. These capabilities must be specified and implemented by the user.

2 Compliance

2.1 Introduction

The Information Exchange Framework Reference Architecture defines four (4) separate information sharing and safeguarding service patterns (File Share, Email, Instant Messaging and Structured Messaging). Each pattern forms a separate compliance point. An implementer may select to implement one or more of the four compliance points.

2.2 Selecting a Compliance Point

There are separate compliance points defined for each of the four information sharing types addressed by this reference architecture, i.e.:

1. Structured Messaging;
2. File Sharing;
3. Email; and
4. Instant Messaging / Chat rooms.

A user, vendor or integrator may select one of the compliance points described below to address their information sharing and safeguarding needs.

2.3 Compliance Points

To express compliance to this reference architecture, an implementer, vendor, or integrator must implement all six mandatory components and the Policy Enforcement Point that addresses their information sharing and safeguarding needs, i.e.:

1. Structured Messaging offers (Clause 2.6.1) provide two compliant configurations:
    a. Proxy Component Operations (See Clause 2.6.1.2), and
    b. Integrated Data Service (see Clause2.6.1.1);
2. File Sharing (Clause 2.6.2);
3. Email (Clause 2.6.3); and
4. Instant Messaging (Clause 2.3.2.3).

The six mandatory components for each configuration are:

1. Policy Administration Point (Clause 8);
2. Policy Decision Point (Clause 9);
3. IEF Security Services Gateway (Clause 12);
4. Cryptographic Transformation Service (Clause 13);
5. Trusted Logging Services (Clause 15); and
6. Secure Messaging Bus (Clause 14).

Figure 2 outlines a generalized architecture including all four information sharing types, including the two forms of Structured messaging integration.

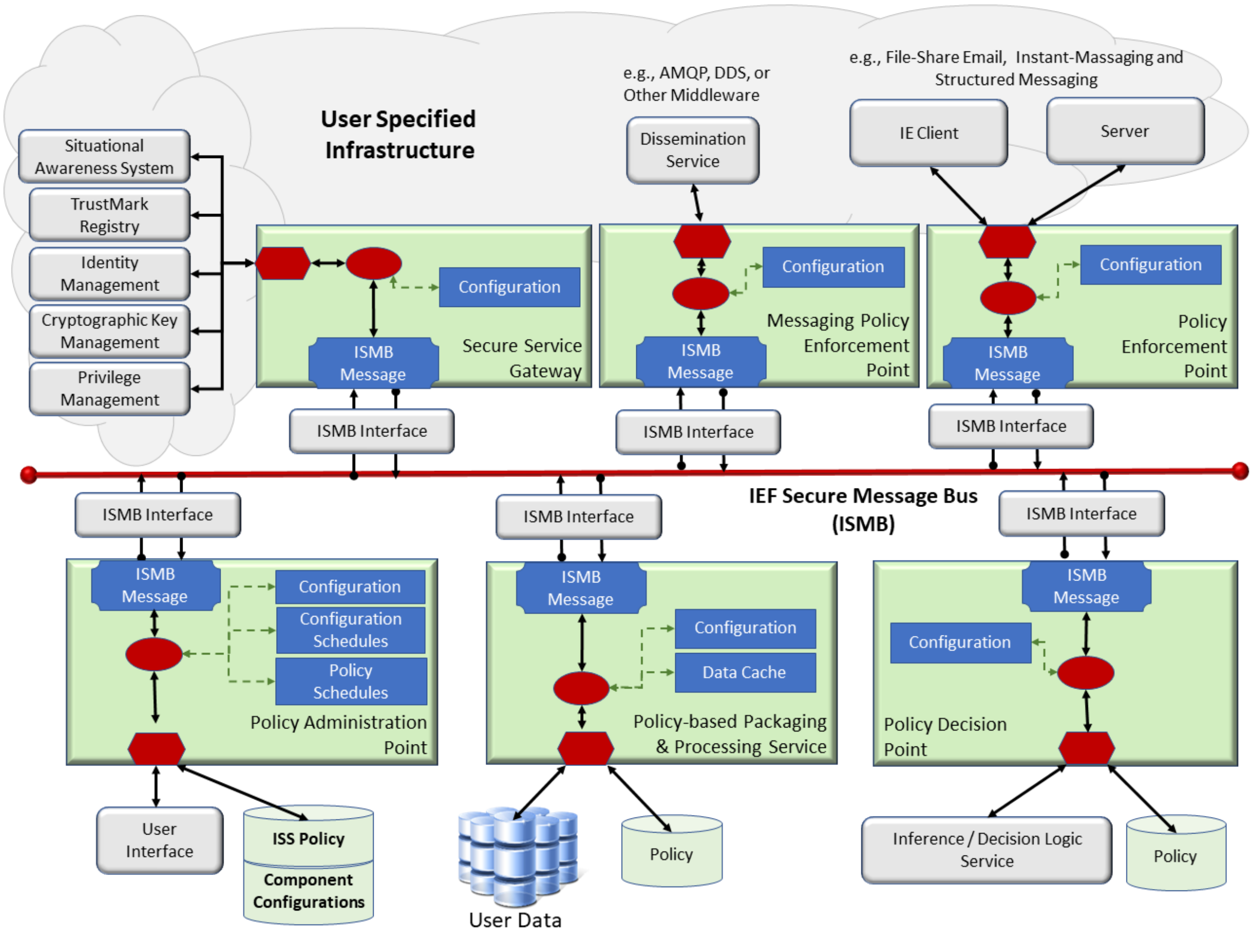

User Specified Infrastructure
e.g., AMQP, DDS, or Other Middleware
e.g., File-Share Email, Instant-Massaging and Structured Messaging
Situational Awareness System
TrustMark Registry
Identity Management
Cryptographic Key Management
Privilege Management
Dissemination Service
IE Client
Server
Configuration
ISMB Message
Secure Service Gateway
Messaging Policy Enforcement Point
Policy Enforcement Point
ISMB Interface
IEF Secure Message Bus (ISMB)
Configuration Schedules
Policy Schedules
Policy Administration Point
Data Cache
Policy-based Packaging & Processing Service
Policy Decision Point
User Interface
ISS Policy
Component Configurations
User Data
Policy
Inference / Decision Logic Service

Figure 2 – General Compliance

2.4 Client Server Proxy

Figure 3 illustrates an integration pattern whereby the IEF is used as a set of proxy information sharing and safeguarding services and the client application and a corresponding server (e.g., email, file, tect messaging and data) are provided on the users' own network and security domain. Within the context of this reference architecture, its form of compliance applies to:

1. File Share (e.g., user application or file browser and file server);
2. Email Client (Mail Client or User Agent and Mail Transfer or delivery Agents (servers));
3. Instant Messaging (IM Client and Message Oriented Middleware); and
4. Structured Messaging (User Application, PPS/Database-Server).

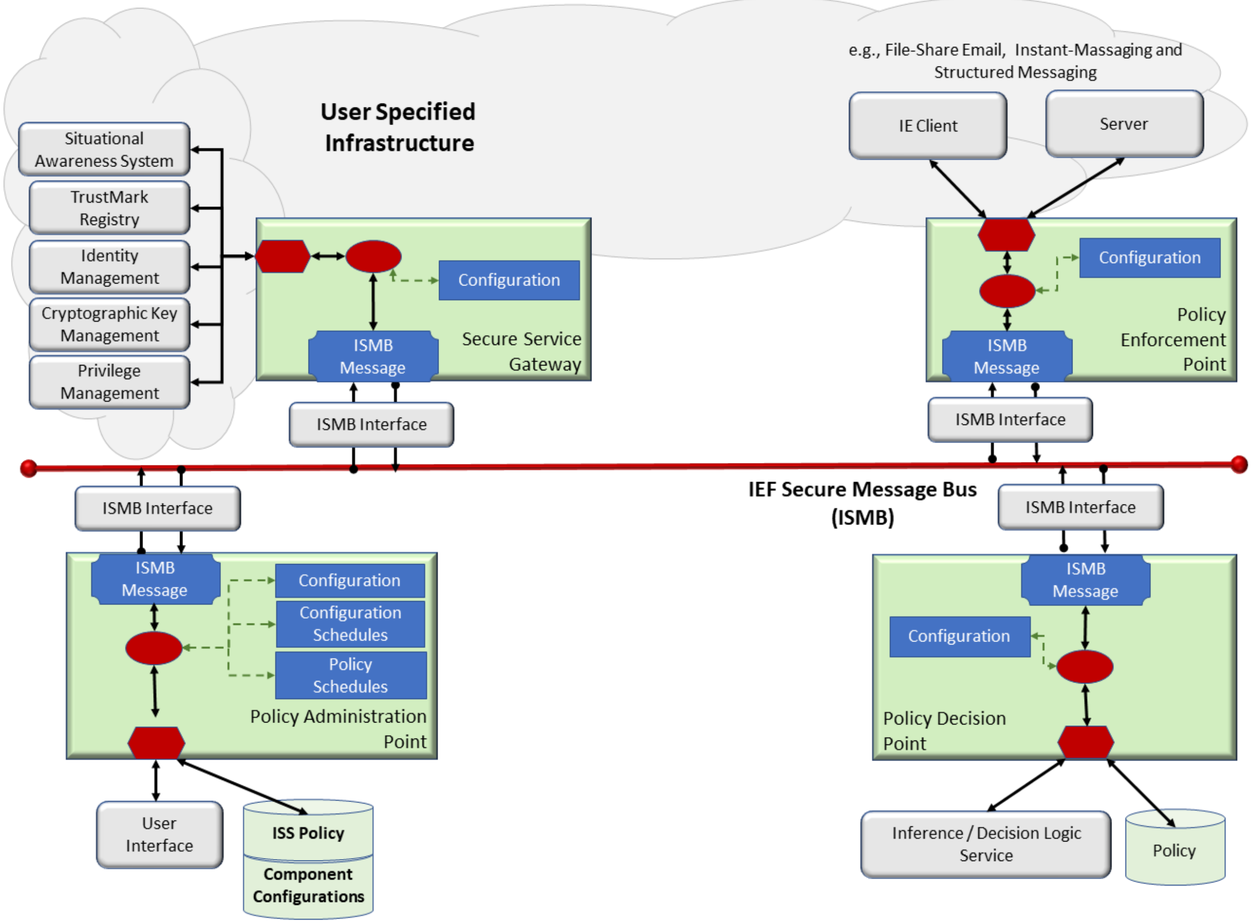


Figure 3 - Client with Server Compliance

For each of these integration patterns, the PEP introduces a proxy (proxy-server) that intercepts each transfer between the server and client application. Each PEP is tailored for the specific service and protocol. Each transfer is decomposed into its individual parts (Information Element(s)) and the metadata (e.g., security markings, caveats, and privacy markings) are extracted and the PEP verifies that each recipient is authorized to access the information content being transferred. As the PEP uses metadata (marks, tags, or labels) to validate and verify that users are authorized to send or receive included information elements, each transferred information element must be appropriately marked by the user (or system) prior to the transfer. This often means that the User Client application must include or be integrated with data/information classification software that directs the user to mark information elements and binds those markings to the elements.

2.5 IEF Embedded Operation

The following figure illustrates an integration pattern that can be used for the PPS/Messaging-PEP to isolate user data from the external network environment, limiting user access to PEP communications.

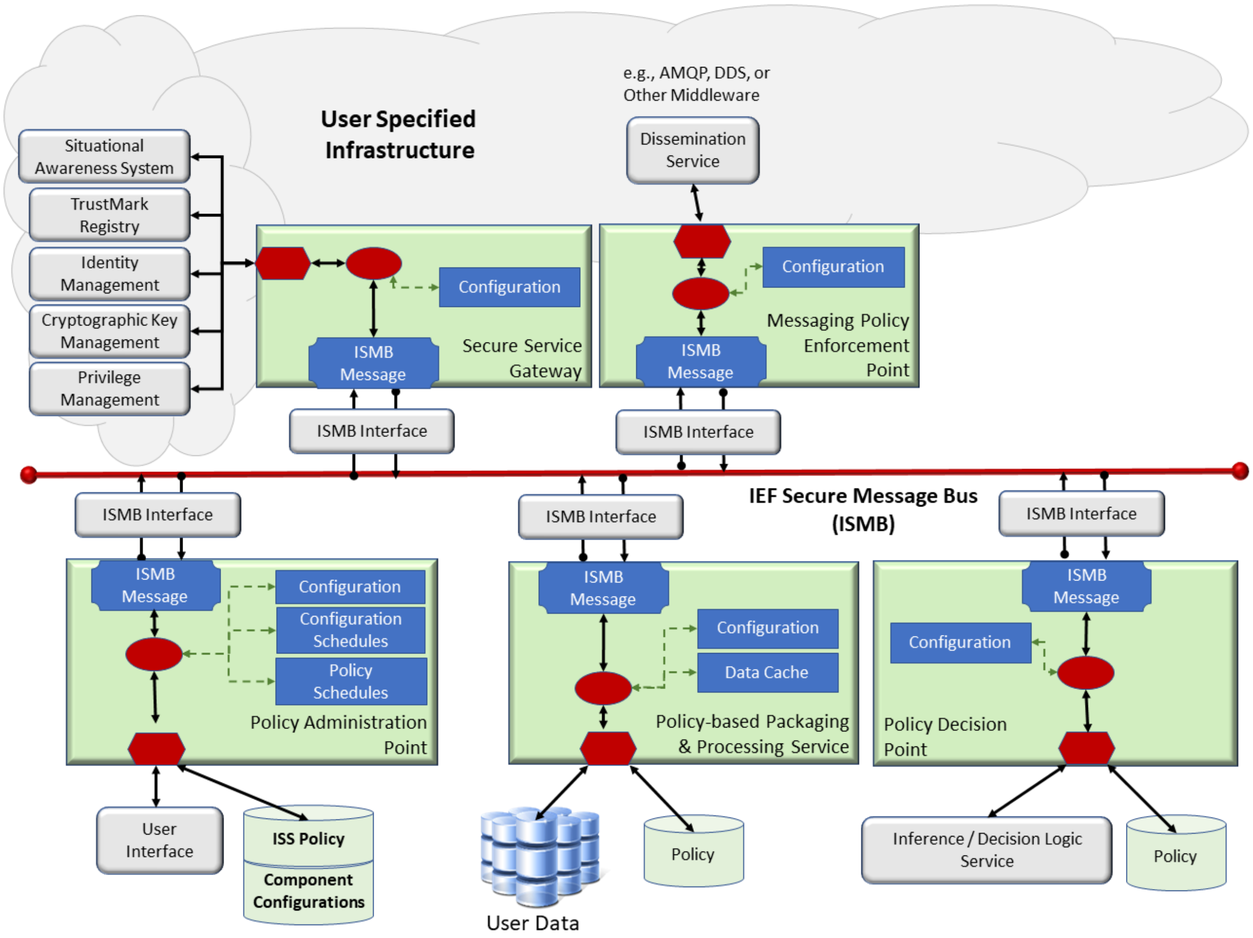


Figure 4 – Integrated Data Service

2.6 Compliance Options

The following clauses describe IEF Compliance Points. In general terms, a compliant implementation would address all manadatory requirements for specified IEF components for one or more of the structured messaging, email, text messating (chat) and fileshare configurations. An implemenetor may implement one of more of the compliance points:

1. Structured Messaging:
    a. PPS Proxy configuration, and/or
    b. PPS embedded configuration;
2. File Sharing;
3. Email; and/or
4. Intastant Messaging.

2.6.1 Structured Messaging Conformance

Structured messaging compliance provides selective sharing of structured data, packaged as messages, conforming to canonical information models (e.g., National Information Exchange Model (NIEM), Emergency Data Exchange Language (EDXL) or HL7), and each recipients' authorizations to access and use specific data content. There are two configuration options provided for the Packaging and Processing Service (PPS; Clause 11) and Messaging PEP (Clause 10.5):

1. PPS Proxy Component Operations (See Clause10.5.2); and

PPS Integrated Component Operations (see Clause 10.5.1).

2.6.1.1 PPS proxy configuration

The Policy-based Packaging and Processing Services (PPS) provide a policy-based approach to the secure preparation and processing of structured messages (e.g., NIEM, CAP, EDXL, and HL7) at machine speeds. The PPS may be implemented in the user's own network as a proxy server in the same manner as a file sharing, email, or IM server. In this configuration the PPS and the Messaging-PEP are connected as illustrated below. This form of integration provides two options:

1. User Client Application Integration; and
2. Middleware Application Integration.

The following figure illustreate proxy server implementation that conforms to the IEF-RA. In this configuration, the data protection elements are split between the IEF access and release control services in the PEP and the data protection services of the PPS. There is a greater reliance on network security capabilities to assure all routing is through the IEF proxt service.

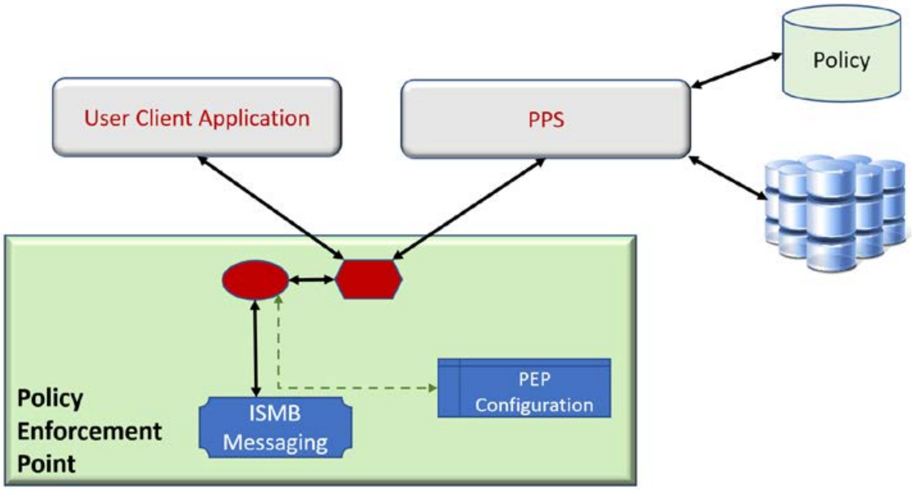


Figure 5 – User Client (Messaging) PEP

The following figure illustrates this proxy based apprach with the users' middleware infrastructure.

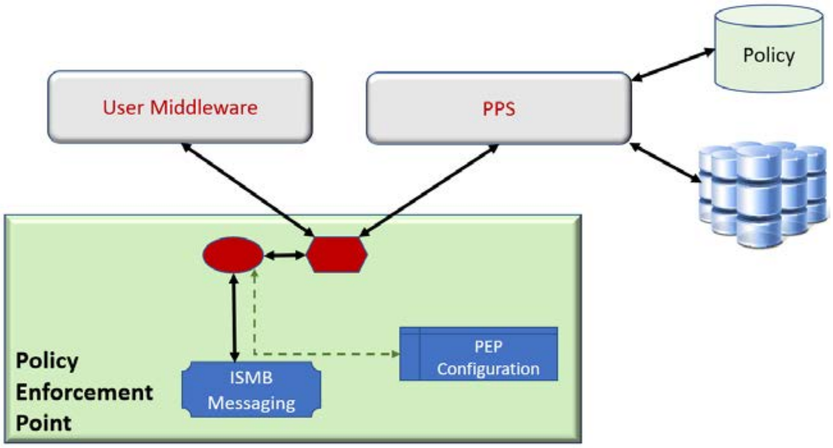


Figure 6 - Messaging PEP

The PEP is an integration point between IEF services and the Users' own data environment. The PEP (/Proxy) must be able to:

1. Intercept each information exchange between the users messaging middleware (e.g., AMQP and DDS) and the PPS;
2. Extract the metadata markings (/labels / tags) bound to each file by the user;
3. Gather recipient authorizations to specifically marked information;
4. Stage the authorization process for each user requested file operation (e.g., create, open (e.g. view, execute, edit or print), rename, move, copy, delete, list or search);
5. Access IEF services (e.g., policy adjudication and determination, cryptographic transformation);
6. Enforce policy determinations; and
7. Log the transaction.

2.6.1.2 PPS Integrated Configuration

The integration of the user client to the PPS requires a PEP tailored to intercept communications between a user application and a PPS that operates in a server configuration. The Messaging-PEP validates and verifies that the user is authorized to request information from the specified PPS.

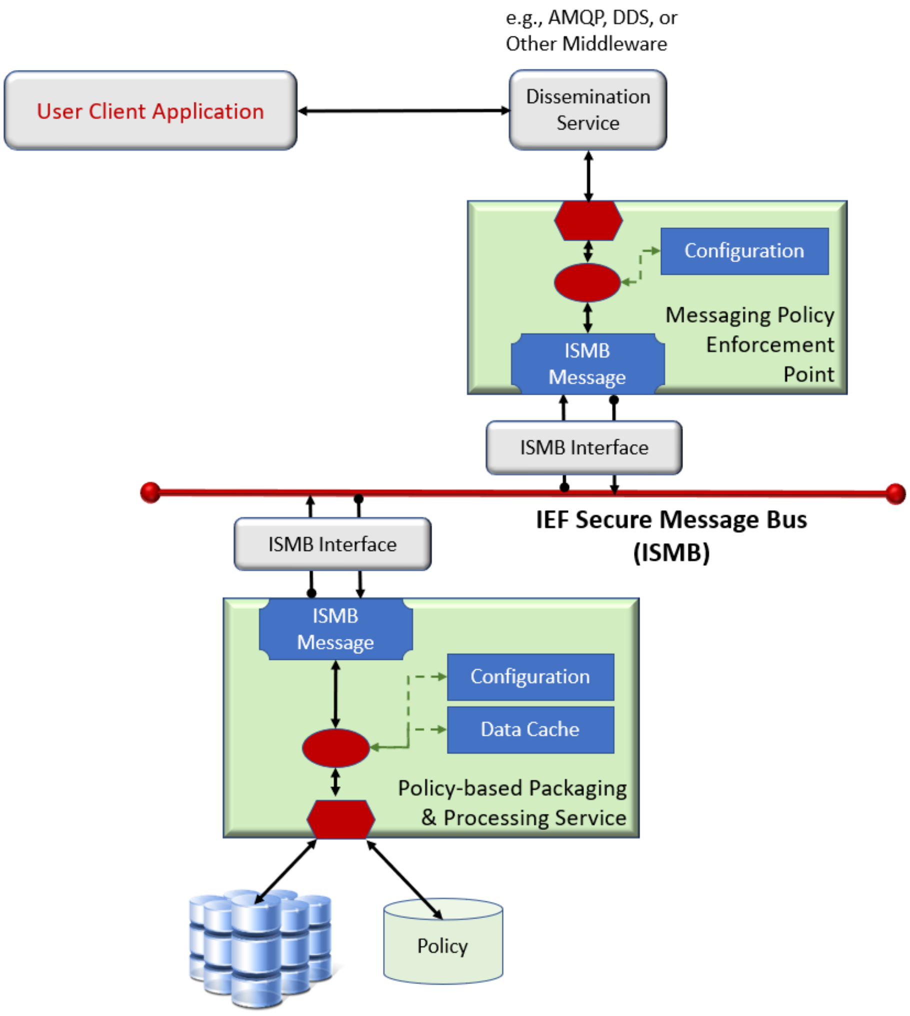


Figure 7 - Messaging PEP

As a service integrated into the IEF Secure Messaging Bus (ISMB) user data in isolated behind the policy enforcement points and the PPS, adding additional protection and a reduction in security threat vectors. Both integrated or proxy configurations conform to this reference architecture.

The PEP is an integration point between IEF services and the Users' own Data environment. The PEP (/Proxy) must be able to:

1. Intercept each information exchange between the users messaging middleware (e.g., REST, AMQP and DDS) and the PPS;
2. Extract the metadata markings (/labels / tags) bound to each file by the user;
3. Gather recipient authorizations to specifically marked information;
4. Stage the authorization process for each user requested file operation (e.g., create, open (e.g. view, execute, edit or print), rename, move, copy, delete, list or search);
5. Access IEF services (e.g., policy adjudication and determination, cryptographic transformation);

6. Enforce policy determinations; and
7. Log the transaction.

2.6.2 File Sharing Conformance

File information sharing and safeguarding compliance requires a PEP that is tailored to intercept user operations or requests between the user application (e.g., office application, file manager or file browser) and a file server or local device persisting the users' folders and files. The File-PEP validates and verifies that the user is authorized to request the operation such as create, open (e.g. view, execute, edit or print), rename, move, copy, delete, list or search for a file or files with specific content or location (e.g., device, drive, and folder).

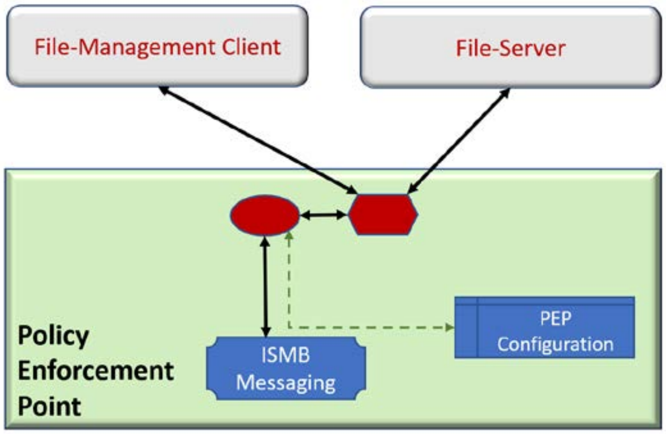


Figure 8 - File Sharing PEP

The PEP is an integration point between IEF services and the Users' own file environment. The PEP (/Proxy) must be able to:

1. Intercept each operation directed from a client application to the protected file share;
2. Extract the metadata markings (/labels / tags) bound to each file by the user;
3. Gather recipient authorizations (/privileges) to specifically marked information;
4. Stage the authorization process for each user requested file operation (e.g., create, open (e.g. view, execute, edit or print), rename, move, copy, delete, list or search);
5. Access IEF services (e.g., policy adjudication and determination, cryptographic transformation);
6. Enforce policy determinations; and
7. Log the transactions.

2.6.3 Email Conformance

Email information sharing and safeguarding compliance requires a PEP that is tailored to intercept emails transiting between an Email Client or User Mail Agent and the Email Server (Transfer or Delivery Agent). This Email-PEP validates and verifies that the recipient of the email is authorized to access the information elements (email body and attachments) contained in each email.

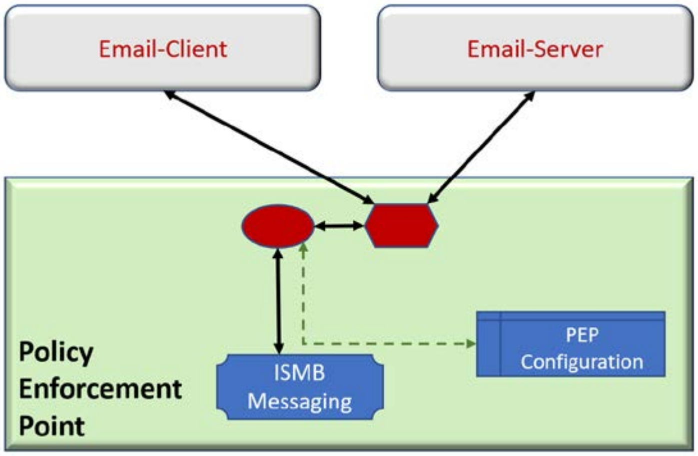


Figure 9 - Email PEP

The PEP is an integration point between IEF services and the Users' own email environment. The PEP (/Proxy) must be able to:

1. Intercept each Email transitioning between Email Client or User Mail Agent and the Email Server;
2. Extract the information elements contained within the email message;
3. Extract the metadata markings (/labels / tags) bound to each information element;
4. Gather recipient authorizations (/privileges) to specifically marked information;
5. Stage the authorization process for each information element;
6. Access IEF services (e.g., policy adjudication and determination, cryptographic transformation);
7. Repackage the email with only the authorized information elements; and
8. Log the transaction.

2.6.4 Instant Messaging Conformance

Instant Messaging information sharing and safeguarding compliance requires a PEP that is tailored to intercept each message between the IM Client and Message Oriented Middleware. The IM-PEP validates and verifies that the user is authorized to request an operation (e.g., create chat room, join chat room, list chat rooms, receive message, send message).

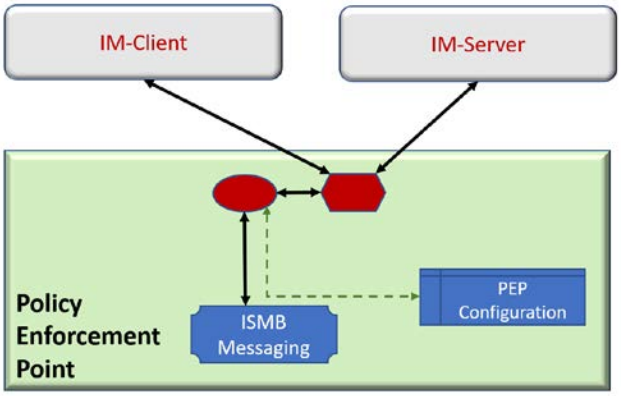


Figure 10 - IM PEP

The PEP is an integration point between IEF services and the Users' own IM environment. The PEP (/Proxy) must be able to:

1. Intercept each operation directed to the protected Message Oriented Middleware;
2. Extract the metadata markings (/labels / tags) bound to each file by the user;
3. Gather recipient authorizations to specifically marked information;
4. Stage the authorization process for each user requested IM operation;
5. Access IEF services (e.g., policy adjudication and determination, cryptographic transformation);
6. Enforce policy determinations; and
7. Log the transaction.

3 References

3.1 Normative References

The following normative documents contain provisions, which through reference in this text, constitute provisions of this reference architecture. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply.

1. eXtensible Access Control Markup Language (XACML); from OASIS; http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.pdf
2. Data Distribution Service™ (DDS™), http://www.omg.org/spec/DDS/
3. DDS Security™ Specification (DDS-SECURITY™), http://www.omg.org/spec/DDS-SECURITY/
4. Unified Profile for the Department of Defense Architecture Framework (DoDAF) and the Ministry of Defence Architecture Framework (MODAF), http://www.omg.org/spec/UPDM/
5. Information Exchange Packaging Policy Vocabulary™ (IEPPV™), http://www.omg.org/spec/IEPPV/
6. Shared Operational Picture Exchange Services (SOPES™) Information Exchange Data Model(IEDM), http://www.omg.org/spec/SOPES/
7. Unified Modeling Language™ (UML®), http://www.omg.org/spec/UML/
8. Extensible Markup Language (XML) 1.0 (Fifth Edition), https://www.w3.org/TR/xml/
9. W3C XML Schema Definition Language (XSD) 1.1 Part 1: Structures, https://www.w3.org/TR/2012/REC-xmlschema11-1-20120405/
10. W3C XML Schema Definition Language (XSD) 1.1 Part 2: Datatypes, https://www.w3.org/TR/2012/REC-xmlschema11-2-20120405/
11. XML Schema Part 1: Structures Second Edition, https://www.w3.org/TR/2004/REC-xmlschema-1-20041028/
12. XML Schema Part 2: Datatypes Second Edition, https://www.w3.org/TR/2004/REC-xmlschema-2-20041028/
13. IEF Reference Architecture RFP , http://www.omg.org/cgi-bin/doc.cgi?mars/2014-3-17

3.2 Reference Materials (Informational)

The following material was used as background for the development of this reference architecture.

1. Information Sharing and Safeguarding, Policy-Driven Data-Centric Solution for Structured Messaging, Abramson & Penwill, 2014, http://asmg-ltd.com/ief/2014-09-10%20ISS%20-%20Policy%20Models%20for%20Structured%20Messaging%20v1-1.pdf
2. Information Sharing and Safeguarding , Policy Modeling for Structured Messaging, Abramson & Penwill, 2014, http://asmg-ltd.com/ief/2014-09-10%20ISS%20-%20PdDc%20Solution%20of%20Structured%20Messaging%20v1-1.pdf
3. Information Assurance Architecture, K. Willet, CRC Press, 2008, ISBN 9780849380679 - CAT# AU8067
4. The Security Development Lifecycle, Howard & Lipner, Microsoft Press, 2006, ISBN/ASIN: 0735622140 ISBN-13: 9780735622142
5. Building a Global Information Assurance Program, Curtis & Campbell, Auerbach Publications, 2003, eBook ISBN 9780203997550

6. Adaptive Information, Pollock & Hodson, Wiley-InterScience, 2004. **ISBN:** 9780471714200
7. Information Security Management, Raggad, CRC Press, 2010, ISBN 10: 1420078542 ISBN 13: 9781420078541
8. Core Security Patterns, Steel et al, Prentice Hall, 2006, ISBN13: 9780131463073 ISBN: 0131463071
9. Information Security Architecture, Killmeyer, Auerbach Publications, 2006, ISBN 10: 0849315492 ISBN 13: 9780849315497
10. Cloud Security and Privacy, Mather et al, O'Rilley Books, 2009, **ISBN:** 9781449391881
11. Asset Protection and Security Management Handbook, POA Publishing, 2003,
12. Information Security Management Handbook, Tipton et al, Auerbach Publications, 2007, ISBN: 0849316030 9780849316036
13. SOA Security, Kanneganti et al, Manning Publications, 2008, ISBN-10: 1932394680 ISBN-13: 978-1932394689
14. http://www.telegraph.co.uk/news/worldnews/wikileaks/10210236/WikiLeaks-five-things-we-learned-from-the-Bradley-Manning-case.html
15. http://www.computerworld.com/article/2485175/security0/snowden-s-role-provided--perfect-cover--for-nsa-data-theft.html
16. http://www.theglobeandmail.com/news/national/convicted-spy-delisle-sold-csis-names-to-russians-court-told/article8030374/
17. Practical Challenges Facing Communities of Interest in the Net-Centric Department of Defense; Connors, Dr. Malloy: http://www.mitre.org/sites/default/files/pdf/06_1254.pdf
18. Understanding the Security, Privacy and Trust Challenges of Cloud Computing; Debabrata Nayak: http://riverpublishers.com/journal/journal_articles/RP_Journal_2245-1439_127.pdf
19. MILS:Architecture for High-Assurance Embedded Computing; Luke, Taylor, Uchenick: http://www.crosstalkonline.org/storage/issue-archives/2005/200508/200508-Vanfleet.pdf
20. XACML, ABAC, Privacy preserving access-controls; Bodriagov: http://www.csc.kth.se/~buc/PPC/Slides/accesscontrololeksandr.pdf
21. What Can You Do with XMPP?, Barrett, 2009:http://fyi.oreilly.com/2009/05/what-can-you-do-with-xmpp.html
22. SAMSON Technology Demonstrator Architectural Design Document Phase IV SD002, Dr. Daniel Charlebois et al, Defence Research and Development Canada, 2013, http://publications.gc.ca/collections/collection_2016/rddc-drdc/D68-3-C3-2013-eng.pdf
23. SAMSON Technology Demonstrator Detailed Design Document Phase IV SD004, Dr. Daniel Charlebois et al, Defence Research and Development Canada, http://publications.gc.ca/collections/collection_2016/rddc-drdc/D68-3-C6-2013-eng.pdf
24. Secure Access Management for Secure Operational Networks (SAMSON), Concept of Operations (CONOPS), http://cradpdf.drdc-rddc.gc.ca/PDFS/unc251/p804697_A1b.pdf
25. "MILS Architecture", 2014, EURO-MILS white paper http://euromils.eu/downloads/2014-EURO-MILS-MILS-Architecture-white-paper.pdf

3.3 Additional Specifications and standards

These specifications are not referenced in this reference architecture but may form part of an IEF implementation:

1. Logical Entity eXchange Specifications 4.0 (LEXS), https://lexs.codeplex.com/
2. SAML V2.0, http://saml.xml.org/saml-specifications
3. Extensible Messaging and Presence Protocol (XMPP), Source: http://www.ietf.org/rfc/rfc3920.txt, Source https://xmpp.org/.
4. Instant Messaging and Presence Protocol (IMPP), https://www.ietf.org/rfc/rfc2779.txt
5. National Information Exchange Model (NIEM), http://www.NIEM.gov

4 Terms and Definitions

The focus of this reference architecture is the development of a formal vocabulary (terms and definitions) for the specification and design of information/data packaging policy (/business rules and Constraints). The definitions for the Information Exchange Framework Vocabulary elements are included in Annex F.

To assist the reader who may not be familiar with the information sharing and safeguarding domain, Annex F provides a glossary of these terms and acronyms. These definitions are provided for information purposes only.

5 Symbols/Acronyms

5.1 Symbols

There are no additional symbols defined for this reference architecture. All symbols used in this reference architecture are based on standard UML.

6 Additional Information

6.1 Intended Audience

This reference Architecture will be of interest to end users, analysts and integrators who define information exchange specifications. It will also be of interested to tool vendors interested in developing products and services that enable information sharing and safeguarding solutions. It can be an information source for end users, auditors and developers who seek to understanding the the services that enforce and govern secure and trusted information sharing.Acknowledgements

The following organizations are the direct submitters to this reference architecture:

- Advanced Systems Management Group (ASMG) Ltd.
- Defence Research and Development Canada (DRDC), Centre for Security Sciences (CSS)

ASMG personnel acted as the principle authors and editors of the specification and models.

DRDC CSS contributed the specification and design documents for the Secure Access Management for Secure Operational Networks (SAMSON) Technology Demonstration Project (TDP) and Trusted Information Exchange Services (TIES) TDP as foundational documents for this reference architecture.

6.2 Contributors (/Contributing Entities)

In particular the submitter would like to acknowledge the participation and contribution from the following individuals: Michael Abramson (ASMG), Jean Claude Lecomte (ASMG), , Oliver Crosby (ASMG), Anne Fry (ASMG), Eric Penwill, Vijay Mehra (KYM Advisors), and Dr. Daniel Charlesbois (DRDC/CSS).

The following organizations identified support for the concepts and content included in this reference architecture.

1. Advanced Systems Management Group (ASMG) Ltd.
2. Defence Research and Development Canada (DRDC) / Centre for Security Sciences (CSS);
3. MIAB Systems Ltd;
4. Lecomte Systems; and
5. KYM Advisors

6.3 Additional Materials

N/A

6.4 IEF RA Objective

The IEF Reference architecture is intended to provide readers with an architectural overview of an integration of software services that provides Policy-driven Data-centric Information Sharing and Safeguarding (ISS). The architecture provides specialized definitions for common security services that conform to the protection needs of information elements (i.e., File, Email, Text/Instant Message, and Structured Message). These services include:

- Policy Enforcement Point (PEP);
- Policy Decision Point (PDP); and
- Policy Administration Point (PAP).

The IEF draws many of the access control concepts from the XACML and SAML specifications. Where necessary these concepts were were specialized to to the requirements of "InformationElements" and how and where they are stored, used and shared within and between individuals and groups operating at different levels of trust, yet having to share sensitive information to meet business needs and outcomes.

The IEF also presents an architectural approach that seeks to integrate with existing security services and infrastructure without a large scale need to rip and replace existing components. Existing services would be integrated through the IEF Security Services Gateway (ISSG). The ISSG provides a single point of integration to client services, including:

- Identity, Credential and Access Management Services (ICAM);
- Cryptographic Key Management Services (key generation and escrow);
- Directory Services (e.g. Active Directory or LDAP); and
- Situational Awareness or Incident Management Services (providing operational context).

The user’s information processing applications and information exchange (file-share, email, text-messaging and middleware) services are integrated through a proxy-server that intercepts each transaction and provides it to the specialized PEP for the enforcement user specified access and release control policies. The interaction (messaging) between IEF components is isolated (physically or logically) from the rest of the users networking environment through the implementation of the IEF Secure Messaging Bus (ISMB). Platform specific Implementations (PSM) for the ISMB have been implemented using DDS and XMPP infrastructures.

6.5 Modeling Conventions

The IEF is modeled using standard UML and UML Profiles:

- Class Diagrams;
- Sequence Diagrams; and
- IEPPV Diagrams.

6.6 OMG Related Work

The following clauses identify other OMG Initiatives that relate to the IEF RA.

IEPPV
The Information Exchange Packaging Policy Vocabulary (formal/2015-05-06) defines the policy vocabulary and UML profile for developing PPS Policies.

UAF
The Unified Architecture Framework (dtc/2016-08-01) enables the user to specify the IEF deployment in conjunction with the rest of their enterprise Architecture. The IEPPV is integrated into the UAF, which further aligns these efforts.

UPDM
See UAF.

DDS
The Data Distributions Service can be integrated as the ISMB or a Data distribution capability in the User’s environment.

DDS-Security
DDS Security can be used to further fortify the security capability of the IESM and User data distribution capability.

Threat & Risk
UML Operational Threat & Risk Model provides an information exchange format based on NIEM and an explicit mapping to STIX . Other exchange formats, such as CAP may be supported as well. The IEF can be used to enable responsible information sharing using these messaging semantics.

Tagging and Labeling

Initiative for C4I	C4I, MARS, Data Residency, Systems Assurance and other groups are seeking a Tagging and labeling specification. If this is realized – the metadata definitions in Clause 17 and Annex C, and the enumerations in Annex D can be replaced with the resulting specification. The information in this Clause and Annexes is included to assist users in advance of the proposed specification.

7 IEF Reference Architecture Overview

7.1 IEF Services

This reference architecture defines the Information sharing and safeguarding architecture for the Information Exchange Framework. The reference architecture defines a set of ISS services and service interfaces:

- That enable the implementation of out of the box services that will be policy-driven data-centric services that can be integrated to provide secure information sharing using off-the-shelf:
  - Email;
  - Common File-share;
  - Instant Messaging (Chat); and
  - Structured Messaging.
- That enable the integration of user specified Information Security / Assurance services and infrastructure, specifically: Authentication, Access Control, Cryptography (Key Management, Key Escrow and Transformation), and Privilege management.
- That enable the integration of off-the-shelf User Applications and Dissemination Services.

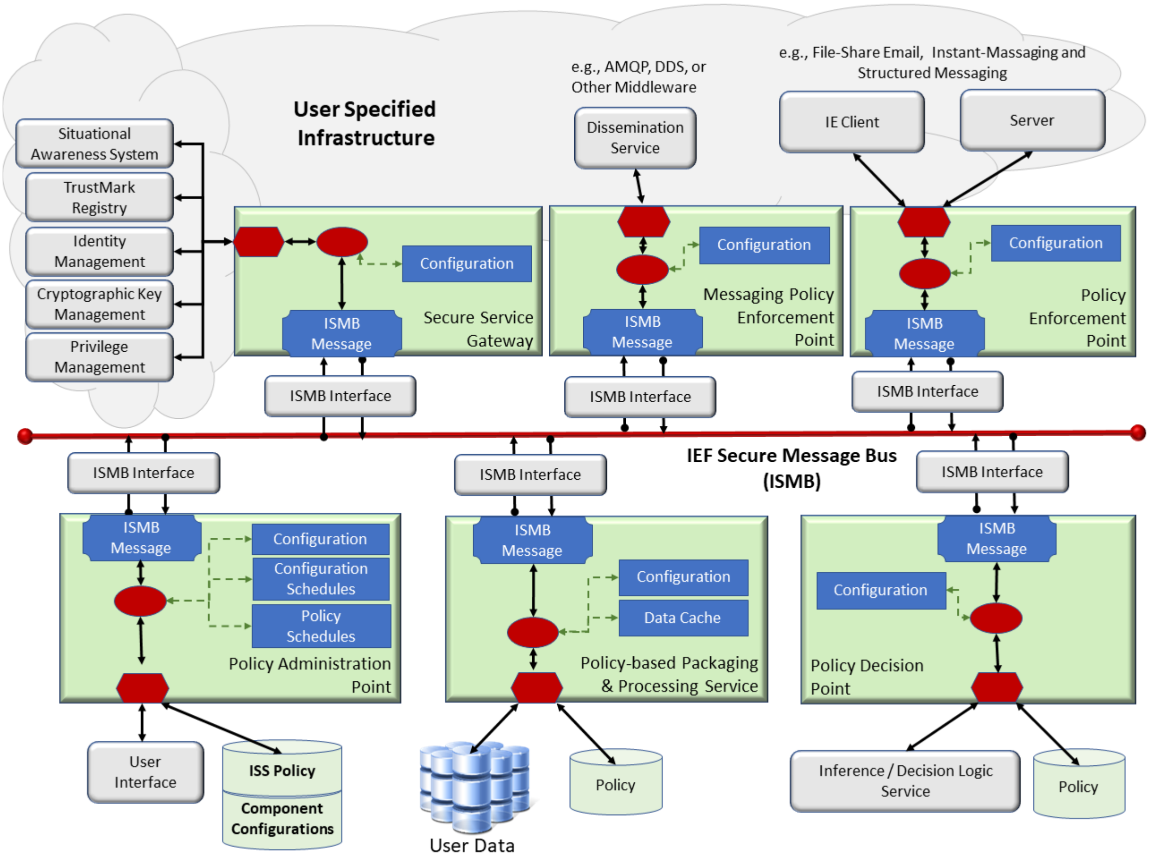


Figure 11 - IEF Service Elements

The reference architecture defines seven services that combine to provide an information sharing and safeguarding capability:

- Policy Enforcement Points (PEP) for Email, File Share, Instant Messaging (Chat), Receiver Direct Messaging and Session Directed Messaging;
- Policy Decision Point (PDP);
- Policy Administration Point (PAP);
- Policy-based Packaging Service (PPS);
- Security Services Gateway (SSG);
- Cryptographic Transformation Services (CTS); and
- Trusted (Tamper Resistant) Logging Services (TLS).

7.2 Simple Deployment

The IEF is designed as an interoperable set of data authorization services that use a proxy server to intercept transactions between a user application and its data distribution services and assures that:

1. The provider is authorized to release the information element in the exchange (message) to the consumer using the specified service; and
2. The recipients(s) are authorized to access and process the information elements in the exchange.

If either of the conditions are not met, the IEF blocks the exchange and issue an alert to the administrator of the environment. Any number of IEF Service Suites can be deployed in the user or community (e.g., NATO) environment to address the size and complexity. Each IEF service suite interfaces with the user's environment through the PEP Proxy and the Security Services Gateway.

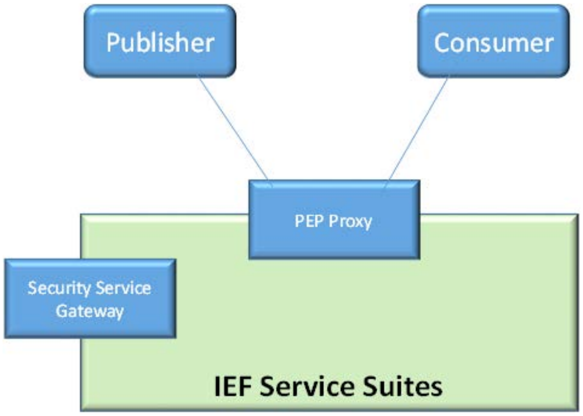


Figure 12 - IEF Instance Deployment

The PEP Proxy and Security Service Interface provide the interfaces to the users' or community's defined communication services and security services. The PEP Proxies:

- Intercept messages between the communication server (e.g., file, email, instant-messaging, and data) and the user’s application. The PEP uses the metadata in the message to verify that the receiver is authorized to access the information in the message; and
- Intercept messages between the user’s application and the communication server (e.g., file, email, instant-messaging, and data). The PEP uses the metadata in the message to verify that the sender is authorized to release the content of the message on the specified communications channel to the specified users.

The Security Services Gateway enables the IEF components (e.g., PEP, PDP, and PAP) to retrieve information (e.g., identity attributes, user authorizations, cryptographic keys and situational attributes) from the users’ or community’s infrastructure. The gateway translates the IEF Messages (Annex A) to the formats and protocols used by the users’ services and infrastructure.

7.3 Interoperable vs. Integrated Services

The IEF-RA defines an interoperable set of software services that communicate using a standardized set of messages over a secure messaging or data bus (ISMB). This form of interoperabilityto:

1. Enables the integration of services from multiple vendors;
2. Enables the addition of multiple services (Figure 13) of the same kind to enable the environment to scale to large data volumes; and
3. Enables the deployment of multiple policy sets in a manner that permitsthe user to participate in multiple communities, using multiple communication interfaces simultaneously (e.g., add PEPs tailored to the communication and service buses applied (e.g., DDS, J2EE, other).

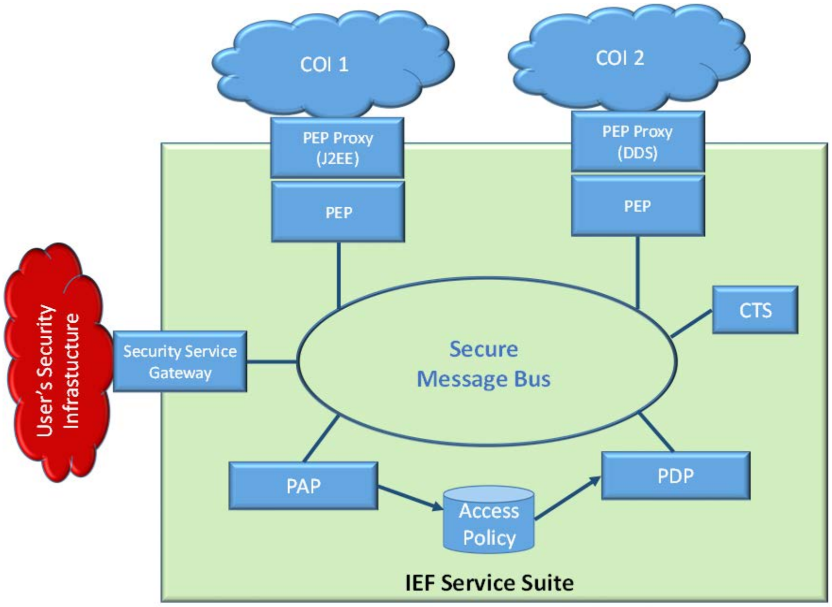


Figure 13 - Interoperable Services

7.4 IEF Component Diagrams

The following clauses describe the elements comprising the core of the Information Exchange Framework:

- IEF Secure Messaging Bus (ISMB);
- Policy Administration Point (PAP);
- Policy Enforcement Points (PEP):
    - Massaging-PEP,
    - Email-PEP,
    - File-PEP, and
    - IM-PEP;
- Policy Decision Point (PDP);
- IEF Security Services Gateway (ISSG);
- Supporting Services:
    - Secure Logging Services, and
    - Cryptographic Services.

The elements combine to provide policy-driven data-centric information safeguards for data at rest or in transit. The PAP, PEP, and PDP reflect a specialization of the XACML specification.

7.4.1 IEF Core Components

The following figure identifies the IEF core components defined by this reference architecture. These components combine to provide a Policy-driven Data-centric Information Sharing and Safeguarding (ISS) environment for structured messaging, file-sharing, instant (/text) messaging and email applications.

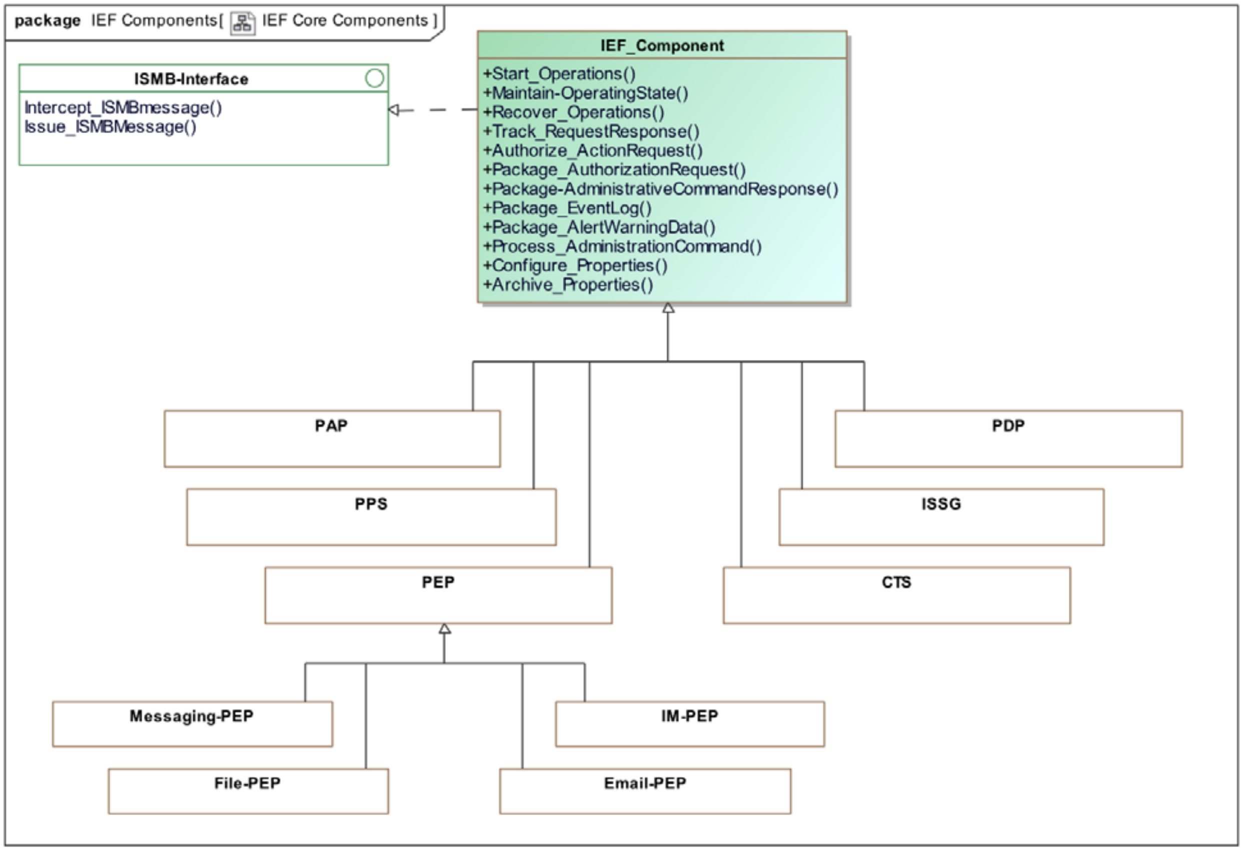


Figure 14 -IEF Core Components

The following table describes the elements illustrated in diagram “IEF Core Components”.

Table 1 - IEF Core Components

Element Name	Operations
CTS	The Cryptographic Transformation Service (CTS) is the IEF component that encrypts and decrypts InformationElements as authorized by policy. The CTS is a bridging component that links cryptographic action requests from the PEP to a FIPS-complaint software module for execution.
Email-PEP	The E-mail PEP operates between the e-mail client and the Mail server. The PEP intercepts each request, determines if the user is authorized to perform the requested action given sensitivities of the information assets (email body and each attachment) involved.

Table 1 - IEF Core Components

Element Name	Operations
File-PEP	The File-PEP operates between a user application and the file server. The PEP intercepts each request, determines if the user is authorized to perform the requested action given sensitivities of each file.
IEF_Component	The IEF-Component identifies a set of software functions and methods that are provided by each of the IEF components: • Policy Administration Point (PAP); • Policy Decision Point (PDP); • Policy Enforcement Point (PEP); • Packaging and Processing Service (PPS); • Cryptographic Transformation Services (CTS); and • IEF Security Services Gateway (ISSG).
IM-PEP	The IM-PEP provides features that intercept all interactions between the users' Instant or Text Massaging client and the Instant Messaging (IM) server. The PEP intercepts each interaction between an IM client and server to determine if the user (sender and receiver) has the appropriate privileges (/authorizations) to perform the requested action.
ISSG	The IEF Security Services Gateway (ISSG) provides a single point for users (vendors and integrators) to integrate IEF components with the users' own security services (e.g., Identity, credential, access-control, and key management) and infrastructure. The ISSG provides the interface to all User specified (delivered) services and infrastructure (e.g., Identity Management).
Messaging-PEP	The Messaging-PEP provides access and release control for: • User applications requesting information from a PPS, and • PPS pushing information to a user application or middleware. The Messaging-PEP authorizes release and receipt based on the users (sender and receiver) privileges and current user specified policy. On receipt, the Messaging-PEP verifies that the local PPS and data-store(s) are authorized to access, process and store the contents of the Message. On Release, the Messaging-PEP ensures that each recipient to the message is authorized to receive the contents of the message, and that the specific communication mechanism (e.g., topic, queue, communication channel) is authorized to transport the contents of the message.

Table 1 - IEF Core Components

Element Name	**Operations**
PAP	The Policy Administration Point (PAP) provides an authorized user (administrator) with an interface to access services needed to manage and administer the configuration and policy environments of IEF Components. The PAP user interface provides the ability to generate and issue command messages (see PAP-Command Message) that direct the IEF Component to perform specified functions. It is anticipated that the user interface will be integrated into the users' own system management or security management services - for this reason, the interface and the PEP are separated services. The core PAP incorporates the PEP responsibilities to authorize each administrator request.
PDP	The Policy Decision Point adjudicates access to, or the release of resources to a specified user. based on: • The sensitivity of the resource; • The privileges of the user; and • (Optional) operational context in which the decision is being made.
PEP	A Policy Enforcement Point (PEP) intercepts each InformationElement transiting between a user client application and the server (Email, Instant Messaging and File Share, and Data) to ensure the requesting user is authorized to perform the requested action on the specified information element(s). The PEP requires that each file is bound with the security, privacy and other user markings required by the PDP to decide if the feature (action) can be performed. There are four (4) specializations for the PEP, including: 1. The File-PEP; 2. The Email-PEP; 3. The IM-PEP; 4. The Messaging-PEP. Each PEP provides features that orchestrate the authorization or rejection of a user access or release request. The PEP packages a PDP-Request Message for each information element in the specified exchange: • Email: Email Body and each attachment; • File: Each individual File; • Instant Message: Each Message and Each Attachment;

Table 1 - IEF Core Components	
Element Name	**Operations**
	• Structured Message: Message, the Digest, each Information Package, each Payload, and each Attachment. Each PEP provides the ability to disassemble an exchange, gather the metadata for each InformationElement, gather authorization data, request authorizations, request cryptographic transformation and reassemble the message for release. The PEP may have the ability to redact information elements based on release instructions from the PDP.
PPS	The Policy-based Packaging and Processing Service (PPS) transitions structured InformationElements (e.g., NIEM, EDXL, and HL7) between data stores and information exchange services in accordance with local information sharing and safeguarding policies conforming to the Information Exchange Packaging Policy Vocabulary (IEPPV). The PPS provides the ability to selectively package (aggregate, transform, mark, filter, structure and format) informationElements for publication to authorized recipients. It also provides the ability to process (parse, transform, and marshal) structured messages and integrate the data elements into the user's data stores.

7.5 IEF Component Interactions

The following sub-clauses outline the messaging interfaces between the core IEF elements.

7.5.1 ISMB Interfaces

The following figure identifies the IEF component interfaces to the ISMB.

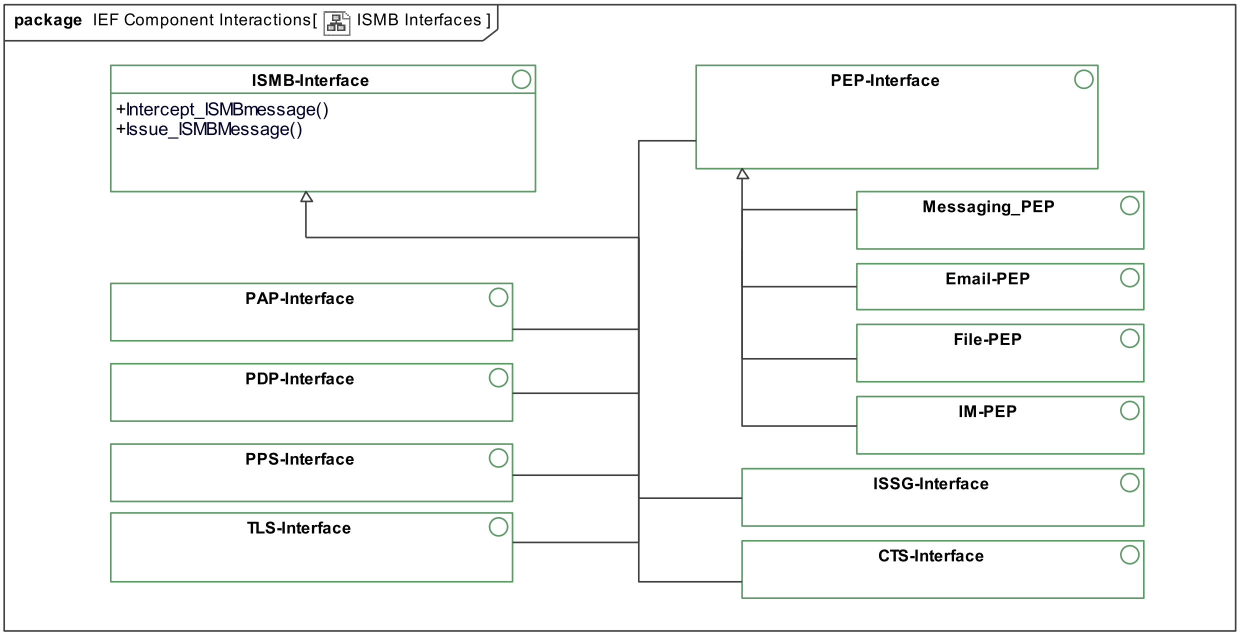


Figure 15 -ISMB Interfaces

The following table identifies and describes the elements and operations illustrated in diagram "ISMB Interfaces".

Table 2 - ISMB Interfaces	
Element Name	**Operations**
CTS-Interface	Interface between the Cryptographic Transformation Services (CTS) and the IEF Secure Messaging Bus (ISMB). Messages applicable to the CTS include: • PAP-Command (receive); • PAP-Command-response (send); • CTS-Request (receive); • CTS-Response (send); and • CTS-LogMessage (send). Interface provides the features needed to intercept and parse messages from the ISMB and package and issue messages to the ISMB.
	Element Type: *Interface* ***Inherited Operations:*** *Intercept_ISMBmessage* inherited from *ISMB-Interface* *Issue_ISMBMessage* inherited from *ISMB-Interface*

Table 2 - ISMB Interfaces

Element Name	**Operations**
Email-PEP	ISMB interface for the specialized PEP governing the exchange of Email messages between the user's Email Client and Email Server.
	Element Type: *Interface* ***Inherited Operations:*** *Intercept_ISMBmessage* inherited from *ISMB-Interface* *Issue_ISMBMessage* inherited from *ISMB-Interface*
File-PEP	ISMB interface for the specialized PEP governing the receipt and release of a file or files between the user application and the file server.
	Element Type: *Interface* ***Inherited Operations:*** *Intercept_ISMBmessage* inherited from *ISMB-Interface* *Issue_ISMBMessage* inherited from *ISMB-Interface*
IM-PEP	ISMB interface for the specialized PEP governing the receipt and release of instant (/text) messages and the Instant Messaging (IM) Server.
	Element Type: *Interface* ***Inherited Operations:*** *Intercept_ISMBmessage* inherited from *ISMB-Interface* *Issue_ISMBMessage* inherited from *ISMB-Interface*
ISMB-Interface	Interface features supported by each IEF Component interface to the ISMB. ISMB Messages Include: • CTS-Request; • CTS-Response; • ISSG-request; • ISSG-Response; • PAP-Command; • PAP-CommandResponse; • PAP-AlertWarning; • PDP-Request; • PDP-Response; • PPS-Publish; • PPS-Receive;

Table 2 - ISMB Interfaces

Element Name	**Operations**
	• PPS-Request; and • TLS-LogMessage.
	Element Type: *Interface* ***Owned Operations:*** *Intercept_ISMBmessage:* The ISMB-Interface provides features that listen to the IEF Secure Messaging Bus (ISMB) for messages directed to the specific component, intercepts the appropriate messages and directs the message for processing. The interface then parses and extracts metadata and InformationElements, from the message, and passes the individual element to the component for processing. *Issue_ISMBMessage:* The ISMB-Interface provides features that apply the ISMB Messaging protocol and writes the formatted message to the ISMB (middleware) queue, topic or channel.
ISSG-Interface	Interface between the IEF Security Services Gateway (ISSG) and the IEF Secure Messaging Bus (ISMB). Messages applicable to the ISMB include: • ISSG-Request (receive); • ISSG-Response (send); • PAP-AlertWarning (send); and • TLS-LogMessage (send). Interface provides the features needed to both intercept and parse messages from the ISMB, and package and issue messages to the ISMB.
	Element Type: *Interface* ***Inherited Operations:*** *Intercept_ISMBmessage* inherited from *ISMB-Interface* *Issue_ISMBMessage* inherited from *ISMB-Interface*
Messaging_PEP	ISMB interface for a PEP specialized for the receipt and release of structured messages. This PEP is paired with the Policy-based Packaging and Processing Service (PPS).
	Element Type: *Interface* ***Inherited Operations:*** *Intercept_ISMBmessage* inherited from *ISMB-Interface* *Issue_ISMBMessage* inherited from *ISMB-Interface*

Table 2 - ISMB Interfaces

Element Name	**Operations**
PAP-Interface	Interface between the Policy Administration Point (PAP) and the IEF Secure Messaging Bus (ISMB). Messages applicable to the PAP include: • PAP-Command (send); • PAP-CommandResponse (receive); • PAP-AlertWarning (receive); • PDP-OperationAuthorizationRequest (sent); • PDP-OperationAuthorizationResponse (receive); • ISSG-Request (sent); • ISSG-Response (receive); and • TLS-LogMessage (send). Interface provides the features needed to intercept and parse messages from the ISMB and package and issue messages to the ISMB.
	Element Type: *Interface* ***Inherited Operations:*** *Intercept_ISMBmessage* inherited from *ISMB-Interface* *Issue_ISMBMessage* inherited from *ISMB-Interface*
PDP-Interface	ISMB interface for Policy Decision Point (PDP). Messages applicable to the PDP include: • PDP-Request (receive); • PDP-Response (send); • PAP-Command (receive); • PAP-CommandResponse (send); • PAP-AlertWarning (send); • ISSG-Request (send); • ISSG-Response (receive); and • TLS-LogMessage (send). Interface provides the features needed to intercept and parse messages from the ISMB and package and issue messages to the ISMB.

Table 2 - ISMB Interfaces

Element Name	Operations
	Element Type: *Interface* ***Inherited Operations:*** *Intercept_ISMBmessage* inherited from *ISMB-Interface* *Issue_ISMBMessage* inherited from *ISMB-Interface*
PEP-Interface	ISMB interface for a PEP, which is responsible for enforcing access and release control policy determination by the PDP. Messages applicable to the PAP include: • PAP-Command (receive); • PAP-CommandResponse; • PAP-AlertWarning (send); • PDP-Request (send); • PDP-Response (receive); • PPS-Publish (receive); • PPS-Receive (send); • PPS-Request (send); • ISSG-Request (send); • ISSG-Response (receive); and • TLS-LogMessage (send).
	Element Type: *Interface* ***Inherited Operations:*** *Intercept_ISMBmessage* inherited from *ISMB-Interface* *Issue_ISMBMessage* inherited from *ISMB-Interface*
PPS-Interface	ISMB interface for a Policy-based Packaging Service (PPS). Messages applicable to the PPS include: • PPS-Receive (receive); • PPS-Publish (send); • PPS-Request (receive); • PAP-Command (receive); • PAP-CommandResponse; • PAP-AlertWarning (send); • ISSG-Request (send); • ISSG-Response (receive);

Table 2 - ISMB Interfaces	
Element Name	**Operations**
	• CTS-Request (send); • CTS-Response (receive); and • TLS-LogMessage (send). Interface provides the features needed to intercept and parse messages from the ISMB and package and issue messages to the ISMB.
	Element Type: *Interface* ***Inherited Operations:*** *Intercept_ISMBmessage* inherited from *ISMB-Interface* *Issue_ISMBMessage* inherited from *ISMB-Interface*
TLS-Interface	ISMB interface to the IEF Trusted Logging Service (TLS). Messages applicable to the PAP include: • TLS-LogMessage (receive); • PAP-Command (receive); and • PAP-CommandResponse (send).
	Element Type: *Interface* ***Inherited Operations:*** *Intercept_ISMBmessage* inherited from *ISMB-Interface* *Issue_ISMBMessage* inherited from *ISMB-Interface*

7.5.2 Component Interfaces

The following figure illustrates the dependencies between the core IEF components. Each of the identified components use the IEF Secure Messaging Bus (ISMB) to communicate with, and obtain services from other IEF components. The ISMB isolates the IEF inter-component communication from the user's broader information sharing environment. The messages exchanged between the IEF components include:

- PAP-Command;
- PAP-Command Response;
- PAP-AlertWarning;
- PDP-Information Exchange Authorization Request;
- PDP-Information Exchange Authorization Response;
- PDP-Operation Authorization Request;
- PDP-Operation Authorization Response;

- PPS-Receive;
- PPS-Publish;
- PPS-Request;
- ISSG-Request;
- ISSG-Response;
- CTS-Request;
- CTS-Response; and
- TLS Log Entry.

Content Descriptions for each of these messages are found in Annex A: ISMB Messages.

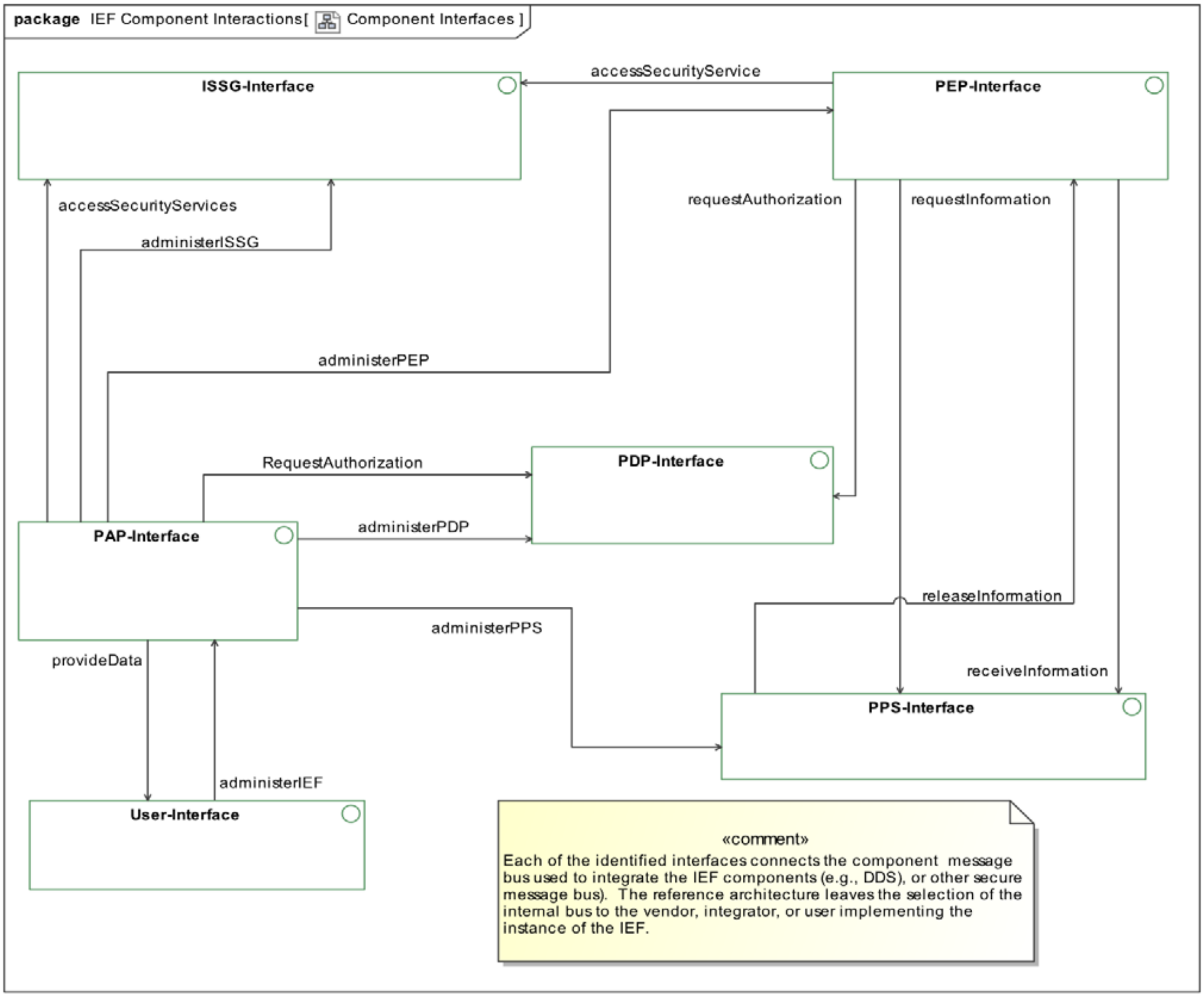

package IEF Component Interactions[Component Interfaces]
ISSG-Interface
PEP-Interface
accessSecurityService
accessSecurityServices
administerISSG
requestAuthorization
requestInformation
administerPEP
RequestAuthorization
PDP-Interface
PAP-Interface
administerPDP
releaseInformation
administerPPS
provideData
receiveInformation
PPS-Interface
administerIEF
User-Interface
«comment»
Each of the identified interfaces connects the component message bus used to integrate the IEF components (e.g., DDS), or other secure message bus). The reference architecture leaves the selection of the internal bus to the vendor, integrator, or user implementing the instance of the IEF.

Figure 16 -Component Interfaces

The following table describes the core features of the associations illustrated in diagram "Component Interfaces".

Table 3 – Component Interfaces

Element	Interface Interactions
ISSG-Interface	Interface between the IEF Security Services Gateway (ISSG) and the IEF Secure Messaging Bus (ISMB). Messages applicable to the ISMB include: • ISSG-Request (receive); • ISSG-Response (send); • PAP-AlertWarning (send); and • TLS-LogMessage (send). Interface provides the features needed to both intercept and parse messages from the ISMB, and package and issue messages to the ISMB. The ISSG-Interface enables the component to access the following services needed to perform its assigned function: ***The component does not use other IEF components to perform its function.***
PAP-Interface	Interface between the Policy Administration Point (PAP) and the IEF Secure Messaging Bus (ISMB). Messages applicable to the PAP include: • PAP-Command (send); • PAP-CommandResponse (receive); • PAP-AlertWarning (receive); • PDP-OperationAuthorizationRequest (sent); • PDP-OperationAuthorizationResponse (receive); • ISSG-Request (sent); • ISSG-Response (receive); and • TLS-LogMessage (send). Interface provides the features needed to intercept and parse messages from the ISMB and package and issue messages to the ISMB. The PAP-Interface enables the component to access the following services needed to perform its assigned function: ***The component does not use other IEF components to perform its function.***

PDP-Interface	ISMB interface for Policy Decision Point (PDP). Messages applicable to the PDP include: • PDP-Request (receive); • PDP-Response (send); • PAP-Command (receive); • PAP-CommandResponse (send); • PAP-AlertWarning (send); • ISSG-Request (send); • ISSG-Response (receive); and • TLS-LogMessage (send). Interface provides the features needed to intercept and parse messages from the ISMB and package and issue messages to the ISMB. The PDP-Interface enables the component to access the following services needed to perform its assigned function: ***The component does not use other IEF components to perform its function.***
PEP-Interface	ISMB interface for a PEP, which is responsible for enforcing access and release control policy determination by the PDP. Messages applicable to the PAP include: • PAP-Command (receive); • PAP-CommandResponse; • PAP-AlertWarning (send); • PDP-Request (send); • PDP-Response (receive); • PPS-Publish (receive); • PPS-Receive (send); • PPS-Request (send); • ISSG-Request (send); • ISSG-Response (receive); and • TLS-LogMessage (send). The PEP-Interface enables the component to access the following services needed to perform its assigned function: ***The component does not use other IEF components to perform its function.***
PPS-Interface	ISMB interface for a Policy-based Packaging Service (PPS). Messages applicable to the PPS include: • PPS-Receive (receive);

	• PPS-Publish (send); • PPS-Request (receive); • PAP-Command (receive); • PAP-CommandResponse; • PAP-AlertWarning (send); • ISSG-Request (send); • ISSG-Response (receive); • CTS-Request (send); • CTS-Response (receive); and • TLS-LogMessage (send). Interface provides the features needed to intercept and parse messages from the ISMB and package and issue messages to the ISMB. The PPS-Interface enables the component to access the following services needed to perform its assigned function: ***The component does not use other IEF components to perform its function.***
User-Interface	Interface between the User Interface and the Policy Administration Point (PAP). This interface provides an authorized user to access PAP features in accordance with the user's privileges and IEF Access control policies. The User-Interface enables the component to access the following services needed to perform its assigned function: ***The component does not use other IEF components to perform its function.***

7.5.3 Additional Component Interfaces

The following figure illustrates the dependencies between the core IEF components and the supporting components: Messaging Service and Logging Service. Each of the components provides an interface to the IEF Secure Messaging Bus (ISMB) that enables communication with other IEF components using a standard based XML messaging protocol.

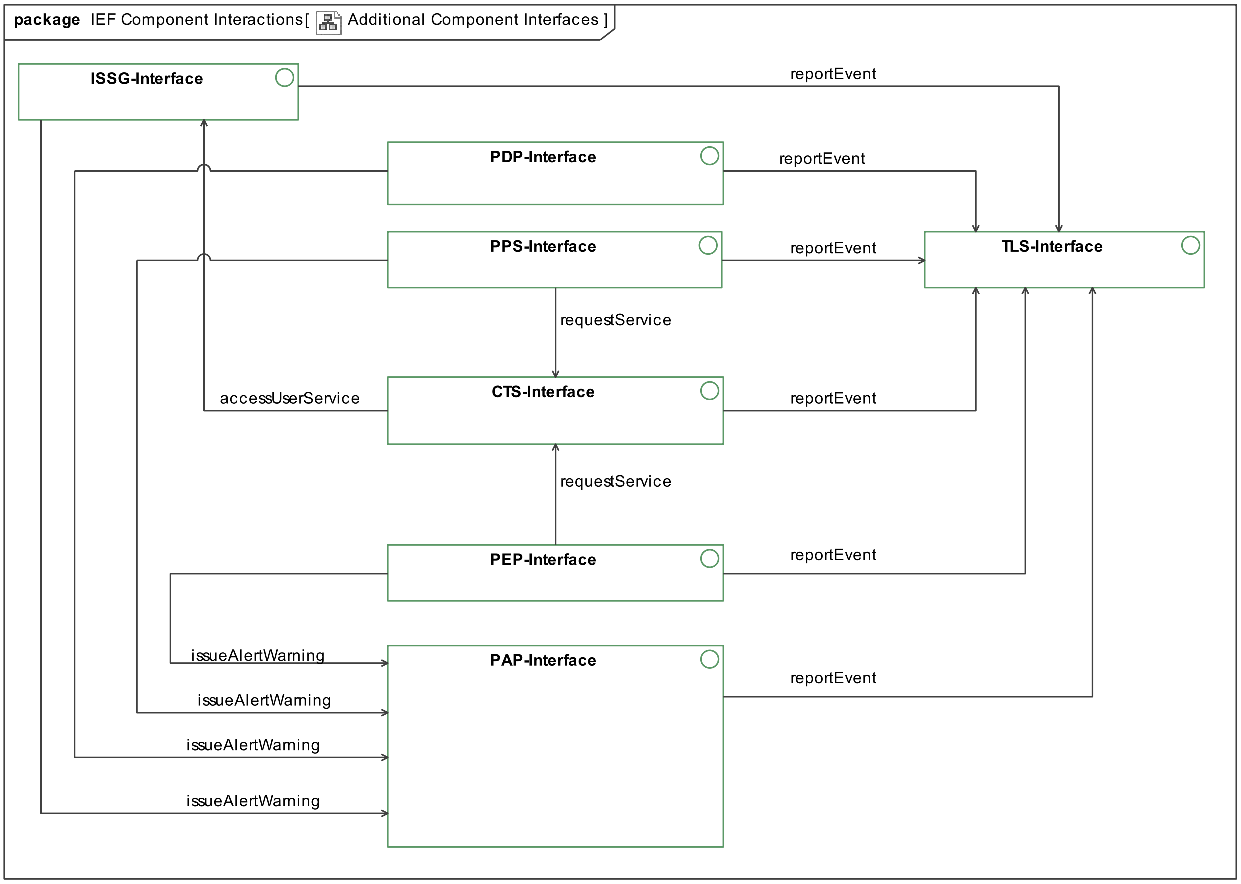


Figure 17 -**Additional Component Interfaces**

These interfaces are described with the IEF components provisoning the interfaces.

8 Policy Administration Point

The Policy Administration Point[5] is a single entity which is the source of policies for a given domain. The IEF defines a general architecture for a PAP, which identifies its core sub-components, its functions, and its interfaces (e.g., protocols and content).

8.1 PAP in an Inter-Agency Environment

A core design principle is that the ownership/control of policies rests with the owner/steward of the data asset. This implies that:

1. Policies (rules /instructions / constraints) are developed, tested, administered and managed separately from the processes, services and technologies used to enforce them;
2. Data owners/stewards have full control over the release of or access to data/information assets; and
3. Policies (executable, or actionable policies) reflect and are traceable to the policy instruments endorsed by each data owner / steward.

It is critical that the IEF separate the concerns of data stakeholders/owners (control access to, and release of information assets), and the needs of communities to develop and deploy common/shared services and infrastructure. IEF conformant environments enables stakeholders to develop ISS policies (/rules / metadata) under a process that is independent of the process for developing the services that enforce the ISS policies (/rules). A machine-readable serialization of ISS policy is ingested, at runtime, by common/shared services that enforce stakeholder specified rules. The IEF identifies and defines service interface specifications (here-in), for policy administration, decision and enforcement services that enables the integration of one or multiple vendor components into a desired ISS capability.

8.2 Policy Administration Point (PAP)

The Policy Administration Point (PAP) provides an authorized user with an interface and services needed to manage and administer IEF components and policies. Within the limits of the user's security authorizations and policy, the PAP enables the user with the ability to manage and administer IEF Component policies and component configurations.

8.2.1 PAP Component Operations

The following figure illustrates the core features of the Policy Administration Point (PAP) and its features that enable an authorized user (IEF Administrator) to manage and administer IEF operations, component configurations, and component policy.

[5] The Policy Administration Point descriptions and definitions were derived from those XACML definitions. (https://wiki.oasis-open.org/xacml/Policy%20Administration%20Point%20Architecture)

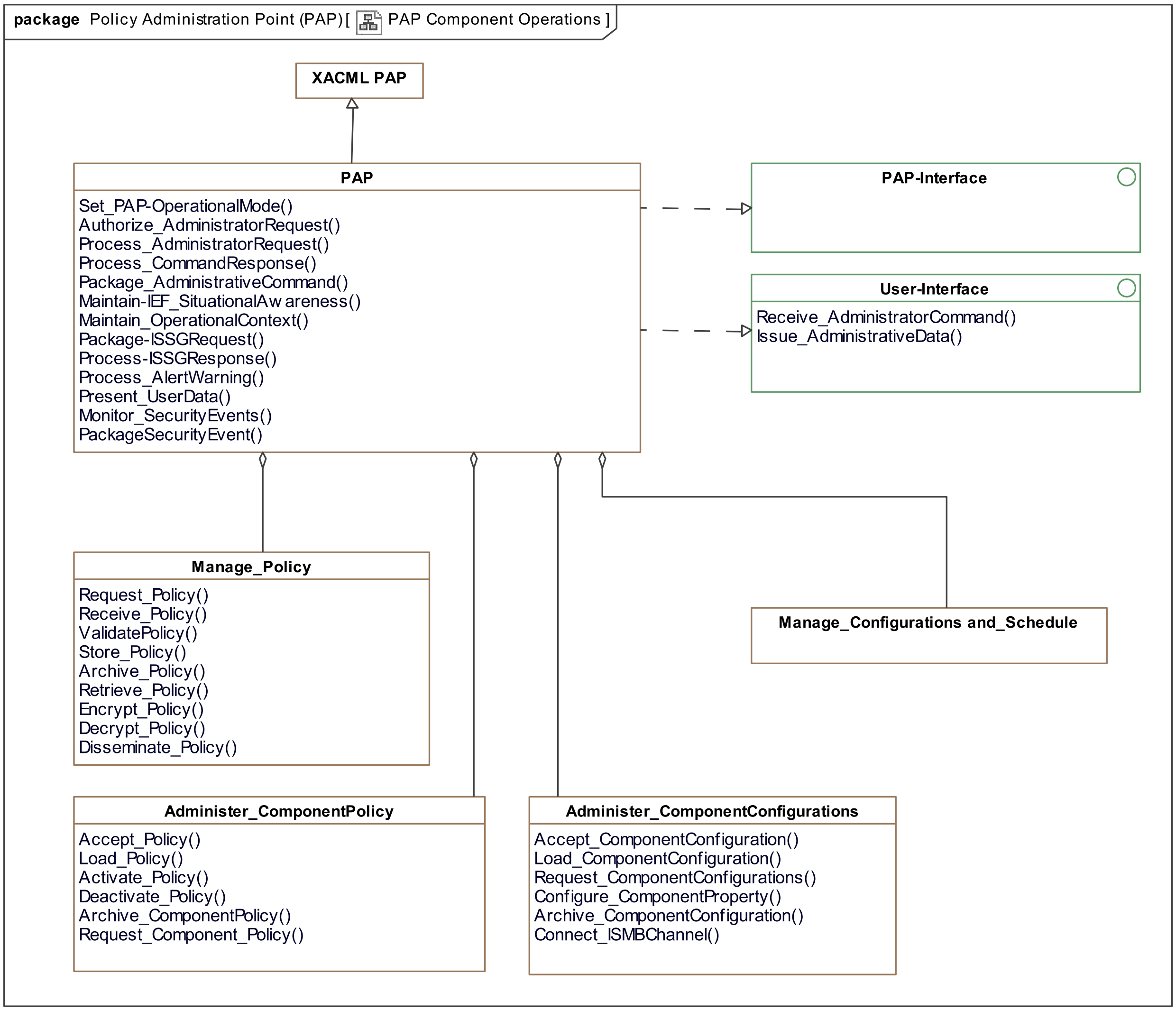

package Policy Administration Point (PAP) [PAP Component Operations]
XACML PAP
PAP
Set_PAP-OperationalMode()
Authorize_AdministratorRequest()
Process_AdministratorRequest()
Process_CommandResponse()
Package_AdministrativeCommand()
Maintain-IEF_SituationalAw areness()
Maintain_OperationalContext()
Package-ISSGRequest()
Process-ISSGResponse()
Process_AlertWarning()
Present_UserData()
Monitor_SecurityEvents()
PackageSecurityEvent()
PAP-Interface
User-Interface
Receive_AdministratorCommand()
Issue_AdministrativeData()
Manage_Policy
Request_Policy()
Receive_Policy()
ValidatePolicy()
Store_Policy()
Archive_Policy()
Retrieve_Policy()
Encrypt_Policy()
Decrypt_Policy()
Disseminate_Policy()
Manage_Configurations and_Schedule
Administer_ComponentPolicy
Accept_Policy()
Load_Policy()
Activate_Policy()
Deactivate_Policy()
Archive_ComponentPolicy()
Request_Component_Policy()
Administer_ComponentConfigurations
Accept_ComponentConfiguration()
Load_ComponentConfiguration()
Request_ComponentConfigurations()
Configure_ComponentProperty()
Archive_ComponentConfiguration()
Connect_ISMBChannel()

Figure 18 -PAP Component Operations

The following table identifies and describes the elements and operations illustrated in diagram "PAP Component Operations".

Table 4 - PAP Component Operations	
Element Name	**Operations**
Administer_ComponentConfigurations	The PAP provides features that enable an authorized user (/administrator) to configure the operation of IEF components (e.g., PAP, PDP, PEPs, ISSG and ISMB) in the operational environment. The user interface and command generation features for each component type form part of the PAP that are issued to the individual components where they are executed.

Table 4 - PAP Component Operations	
Element Name	**Operations**
	These features enable an authorized user to manage and administer the operational configurations of IEF components assigned to the PAP. Management features include the ability to: • Configure IEF Components: ISMB, PDPs, PEPs, PPSs, and ISSG; • Administer information about component configurations, profiles and policies; • Define and administer policy and configuration deployment schedules; • Manage and maintain environment policy store; • Log all changes to the PAP and component configurations; and • Log all changes to PDP and PPS policy environments. The IEF Reference Architecture in primarily focused on the identification of interfaces and required functions to operate a policy-driven data-centric ISS solution - the determination on whether these are manual or automated functions and interfaces is determined by the user and integrators. In each case the PAP packages a PAP-Command containing the user's directive and issues the message to the IEF component using the ISMB.
	Element Type: *Class* ***Owned Operations:*** *Accept_ComponentConfiguration:* The PAP provides features that enable an authorized user to package and issue a PAP-Command message that instructs an IEF component to receive a component configuration included in a PAP-Command message or from a specified location in the IEF protected information store. *Load_ComponentConfiguration:* The PAP provides features that enable an authorized user to package and issue a PAP-Command message that directs an IEF component to load a previously accepted component configuration. *Request_ComponentConfigurations:* The PAP provides features that enable an authorized user to package and issue a PAP-Command message that instructs an IEF component to package its operating configuration and store it to a specified location in the IEF protected information store. *Configure_ComponentProperty:*

Table 4 - PAP Component Operations

Element Name	**Operations**
	The PAP provides features that enable an authorized user to package and issue a PAP-Command message that instructs a component to change or alter the value of one or more of its configurable component properties. The component properties determine how each component performs its role within the IEF environment and communicates with supporting services. The PAP provides the interface for an authorized user to issue changes to one or more components, which execute the changes. *Archive_ComponentConfiguration:* The PAP provides features that enable an authorized user to package and issue a PAP-Command message that instructs a component to assemble and package its current configuration and return to the PAP (see PAP-CommandResponse message) or persist it as a file in a Secure Asset Container within the IEF protected information store. *Connect_ISMBChannel:* The PAP provides features that enable an authorized user to package and issue a PAP-Command message that instructs an IEF component to use an ISMB communication channel.
Administer_ComponentPolicy	The PAP provides features that enable an authorized user (/administrator) to direct a PDP or PPS to adjust or configure its policy environment. In each case the PAP packages a PAP-Command message containing the user's directives and issues the message to the PDP or PPS to be executed.
	Element Type: *Class* ***Owned Operations:*** *Accept_Policy:* The PAP provides features that enable an authorized user to package and issue a PAP-Command message that instructs an IEF component to receive a policy or policy sets included in a PAP-Command message or from a specified location in the IEF protected information store. The instructions are packaged and issued as a PAP-Command message. *Load_Policy:* The PAP provides features that enable an authorized user to package and issue a PAP-Command message that directs either a PDP or PPS to load a previously accepted policy or set of policies into its policy environment. *Activate_Policy:* The PAP provides features that enable an authorized user to package and issue a PAP-Command message that instructs a

Table 4 - PAP Component Operations

Element Name	**Operations**
	PDP or PPS to activate one or more of its policies previously loaded into its policy environment. *Deactivate_Policy:* The PAP provides features that enable an authorized user to package and issue a PAP-Command message that instructs a PDP or PPS to deactivate one or more of its policies that was previously loaded into its policy environment. *Archive_ComponentPolicy:* The PAP provides features that enable an authorized user to package and issue a PAP-Command message to instruct a PDP or PPS to package its current policy environment and persist it to a specified location in the IEF environment or return it to the PAP as part of a PAP-CommandResponse Message. *Request_Component_Policy:* The PAP provides features that enable an authorized user to package and issue a PAP-Command message that instructs a PPS, or PDP to package one or more policies and return them as part of a PAP-CommandResponse message.
Manage_Configurations_and_Schedule	(Optional) PAP features that enable an authorized user to plan, schedule and execute a policy dissemination and activation schedule for a deployed IEF environment. These features are intended to enable the scheduled adaptation of IEF component operation to the phases of a mission or operation. The features may include: • Plan a policy dissemination and activation schedule. • Plan a configuration schedule for each IEF component. • Load an IEF configuration file(s). • Disseminate component configurations: ○ Generate PAP-Command (configuration change) message; ○ Issue PAP-Command; and ○ Log Event. • Deactivate policy schedules. • Tailor policy schedules. • Execute policy schedules. • Activate configuration schedules; ○ Track operational time and/or operational context;

Table 4 - PAP Component Operations

Element Name	**Operations**
	○ Generate PAP-Command (policy change) message (change policy configuration of the PPS and/or PDP); ○ Log Change; ○ Issue PAP-Command message; and ○ Log response. • Deactivate configuration schedules. • Modify configuration schedules.
	Element Type: *Class*
Manage_Policy	PAP features that enable an authorized user to manage ISS policy sets for the PDPs (access and release control policy) and PPSs (data packaging and processing policy) in the environment. Multiple sets of policies may be required to accommodate changes in operational context (e.g., threat, risk, operational partners, mission phase, roles and responsibilities). Each change in context may require a change in data and information sharing and safeguarding requirements. The PAP must provide the user with the ability to manage these changes.
	Element Type: *Class* ***Owned Operations:*** *Request_Policy:* The PAP provides features that enable an authorized user to request, receive and process a Policy Set from a user specified Policy Management or Development Environment. The request is packaged and issued as an ISSG-Request message. The policies are received as a SecureAssetContainer through the ISSG. (see ISSG-Request message) *Receive_Policy:* The PAP provides features that enable the PAP to receive and process a requested Policy Set received from the ISSG. The Policy Set is received as a SecureAssetContainer from the ISSG. (see ISSG-Response message) *ValidatePolicy:* The PAP provides features that validate and verify the format and content of an ISS policySet for the PDP or PPS. These features assure the provisioned policy is from an authorized source, and conforms to ISS policy language specifications (e.g., IEPPV). ISS policies are XML documents that describe the rules and constraints used by the PPS and PDP to govern

Table 4 - PAP Component Operations	
Element Name	**Operations**
	the packaging and release of information elements from the IEF. *Store_Policy:* The PAP provides feature(s) that enable an authorized user to persist a set of policies to a specified location within the IEF protected information store. The PAP stores policies pertinent to anticipated operational situations (/contexts). Policies are stored as a file within a Secure Access Container. *Archive_Policy:* The PAP provides features that package policy sets for transmission to a user specified policy management or development environment. Policies exported as a file within Secure Access Container through the ISSG. *Retrieve_Policy:* The PAP provides features that retrieve a policy set from the IEF protected information store. The policy sets are stored as a file within a Secure Access Container. *Encrypt_Policy:* The PAP provides features that package and issue a request to the CTS to transform the policy or policy-set into an unintelligible form using the Cryptographic Key provided by the Key Generation Service. *Decrypt_Policy:* The PAP provides features that package and issue a request to the CTS to transform the content of a policy set back into its original form using the unique symmetric key retrieved from escrow. *Disseminate_Policy:* The PAP provides features that enable an authorized user to disseminate a policy or set of policies to the PDP or PEP.
PAP	The Policy Administration Point (PAP) provides an authorized user (administrator) with an interface to access services needed to manage and administer the configuration and policy environments of IEF Components. The PAP user interface provides the ability to generate and issue command messages (see PAP-Command Message) that direct the IEF Component to perform specified functions. It is anticipated that the user interface will be integrated into the users' own system management or security management services - for this reason, the interface and the PEP are separated services. The core PAP incorporates the PEP responsibilities to authorize each administrator request.

Table 4 - PAP Component Operations

Element Name	**Operations**
	Element Type: *Class* ***Owned Operations:*** *Set_PAP-OperationalMode:* The PAP provides features that enable an authorized user to select the mode of operation for the PAP. PAP Modes of operation include: • Basic: Provides the user with the most basic functionality of the PAP and access to limited control of each of the IEF components and policies. • Advanced: Provides the user with the full functionality of the PAP and access to fine-grained control of each of the IEF components and policies. • Test: Provides the user with an extended set of features that enable debugging and testing of PAP and/or IEF Operations. *Authorize_AdministratorRequest:* The PAP provides features that intercept user command requests, extract the command type and pertinent data elements, and interact with the PDP to authorize each user request. (See PDP-Authorizationrequest message). *Process_AdministratorRequest:* The PAP provides features that stage the processing of an authorized command request, including: • Manage PAP configuration and operation; • Manage and administer PDP and PPS policy; • Administer IEF component configurations; • Execute policy and configuration schedules; and • Maintain operational awareness. Each user request must be authorized by the PDP prior to execution by the PAP. *Process_CommandResponse:* The PAP provides features that parse the PAP-CommandResponse message from an IEF component, extracts the pertinent data, integrates the data elements into the PAP's situational awareness, and presents the response to the user interface. *Package_AdministrativeCommand:* The PAP provides features that gather the data elements needed for a command request to an IEF component, and

Table 4 - PAP Component Operations

Element Name	Operations
	format the data into the appropriate message form. (See PAP-Command Response Message) *Maintain-IEF_SituationalAwareness:* The PAP provides features that gather, store and maintain data describing the current operating state of the IEF environment and each of its components. Information about each component may include: Component ID and Name; Component Description; Component Role(s); Component Configuration; Provisioned Services; Status and Events (alerts/warnings). *Maintain_OperationalContext:* (Optional) The PAP provides features that collect and maintain information about the operational context for the IEF. The operational context may identify: • The operational phase; • The operational partners; • The partners roles and responsibilities; • Context specific information sharing policies; and • The user specified quality of service. Operational context may be used to influence the configuration of the IEF components, and the application of policy by the PDP. The PAP gathers relevant data though PAP communications with the user's Situational Awareness (SA) or incident management services. Communication (messages) between the PAP and user services is conducted through the ISSG. *Package-ISSGRequest:* The PAP provides features that gather the data elements needed for an ISSG request, and format the data into the appropriate message form. (See PAP-Command Response Message) *Process-ISSGResponse:* The PAP provides features that parse the ISSG-Response message containing situational awareness and context information, extracts the pertinent data, and integrates the data elements into the PAP's situational awareness data store. (see ISSG-Response message) *Process_AlertWarning:* The PAP provides features that receive a PAP-AlertWarning message, extracts its type and relevant data elements, integrates the data into the PAP situational awareness

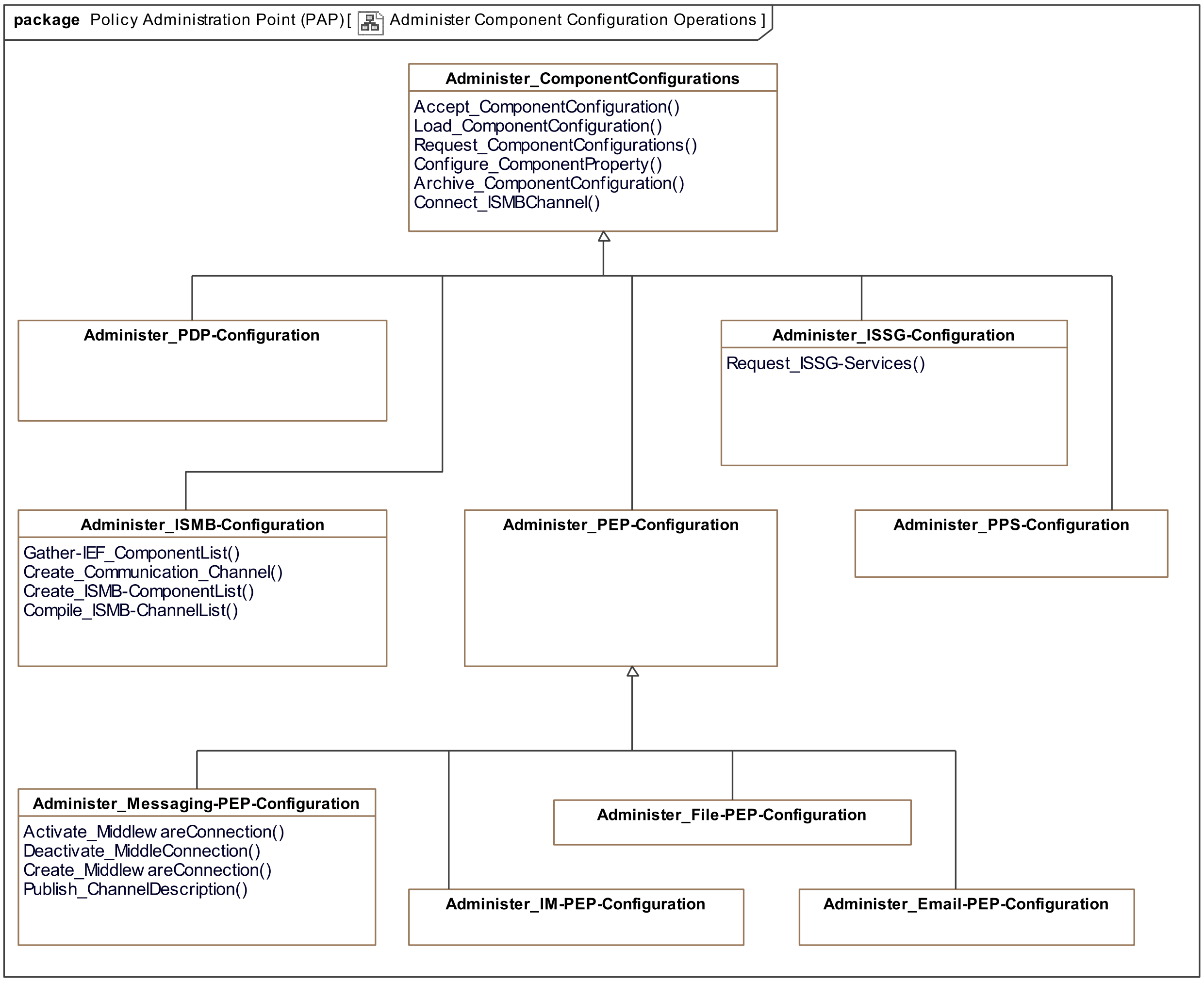
Table 4 - PAP Component Operations	
Element Name	**Operations**
	information and issues the data to the user. (see PAP-AlertWarning message) *Present_UserData:* The PAP provides features that gather information from the execution of PAP features and the situational awareness data store, package the data, and send it for presentation to the user. *Monitor_SecurityEvents:* The PAP provides features that monitor and report (e.g., present to the user) security events provided as PAP-AlertWarning messages from the IEF Components. *PackageSecurityEvent:* The PAP provides features that gather alert, warning and supporting data and package a PAP-SecurityEvent message for release to the Users' security personnel. This information may be distributed using the features of the IM, Email and Messaging PEPs. ***Inherited Operations:*** *Start_Operations* inherited from *IEF_Component* *Maintain-OperatingState* inherited from *IEF_Component* *Recover_Operations* inherited from *IEF_Component* *Track_RequestResponse* inherited from *IEF_Component* *Authorize_ActionRequest* inherited from *IEF_Component* *Package_AuthorizationRequest* inherited from *IEF_Component* *Package-AdministrativeCommandResponse* inherited from *IEF_Component* *Package_EventLog* inherited from *IEF_Component* *Package_AlertWarningData* inherited from *IEF_Component* *Process_AdministrationCommand* inherited from *IEF_Component* *Configure_Properties* inherited from *IEF_Component* *Archive_Properties* inherited from *IEF_Component*
XACML PAP	The IEF PAP was derived from the core concepts defined by the XACML specification.

Table 4 - PAP Component Operations

Element Name	Operations
	Element Type: *Class*

8.2.2 Administer Component Configuration Operations

The following figure further refines the configuration management features of the PAP.



package Policy Administration Point (PAP) [Administer Component Configuration Operations]
Administer_ComponentConfigurations
Accept_ComponentConfiguration()
Load_ComponentConfiguration()
Request_ComponentConfigurations()
Configure_ComponentProperty()
Archive_ComponentConfiguration()
Connect_ISMBChannel()
Administer_PDP-Configuration
Administer_ISSG-Configuration
Request_ISSG-Services()
Administer_ISMB-Configuration
Gather-IEF_ComponentList()
Create_Communication_Channel()
Create_ISMB-ComponentList()
Compile_ISMB-ChannelList()
Administer_PEP-Configuration
Administer_PPS-Configuration
Administer_Messaging-PEP-Configuration
Activate_Middlew areConnection()
Deactivate_MiddleConnection()
Create_Middlew areConnection()
Publish_ChannelDescription()
Administer_File-PEP-Configuration
Administer_IM-PEP-Configuration
Administer_Email-PEP-Configuration

Figure 19 -Administer Component Configuration Operations

The following table identifies and describes the elements and operations illustrated in diagram "Administer Component Configuration Operations".

Table 5 - Administer Component Configuration Operations	
Element Name	**Operations**
Administer_ComponentConfigurations	The PAP provides features that enable an authorized user (/administrator) to configure the operation of IEF components (e.g., PAP, PDP, PEPs, ISSG and ISMB) in the operational environment. The user interface and command generation features for each component type form part of the PAP that are issued to the individual components where they are executed. These features enable an authorized user to manage and administer the operational configurations of IEF components assigned to the PAP. Management features include the ability to: • Configure IEF Components: ISMB, PDPs, PEPs, PPSs, and ISSG; • Administer information about component configurations, profiles and policies; • Define and administer policy and configuration deployment schedules; • Manage and maintain environment policy store; • Log all changes to the PAP and component configurations; and • Log all changes to PDP and PPS policy environments. The IEF Reference Architecture in primarily focused on the identification of interfaces and required functions to operate a policy-driven data-centric ISS solution - the determination on whether these are manual or automated functions and interfaces is determined by the user and integrators. In each case the PAP packages a PAP-Command containing the user's directive and issues the message to the IEF component using the ISMB.
	Element Type: *Class* ***Owned Operations:*** *Accept_ComponentConfiguration:* The PAP provides features that enable an authorized user to package and issue a PAP-Command message that instructs an IEF component to receive a component configuration included in a PAP-Command message or from a specified location in the IEF protected information store. *Load_ComponentConfiguration:*

Table 5 - Administer Component Configuration Operations	
Element Name	**Operations**
	The PAP provides features that enable an authorized user to package and issue a PAP-Command message that directs an IEF component to load a previously accepted component configuration. *Request_ComponentConfigurations:* The PAP provides features that enable an authorized user to package and issue a PAP-Command message that instructs an IEF component to package its operating configuration and store it to a specified location in the IEF protected information store. *Configure_ComponentProperty:* The PAP provides features that enable an authorized user to package and issue a PAP-Command message that instructs a component to change or alter the value of one or more of its configurable component properties. The component properties determine how each component performs its role within the IEF environment and communicates with supporting services. The PAP provides the interface for an authorized user to issue changes to one or more components, which execute the changes. *Archive_ComponentConfiguration:* The PAP provides features that enable an authorized user to package and issue a PAP-Command message that instructs a component to assemble and package its current configuration and return to the PAP (see PAP-CommandResponse message) or persist it as a file in a SecureAssetContainer within the IEF protected information store. *Connect_ISMBChannel:* The PAP provides features that enable an authorized user to package and issue a PAP-Command message that instructs an IEF component to use an ISMB communication channel.
Administer_Email-PEP-Configuration	PAP component administration features specific to an Email-PEP and Proxy.
	Element Type: *Class* ***Inherited Operations:*** *Accept_ComponentConfiguration* inherited from *Administer_ComponentConfigurations* *Load_ComponentConfiguration* inherited from *Administer_ComponentConfigurations* *Request_ComponentConfigurations* inherited from *Administer_ComponentConfigurations*

Table 5 - Administer Component Configuration Operations	
Element Name	**Operations**
	Configure_ComponentProperty inherited from *Administer_ComponentConfigurations* *Archive_ComponentConfiguration* inherited from *Administer_ComponentConfigurations* *Connect_ISMBChannel* inherited from *Administer_ComponentConfigurations*
Administer_File-PEP-Configuration	PAP component administration features specific to a File-PEP and proxy.
	Element Type: *Class* ***Inherited Operations:*** *Accept_ComponentConfiguration* inherited from *Administer_ComponentConfigurations* *Load_ComponentConfiguration* inherited from *Administer_ComponentConfigurations* *Request_ComponentConfigurations* inherited from *Administer_ComponentConfigurations* *Configure_ComponentProperty* inherited from *Administer_ComponentConfigurations* *Archive_ComponentConfiguration* inherited from *Administer_ComponentConfigurations* *Connect_ISMBChannel* inherited from *Administer_ComponentConfigurations*
Administer_IM-PEP-Configuration	PAP component administration features specific to an IM-PEP and Proxy.
	Element Type: *Class* ***Inherited Operations:*** *Accept_ComponentConfiguration* inherited from *Administer_ComponentConfigurations* *Load_ComponentConfiguration* inherited from *Administer_ComponentConfigurations* *Request_ComponentConfigurations* inherited from *Administer_ComponentConfigurations* *Configure_ComponentProperty* inherited from *Administer_ComponentConfigurations* *Archive_ComponentConfiguration* inherited from *Administer_ComponentConfigurations* *Connect_ISMBChannel* inherited from *Administer_ComponentConfigurations*

Table 5 - Administer Component Configuration Operations

Element Name	Operations
Administer_ISMB-Configuration	PAP component administration features specific to the IEF Secure Messaging Bus (ISMB).
	Element Type: *Class* ***Owned Operations:*** *Gather-IEF_ComponentList:* The PAP provides features that enable an authorized user to package and issue a PAP-Command message that instructs the ISMB to provide a list of components in the environment. (see PAP-CommandResponse Message). *Create_Communication_Channel:* The PAP provides features that enable an authorized user to package and issue a PAP-Command message that instructs the ISMB to create a new communication channel (e.g., Topic or Queue) through which specified IEF Components can communicate. *Create_ISMB-ComponentList:* The PAP provides features that enable an authorized user to package and issue a PAP-Command message that instructs the ISMB controller to generate a list of connected components and return that list to the PAP. (see PAP-CommandResponse) *Compile_ISMB-ChannelList:* The PAP provides features that enable an authorized user to package and issue a PAP-Command message that instructs the ISMB to generate a list of communication channels (topic, queues, other) and provide it to the PAP. ***Inherited Operations:*** *Accept_ComponentConfiguration* inherited from *Administer_ComponentConfigurations* *Load_ComponentConfiguration* inherited from *Administer_ComponentConfigurations* *Request_ComponentConfigurations* inherited from *Administer_ComponentConfigurations* *Configure_ComponentProperty* inherited from *Administer_ComponentConfigurations* *Archive_ComponentConfiguration* inherited from *Administer_ComponentConfigurations* *Connect_ISMBChannel* inherited from *Administer_ComponentConfigurations*

Table 5 - Administer Component Configuration Operations

Element Name	**Operations**
Administer_ISSG-Configuration	PAP component administration features specific to an IEF Security Services Gateway.
	Element Type: *Class* ***Owned Operations:*** *Request_ISSG-Services:* The PAP provides features that enable an authorized user to package and issue a PAP-Command message that instructs the ISSG to generate a list of services available through the gateway and return that list to the PAP (PAP-CommandResponse). ***Inherited Operations:*** *Accept_ComponentConfiguration* inherited from *Administer_ComponentConfigurations* *Load_ComponentConfiguration* inherited from *Administer_ComponentConfigurations* *Request_ComponentConfigurations* inherited from *Administer_ComponentConfigurations* *Configure_ComponentProperty* inherited from *Administer_ComponentConfigurations* *Archive_ComponentConfiguration* inherited from *Administer_ComponentConfigurations* *Connect_ISMBChannel* inherited from *Administer_ComponentConfigurations*
Administer_Messaging-PEP-Configuration	PAP component administration features specific to a Messaging-PEP.
	Element Type: *Class* ***Owned Operations:*** *Activate_MiddlewareConnection:* The PAP provides features that enable an authorized user to package and issue a PAP-Command message that instructs the Messaging-PEP to activate a communication channel in order to share information using the integrated middleware. *Deactivate_MiddleConnection:* The PAP provides features that enable an authorized user to package and issue a PAP-Command message that instructs the Messaging-PEP to deactivate a communication channel in order to share information using the integrated middleware. *Create_MiddlewareConnection:*

Table 5 - Administer Component Configuration Operations

Element Name	**Operations**
	The PAP provides features that enable an authorized user to package and issue a PAP-Command message that instructs the Messaging-PEP to create and configure a connection to the integrated middleware using the configuration parameters enclosed in PAP-Command message or a specified configuration file. *Publish_ChannelDescription:* The PAP provides features that enable an authorized user to package and issue a PAP-Command message that instructs the Messaging-PEP to publish the characteristics of a communication channel to a data or data-discovery registry. ***Inherited Operations:*** *Accept_ComponentConfiguration* inherited from *Administer_ComponentConfigurations* *Load_ComponentConfiguration* inherited from *Administer_ComponentConfigurations* *Request_ComponentConfigurations* inherited from *Administer_ComponentConfigurations* *Configure_ComponentProperty* inherited from *Administer_ComponentConfigurations* *Archive_ComponentConfiguration* inherited from *Administer_ComponentConfigurations* *Connect_ISMBChannel* inherited from *Administer_ComponentConfigurations*
Administer_PDP-Configuration	PAP component administration features specific to a Policy Decision Point (PDP).
	Element Type: *Class* ***Inherited Operations:*** *Accept_ComponentConfiguration* inherited from *Administer_ComponentConfigurations* *Load_ComponentConfiguration* inherited from *Administer_ComponentConfigurations* *Request_ComponentConfigurations* inherited from *Administer_ComponentConfigurations* *Configure_ComponentProperty* inherited from *Administer_ComponentConfigurations* *Archive_ComponentConfiguration* inherited from *Administer_ComponentConfigurations* *Connect_ISMBChannel* inherited from *Administer_ComponentConfigurations*

Table 5 - Administer Component Configuration Operations

Element Name	Operations
Administer_PEP-Configuration	PAP component administration features specific to all Policy Enforcement Points (PEP).
	Element Type: *Class* ***Inherited Operations:*** *Accept_ComponentConfiguration* inherited from *Administer_ComponentConfigurations* *Load_ComponentConfiguration* inherited from *Administer_ComponentConfigurations* *Request_ComponentConfigurations* inherited from *Administer_ComponentConfigurations* *Configure_ComponentProperty* inherited from *Administer_ComponentConfigurations* *Archive_ComponentConfiguration* inherited from *Administer_ComponentConfigurations* *Connect_ISMBChannel* inherited from *Administer_ComponentConfigurations*
Administer_PPS-Configuration	PAP component administration features specific to a Policy-based Packaging and Processing Service (PPS).
	Element Type: *Class* ***Inherited Operations:*** *Accept_ComponentConfiguration* inherited from *Administer_ComponentConfigurations* *Load_ComponentConfiguration* inherited from *Administer_ComponentConfigurations* *Request_ComponentConfigurations* inherited from *Administer_ComponentConfigurations* *Configure_ComponentProperty* inherited from *Administer_ComponentConfigurations* *Archive_ComponentConfiguration* inherited from *Administer_ComponentConfigurations* *Connect_ISMBChannel* inherited from *Administer_ComponentConfigurations*

8.2.3 Administer Component Policy Operations

The following figure further refines the policy management features of the PAP.

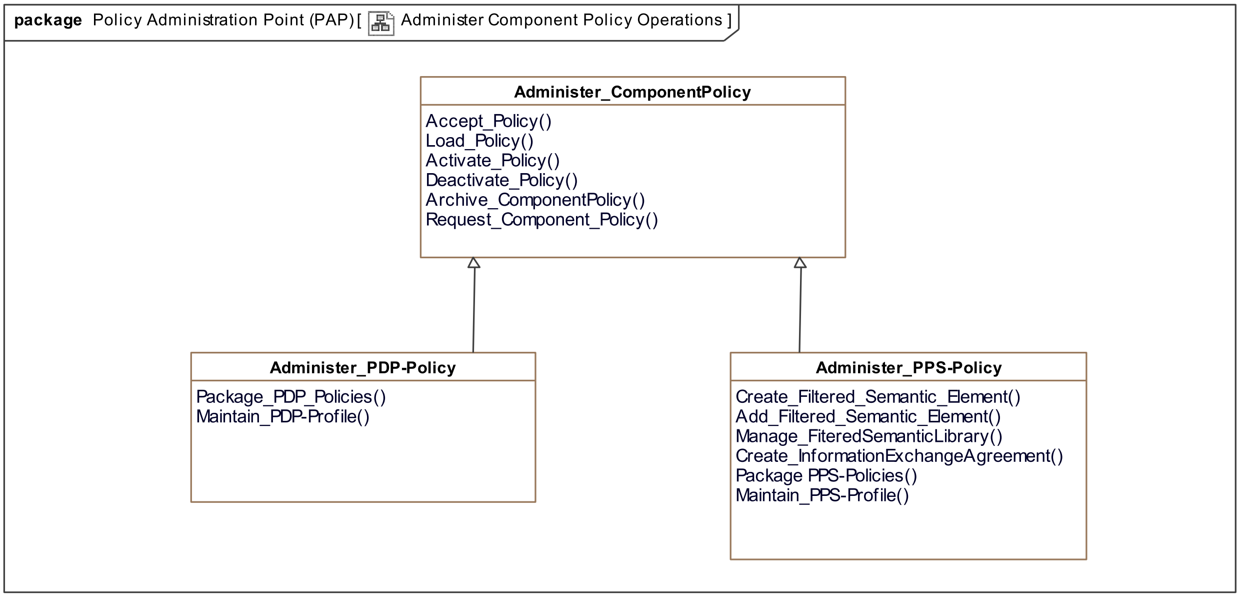

package Policy Administration Point (PAP) [Administer Component Policy Operations]
Administer_ComponentPolicy
Accept_Policy()
Load_Policy()
Activate_Policy()
Deactivate_Policy()
Archive_ComponentPolicy()
Request_Component_Policy()
Administer_PDP-Policy
Package_PDP_Policies()
Maintain_PDP-Profile()
Administer_PPS-Policy
Create_Filtered_Semantic_Element()
Add_Filtered_Semantic_Element()
Manage_FiteredSemanticLibrary()
Create_InformationExchangeAgreement()
Package PPS-Policies()
Maintain_PPS-Profile()

Figure 20 -Administer Component Policy Operations

The following table identifies and describes the elements and operations illustrated in diagram "Administer Component Policy Operations".

Table 6 - Administer Component Policy Operations

Element Name	Operations
Administer_ComponentPolicy	The PAP provides features that enable an authorized user (/administrator) to direct a PDP or PPS to adjust or configure its policy environment. In each case the PAP packages a PAP-Command message containing the user's directives and issues the message to the PDP or PPS to be executed. **Element Type**: *Class* ***Owned Operations:*** *Accept_Policy:* The PAP provides features that enable an authorized user to package and issue a PAP-Command message that instructs an IEF component to receive a policy or policy sets included in a PAP-Command message or from a specified location in the IEF protected information store. The instructions are packaged and issued as a PAP-Command message. *Load_Policy:* The PAP provides features that enable an authorized user to package and issue a PAP-Command message that directs

Table 6 - Administer Component Policy Operations

Element Name	**Operations**
	either a PDP or PPS to load a previously accepted policy or set of policies into its policy environment. *Activate_Policy:* The PAP provides features that enable an authorized user to package and issue a PAP-Command message that instructs a PDP or PPS to activate one or more of its policies previously loaded into its policy environment. *Deactivate_Policy:* The PAP provides features that enable an authorized user to package and issue a PAP-Command message that instructs a PDP or PPS to deactivate one or more of its policies that was previously loaded into its policy environment. *Archive_ComponentPolicy:* The PAP provides features that enable an authorized user to package and issue a PAP-Command message to instruct a PDP or PPS to package its current policy environment and persist it to a specified location in the IEF environment or return it to the PAP as part of a PAP-CommandResponse Message. *Request_Component_Policy:* The PAP provides features that enable an authorized user to package and issue a PAP-Command message that instructs a PPS, or PDP to package one or more policies and return them as part of a PAP-CommandResponse message.
Administer_PDP-Policy	PAP features that enable an authorized user to administer the configurations and policies for one or more PDPs operating within the PAPs designated environment. The features also include a set of features that enable the user to manage the access and release of control policies employed by the PDP.
	Element Type: *Class* ***Owned Operations:*** *Package_PDP_Policies:* PAP features that gather a specified set of PDP- policies (access and release control) selected by a user (IEF Administrator) or by PAP scheduling features (optional) and packages the policies for release, and issues them to the PDP. *Maintain_PDP-Profile:* The PAP features that enable a user (/administrator) to administer profiles for each of the PDPs in the environment. A PDP profile includes: PDP identifier; PDP name; PDP Role(s); Controlled PEPs; PPS active policy set; sets of authorized policies; Sets of authorized operating

Table 6 - Administer Component Policy Operations

Element Name	**Operations**
	(configuration) parameters; archived policy sets; and PDP status and events (alerts/warnings). ***Inherited Operations:*** *Accept_Policy* inherited from *Administer_ComponentPolicy* *Load_Policy* inherited from *Administer_ComponentPolicy* *Activate_Policy* inherited from *Administer_ComponentPolicy* *Deactivate_Policy* inherited from *Administer_ComponentPolicy* *Archive_ComponentPolicy* inherited from *Administer_ComponentPolicy* *Request_Component_Policy* inherited from *Administer_ComponentPolicy*
Administer_PPS-Policy	PAP features that enable an authorized user to administer (Create, Add, Modify, Delete, Activate, Deactivate, and Load) policies for one or more PPSs in the environment. User executed changes are issued to the specified PPS using a PAP-Command Message.
	Element Type: *Class* ***Owned Operations:*** *Create_Filtered_Semantic_Element:* The PAP provides features that enable an authorized user to select an IEPPV FilteredSemanticElement pattern, and set the filters to accommodate the specific operational/mission needs or requirements to responsibly share information with the targeted user/community. *Add_Filtered_Semantic_Element:* The PAP provides features that enable an authorized user to add a configured FilteredSemanticElement to a specified InformationExchange_Specification (/contract). *Manage_FiteredSemanticLibrary:* The PAP provides features that enable an authorized user to acquire, store, sort and access pre-configured collections of FilteredSemanticElements. *Create_InformationExchangeAgreement:* The PAP provides features that enable an authorized user to create a new InformationExchangeSpecification. As part of this process the user specifies: • Session Information;

Table 6 - Administer Component Policy Operations	
Element Name	**Operations**
	- Channel Information; - Message and Network Protocols to be applied; - Security, Privacy and sensitivity restrictions; and - Information Elements to be enabled. *Package PPS-Policies:* The PAP provides features that collect a specified set of PPS- policies (data packaging and processing) selected by a user (IEF Administrator) or by PAP features (optional) that schedule policy changes, format the policies for release, and issue them to the PDP. *Maintain_PPS-Profile:* The PAP provides features that enable a user (/administrator) to administer profiles for each of the PPSs in the environment. A PDP profile includes: PPS identifier; PPS Name; PPS Role(s); Dissemination services and enabling PEPs; active policy set; sets of authorized policies; Sets of authorized operating (configuration) parameters; archived policy sets; and PPS status and events (alerts/warnings). ***Inherited Operations:*** *Accept_Policy* inherited from *Administer_ComponentPolicy* *Load_Policy* inherited from *Administer_ComponentPolicy* *Activate_Policy* inherited from *Administer_ComponentPolicy* *Deactivate_Policy* inherited from *Administer_ComponentPolicy* *Archive_ComponentPolicy* inherited from *Administer_ComponentPolicy* *Request_Component_Policy* inherited from *Administer_ComponentPolicy*

9 Policy Decision Point

The Policy Decision Point (PDP) integrates local (user specified) security services and provides the IEF with the ability to access the user (e.g., identity, credentials, and policy-rights(/attributes)), system (e.g., date, and time) and operational data needed to authorize a users' actions or access, to the release of information to a specified communication channel. The PDP defines both the integration services and interfaces needed to exchange information requested by the IEF authorization and access controls. PDP data interfaces services include:

- Security services, including:
  - Identity Management (IdM) services;
  - Authentication services;
  - Credential Management services; and
  - Attribute Management services.
- Decision management services in the PDP, including:
  - Rules or Inference Services; and
  - Rules Management services.
- Tamper Resistant Logging Services; and
- Alerting and Warning Services.

9.1 Policy Decision Point (PDP)

The Policy Decision Point (PDP) is responsible for bridging to or providing the IEF policy adjudication services. The PDP policy decisions are based on:

- The authorizations / attributes of the user initiating the request;
- The sensitivities / attributes of the resource being requested or acted upon;
- The specified action(s) in the request; and
- (Optional) operational context.

The PEPs, and the PAP leverage the PDP to adjudicate access and release controls. The PEP formulates a policy decision query and submits it to the PDP. Depending on the response from the PDP, the PEP allows or prevents the transaction from taking place. Examples of PEP to PDP policy decision requests would include:

- Is the user authorized to download a document from an IEF protected Information Store?
- Is the user authorized to send the labeled attachment in an email message?
- Is the user authorized to join an IEF protected chat room?
- Is the user allowed to release the message to the specified communication channel?

The generalized decision process follows the following steps:

- Each user information sharing action is intercepted by the PEP;
- The PEP extracts the metadata from the information elements in the exchange;
- The PEP reformulates a decision request and sends it to the PDP using the ISMB;
- The PDP receives the request and retrieves applicable security policy rules from the policy rule set. Interpreting the user's action in the context of the security policy, the PDP returns a policy decision (e.g.,

Permit or Deny). Permitted actions are allowed to proceed to the data service; denied actions are returned to the endpoint with an appropriate error message.

PEPs format their policy decision queries using XACML and, in turn, expect a response message that is formatted using the XACML protocol. The IEF policies are stored in a local data store and accessed via the policy-store interface. The PAP provides the interface to allow the creation, modification, and removal of policies from the policy-store.

9.1.1 PDP Component Operations

The following figure identified the core features and functions of an IEF Policy Decision Point (PDP). The PDP is a processing feature that interprets a policy request and adjudicates the request against the current security access and release policies. The security policy is an expression of the access control rules for the specific security domain and operation. The IEF PDP relies on the XACML as a foundational specification.

The PDP adjudicates access to, or the release of resources to a specified user based on:

- The sensitivity (privacy, confidentiality, classification, or legal significance) of the resource:
    - InformationElement;
    - DataElement;
    - Topics;
    - Queues;
    - Platform;
    - System;
    - Device;
    - Folder;
    - Application;
    - Service (e.g., IEF Component);
    - Communication Channel;
    - Session; and/or
    - Networks.
- The privileges of the user to receive, access, release or process the constituent information.
- (Optional) The location (physical or network) of the recipient of the information.
- (Optional) The operational context in which the decision is being made (e.g.: phase, threat, operational roles (recipient and sender)) – context provisioned by the user situational awareness system via the ISSG and PAP, or user entered through the PAP.
- Users may include:
    - Individual (/persons);
    - Organizations;
    - Role;
    - Community;
    - Topics;
    - Queues;

- Platform;
- Systems;
- Device;
- Applications;
- Services (e.g., IEF Component);
- Communication Channel;
- Session; and
- Networks.

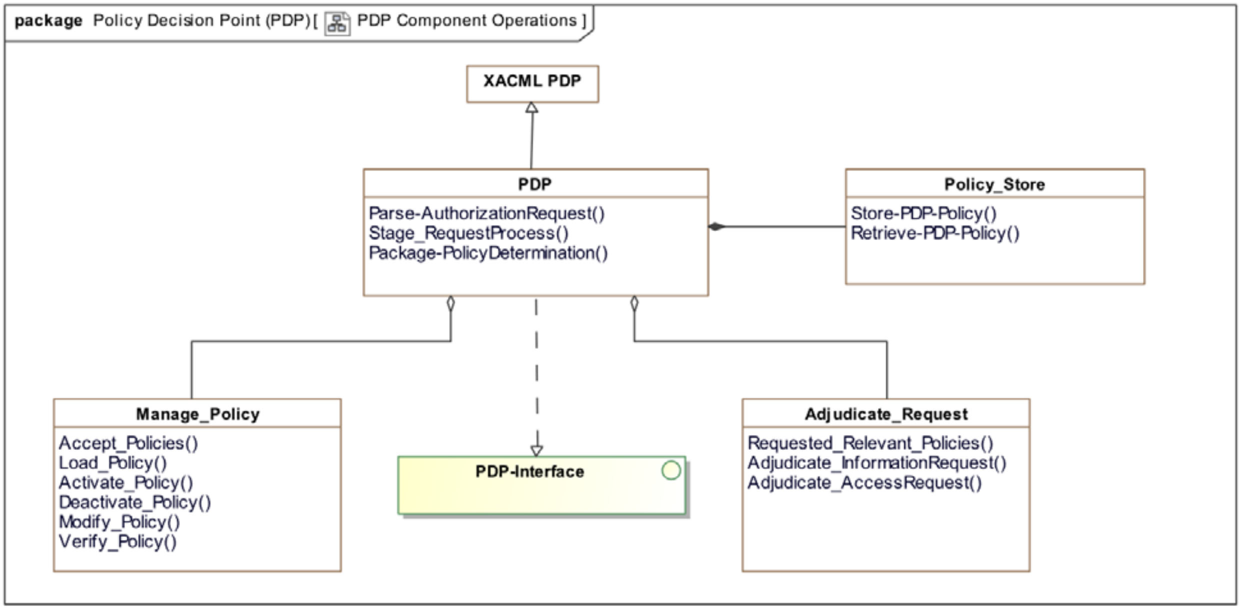


Figure 21 -PDP Component Operations

The following table identifies and describes the elements and operations illustrated in diagram “PDP Component Operations”.

Table 7 - PDP Component Operations

Element Name	Operations
Adjudicate_Request	PDP features that combine to adjudicate a request by an IEF component to authorize: • Release InformationElements to a specified user;

Table 7 - PDP Component Operations

Element Name	**Operations**
	• Access to a specified information element by a specified user; • Execution of specified functions/operations by a specified user or IEF component.
	Element Type: *Class* ***Owned Operations:*** *Requested_Relevant_Policies:* The PDP provides features that retrieve policies appropriate for the adjudication of the specific request. The policies are retrieved from the PDP policy store. *Adjudicate_InformationRequest:* The PDP provides features that validate and verify that a specific user (recipient) is authorized (has the privileges or policy-rights) to release the specified information content to the specified recipients using the specified communication channels. Authorization is based on the users’ explicit privileges, the markings (e.g., classifications and restrictions, and warnings) bound to each information element and PDP policies. If the user's rights are verified, the PDP issues instructions to the PEP to grant access to the user. Individual authorization criteria can be defined for: Files; Email and attachments; Instant messaging and attachments, Structured Messaging; and Web Access. *Adjudicate_AccessRequest:* PDP features that validate and verify that a specific user is authorized to perform the specified function/operation. As the only user interface to the IEF components is through the PAP, the PAP has many of the features of the PEP in addition to the required management and administration tools. The PAP authorizes the administrator logged onto the PAP to perform functions such as: • Receive (/Access) the specified InformationElements (e.g., policies, logs and component configurations); • Send (/Release) the specified information element; • Configure IEF element (e.g., PAP, PEP, PDP, CTS, TLS, or ISSG); • Configure IEF Policies (Data or access control); • Access IEF resources; or • Access IEF administration data. It is anticipated that there may be several categories of administrators, each possessing different privileges,

Table 7 - PDP Component Operations	
Element Name	**Operations**
	authorizations and capabilities. The PAP must allocate access accordingly.
Manage_Policy	PDP features that execute policy management commands from an authorized PAP. PAP policy management commands include: - Accept_Policy; - Activate_Policy; - Deactivate_Policy; - Load_Policy; and - Modify_Policy.
	Element Type: *Class* ***Owned Operations:*** *Accept_Policies:* The PDP provides features that respond to a PAP-Command message directing it to receive a policy or set of policies contained in the message or at a specified location in the IEF Protected Information Store. *Load_Policy:* The PDP provides features that respond to a PAP-Command message directing it to load a policy or set of policies to its operational policy store. *Activate_Policy:* The PDP provides features that respond to a PAP-Command message directing it to activate (turn-on) a policy or set of policies in its operational policy store. A policy must be instantiated in the operational policy store, and in and inactive state at the time of the command, to be activated - otherwise a warning message is issued. *Deactivate_Policy:* The PDP provides features that respond to a PAP-Command message directing it to deactivate (turn-off) a policy or set of policies in its operational policy store. A policy must be instantiated in the operational policy store, and in an active state at the time of the command, to be deactivated - otherwise a warning message is issued. *Modify_Policy:*

Table 7 - PDP Component Operations

Element Name	**Operations**
	The PDP provides features that respond to a PAP-Command message directing it to accept modifications to and existing policy in its environment. These modifications either alter or replace policies in the Policy Store. *Verify_Policy:* IEF Component (PDP & PPS) features that validate and verify a set of policies, confirming that the policies are from a valid source and conform to the IEF Specifications.
PDP	The Policy Decision Point adjudicates access to, or the release of resources to a specified user. based on: • The sensitivity of the resource; • The privileges of the user; and • (Optional) operational context in which the decision is being made.
	Element Type: *Class* ***Owned Operations:*** *Parse-AuthorizationRequest:* The PDP provides features that gather data from a PDP-AuthorizationRequest message and provides the data elements for the adjudication process. *Stage_RequestProcess:* The PDP provides features that assess the request and stage the adjudication process of each authorization decision. The authorization request many contain multiple elements, e.g.: • Multiple InformationElements (e.g., message containing a digest, packages, payloads attachments; or an email with attachments) with multiple recipients would require: ○ The Sender to be authorized to release each individual InformationElement; ○ Each Recipient to be authorized to have access to each InformationElement. • Administrator requires access to multiple IEF resources (/components) to perform the required process. The Administrator must be evaluated as having access rights to each resource and to perform the function (/operation). • IEF component requesting resources through the ISSG.

Table 7 - PDP Component Operations

Element Name	**Operations**
	Policies may change through the course of an operation, resulting in differing determinations depending on context (e.g., phase, threat, role and responsibilities) and a determination of which policies apply. *Package-PolicyDetermination:* Interface features that gather the DataElements comprising the PolicyDetermination and package it for release to the PEP or PAP to be enforced. ***Inherited Operations:*** *Start_Operations* inherited from *IEF_Component* *Maintain-OperatingState* inherited from *IEF_Component* *Recover_Operations* inherited from *IEF_Component* *Track_RequestResponse* inherited from *IEF_Component* *Authorize_ActionRequest* inherited from *IEF_Component* *Package_AuthorizationRequest* inherited from *IEF_Component* *Package-AdministrativeCommandResponse* inherited from *IEF_Component* *Package_EventLog* inherited from *IEF_Component* *Package_AlertWarningData* inherited from *IEF_Component* *Process_AdministrationCommand* inherited from *IEF_Component* *Configure_Properties* inherited from *IEF_Component* *Archive_Properties* inherited from *IEF_Component*
Policy_Store	PDP features that interact with the PDP policy store. These features enable the PDP to store and retrieve access and release control policies.
	Element Type: *Class* ***Owned Operations:*** *Store-PDP-Policy:* PDP features that transfer a set of policies to its active and persistent policy store. *Retrieve-PDP-Policy:* PDP features that transfer a set of policies from the persistent policy store to active memory.

Table 7 - PDP Component Operations

Element Name	**Operations**
XACML PDP	The IEF PDP was derived from the core concepts defined by the XACML specification.
	Element Type: *Class*

10 Policy Enforcement Points

The Policy Enforcement Point (PEP) intercepts a user request to a resource and enforces the decision from the Policy Decision Point (PDP). The PEP queries the PDP to see if the user has access to the resource (in this case data and information resources), and depending on the PDP response, allows the filter to pass or fail. There are five (5) Policy Enforcement Points (PEP) defined by the IEF Reference Architecture. The following clauses define the information safeguards comprising each of the PEPs. For each PEP the Reference Architecture provides a set of Use Case, Abstract Services Classes, Interfaces and Operational Sequences diagrams describing the interface layer for off-the-shelf services, i.e.:

- Receiver Directed Messaging PEP;
- Session Directed Messaging PEP; and
- Specialized PEPs:
    - Email;
    - File Share; and
    - Instant Messaging.

10.1 PEP Component Operations

The following figure identifies the Policy Enforcement Point (PEP) features that constitute part of each of the four type of PEP:

1. Email-PEP (Clause 10.2);
2. File-Share-PEP (Clause 10.3);
3. Instant Messaging /Chat-room (IM-PEP) (Clause 10.4); and
4. Messaging-PEP (Clause 10.5).

The Messaging-PEP offers two configurations:

1. The first, the data provisioning service or PPS into the ISSG; and
2. The second, uses a proxy configuration similar to the email, file-share and IM PEPs.

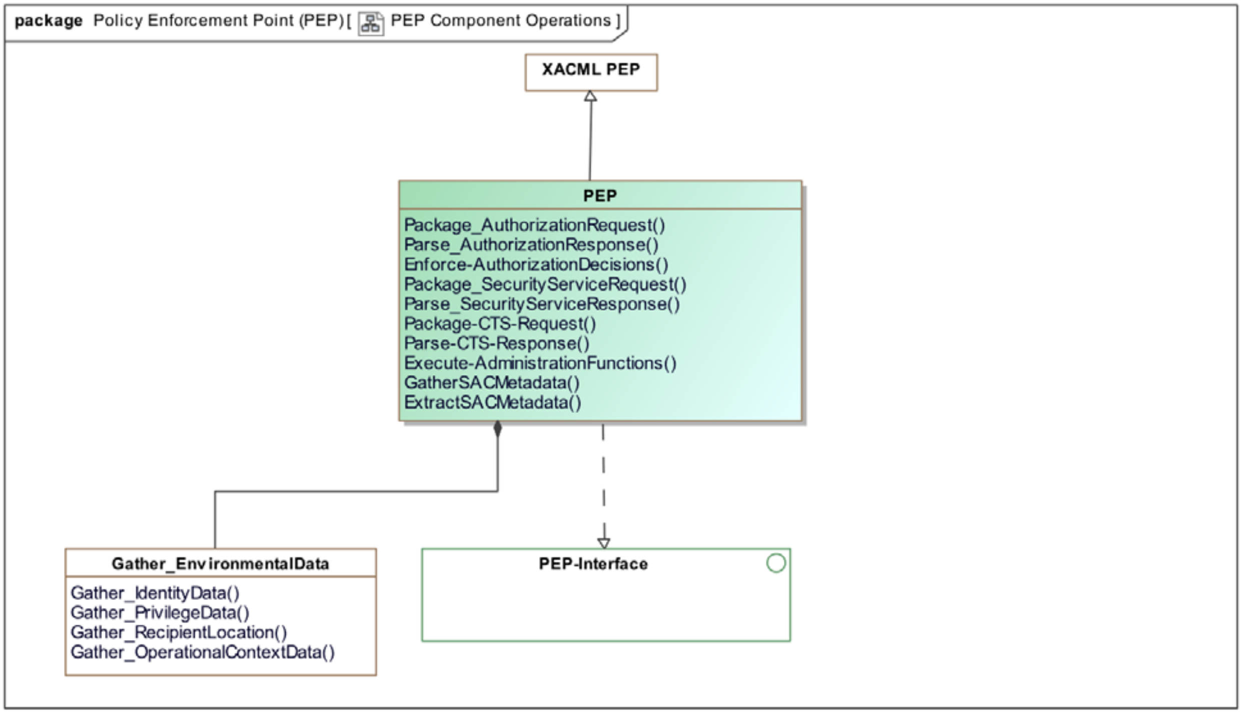

package Policy Enforcement Point (PEP) [PEP Component Operations]
XACML PEP
PEP
Package_AuthorizationRequest()
Parse_AuthorizationResponse()
Enforce-AuthorizationDecisions()
Package_SecurityServiceRequest()
Parse_SecurityServiceResponse()
Package-CTS-Request()
Parse-CTS-Response()
Execute-AdministrationFunctions()
GatherSACMetadata()
ExtractSACMetadata()
Gather_EnvironmentalData
Gather_IdentityData()
Gather_PrivilegeData()
Gather_RecipientLocation()
Gather_OperationalContextData()
PEP-Interface

Figure 22 -PEP Component Operations

The following table identifies and describes the elements and operations illustrated in diagram "PEP Component Operations".

Table 8 - PEP Component Operations	
Element Name	**Operations**
Gather_EnvironmentalData	Collection of PEP features that gather information that may be used by the PDP to adjudicate a user's request to access or release information elements. These features are used if the user's policies are applied differently if the operational context changes, e.g., • Changes in threat level; • Users (e.g., organization) role and responsibilities; and • Operational phase.
	Element Type: *Class* ***Owned Operations:*** *Gather_IdentityData:* The PEP provides features that gather the Identity Information for the sender and/or recipients of the specified InformationElement(s). These features provide the ability to

Table 8 - PEP Component Operations	
Element Name	**Operations**
	request this information from the users' infrastructure services that provide identity management. All requests to the users' specified infrastructure are issued through the Security Services Gateway using an ISSG-Request message. *Gather_PrivilegeData:* The PEP provides features that gather the sender and/or recipient privileges (authorizations or attributes) to access and use specific classifications of information, and communication channels. Requests to the users' specified infrastructure are issued through the Security Services Gateway using an ISSG-Request message. *Gather_RecipientLocation:* (Optional) The PEP provides features that gather information about the recipients' location (physical, or electronic). A recipients' location(s) (e.g., network location, physical location, and device) may impact the user's authorizations and the information content they are authorized to receive. These features may only be available if the IEF has the ability to request the information from the users' situational awareness, incident management or network management systems. These services must be integrated to the Security Services Gateway, and the specific information requested integrated into the internal communications between the PEP and ISSG. Requests to the users' specified infrastructure are issued through the Security Services Gateway using an ISSG-Request message. *Gather_OperationalContextData:* (Optional) The PEP provides features that gather (request) information about the current situation or operational context. These features require the ability to communicate with the user's incident management or situational awareness systems. Information that may influence the access to or the release of information includes the users: • Role and Responsibility; • Phase of the operation; • Operational Threat; • Operational Risk; • Command Intent; • Physical location; • Available communications links; and • Access device.

Table 8 - PEP Component Operations

Element Name	Operations
	Each of these contextual descriptions may affect the handling of policy related to sensitive information and/or the quality of service available to the recipients. This in turn may have an impact on the user's authorizations and the information content they are authorized to receive. Operational context may be integrated into the message to the PDP, which, must incorporate into the PDP policy environment.
PEP	A Policy Enforcement Point (PEP) intercepts each InformationElement transiting between a user client application and the server (Email, Instant Messaging and File Share, and Data) to ensure the requesting user is authorized to perform the requested action on the specified information element(s). The PEP requires that each file is bound with the security, privacy and other user markings required by the PDP to decide if the feature (action) can be performed. There are four (4) specializations for the PEP, including: 1. The File-PEP; 2. The Email-PEP; 3. The IM-PEP; 4. The Messaging-PEP. Each PEP provides features that orchestrate the authorization or rejection of a user access or release request. The PEP packages a PDP-Request Message for each information element in the specified exchange: • Email: Email Body and each attachment; • File: Each individual File; • Instant Message: Each Message and Each Attachment; • Structured Message: Message, the Digest, each Information Package, each Payload, and each Attachment. Each PEP provides the ability to disassemble an exchange, gather the metadata for each InformationElement, gather authorization data, request authorizations, request cryptographic transformation and reassemble the message for release. The PEP may have the ability to redact information elements based on release instructions from the PDP.
	Element Type: *Class* ***Owned Operations:*** *Package_AuthorizationRequest:* Each PEP provides features that gather data and metadata elements required to adjudicate and determine whether or not

Table 8 - PEP Component Operations

Element Name	**Operations**
	an information element is releasable by the sender, to the specified recipients, using the specified communication channel. The PEP also provides the features needed to package a PDP-Request message and issue it to a PDP for adjudication. *Parse_AuthorizationResponse:* The PEP provides features that parse a PDP-Response message and extract the data necessary for other PEP features to enforce the PDP determinations. *Enforce-AuthorizationDecisions:* The PEP provides features that ingest the PDP's determination and instructions (e.g., redact one or more information/data elements) and constraints. The PEP then prepares and packages the information element(s) that are authorized for release to conform to the specified instructions and constraints. If the release is denied, the PEP terminates the processing. *Package_SecurityServiceRequest:* The PEP provides features that package Service Request Data as an ISMB-Message directed to ISSG in order to request information from the user's security infrastructure, e.g.: • To request Sender Identity Information from the User specified Identity Management Services; • To request Receiver Identity Information from the User specified Identity Management Services; • To request Sender Privileges from the User specified Identity and Access Management Services; • To request Receiver Privileges from the User specified Identity and Access Management Services; and • Request Cryptographic Keys from the user specified Key Management system or service. *Parse_SecurityServiceResponse:* The PEP provides features that parse an ISSG-Response message and extract the data elements required by the PEP enforcement features. Service Response Data includes: • CryptographicKey and KeyToken; • Sender Identity Information; • Receiver Identity Information; • Sender Privileges;

Table 8 - PEP Component Operations

Element Name	**Operations**
	- Receiver Privileges; and - Operational Context Data. *Package-CTS-Request:* The PEP provides features that package the cryptographic keys and information elements as a CTS-Request for the CTS to encrypt or decrypt the information element. *Parse-CTS-Response:* The PEP provides features that parse a CTS-Response message and extracts the transformed InformationElement. *Execute-AdministrationFunctions:* PEP features that execute AdministrativeCommands from an authorized Policy Administration Point. PEP administrative functions include: - Activate PEP features; - Deactivate PEP features; - Configure PEP Parameters; - Archive PEP Operational Environment; and - Publish PEP Configuration. *GatherSACMetadata:* The PEP provides features that extract metadata from a Secure Access Container (SAC) in order to determine whether or not the information element has been tampered with, and package an authorization request to the PDP. *ExtractSACMetadata:* The PEP provides features that extract the metadata from the SAC header that is needed to state the authorization of access to or release of an IEF protected file. ***Inherited Operations:*** *Start_Operations* inherited from *IEF_Component* *Maintain-OperatingState* inherited from *IEF_Component* *Recover_Operations* inherited from *IEF_Component* *Track_RequestResponse* inherited from *IEF_Component* *Authorize_ActionRequest* inherited from *IEF_Component* *Package_AuthorizationRequest* inherited from *IEF_Component* *Package-AdministrativeCommandResponse* inherited from *IEF_Component*

Table 8 - PEP Component Operations	
Element Name	**Operations**
	Package_EventLog inherited from *IEF_Component* *Package_AlertWarningData* inherited from *IEF_Component* *Process_AdministrationCommand* inherited from *IEF_Component* *Configure_Properties* inherited from *IEF_Component* *Archive_Properties* inherited from *IEF_Component*
XACML PEP	The IEF PEP was derived from the core concepts defined by the XACML specification.
	Element Type: *Class*

10.2 Email PEP

An IEF Email PEP intercepts each email message as it transits between the email client and the email server. The PEP verifies that:

- The content (message body and all attachments) is encrypted while at rest and in transit;
- The sender of the email has the policy-rights to send the specific email content; and
- Each recipient (To, cc, and bcc) is individually authorized (has the policy-rights) to receive and access the content of the email body and each of the attachments.
- Upon receipt, the policy-rights are again verified prior to any decryption of the email's content.

The Email PEP applies protections to each element (email body and all attachment) included in the email message. Each element must therefore be bound with all relevant sensitivity markings (e.g., privacy, confidentiality, classification, legal significance), and caveats (warning orders and restrictions) necessary for the IEF services to determine the policy defined protections appropriate to the emails content.

10.2.1 Email_PEP Component Operations

The following figure identifies the key features of the Email PEP. The Email PEP intercepts each email transiting between a User Email Client and the Email Server. The PEP validates and verifies: that each participant is authorized to send or receive the information contained in the email; each information element (email body and attachments) is appropriately marked; and each information element is appropriately protected. The Email PEP is implemented as a proxy architecture where e-mail traffic is directed through the proxies to the services that applied the information protection logic.

When a user sends a request to the Mail server to retrieve new e-mail, the PEP:

- Intercepts the request and verifies that the user is authorized to access the server;
- If the user is authorized to access the server, the request is relayed to the server. The PEP then intercepts each mail message for the user retrieved from the server and verifies the user is authorized to receive each individual mail message and each of the included attachments. Performing this action for each mail message returned from the server, the PEP:
    - Disassembles each email message;
    - Determines if the user is authorized to access the contents of the mail message and each of the attachments;
    - If the user is authorized to see the contents:
        - Decrypts the contents of each element;
        - Repackages the mail; and
        - Relays it to the mail message client.
    - If the user is not authorized to receive one or more information elements:
        - Discards the offending element;
        - Decrypts the authorized elements;
        - Repackages the email with only the authorized elements; and
        - Relays it to the mail message client.
    - Reports the results of the transaction to the TLS
    - If required by policy, alerts the administrator to each issue with received email.

When a user sends an mail message to the Mail server, the PEP:

- Intercepts the mail message from the mail client;
- Disassembles each mail message to extract the embedded information elements (body and attachments);
- Verifies the user is authorized to release the content of information elements embedded in the email. If the user is authorized:
    - Verifies that each recipient is authorized to receive the content in the message. If each recipient is authorized, the PEP:
        - Encrypts the contents of each element;
        - Repackages the mail message with the encrypted versions of the body and attachments; and
        - Relays it to the mail message client.
    - If a Recipient is not authorized to receive one or more information elements, return the message to the sender with an error message identifying the policy breach so the sender can address the issue (*Note: automated redaction [removing the offending elements may also be offered*].
    - If a Sender is not authorized to send one or more information elements, returns the message to the sender with an error message identifying the policy breach so the sender can address the issue (*Note: automated redaction [removing the offending elements may also be offered*].
- Reports the results of the transaction to the TLS.

- If required by policy, alerts the administrator to each issue with sent email.

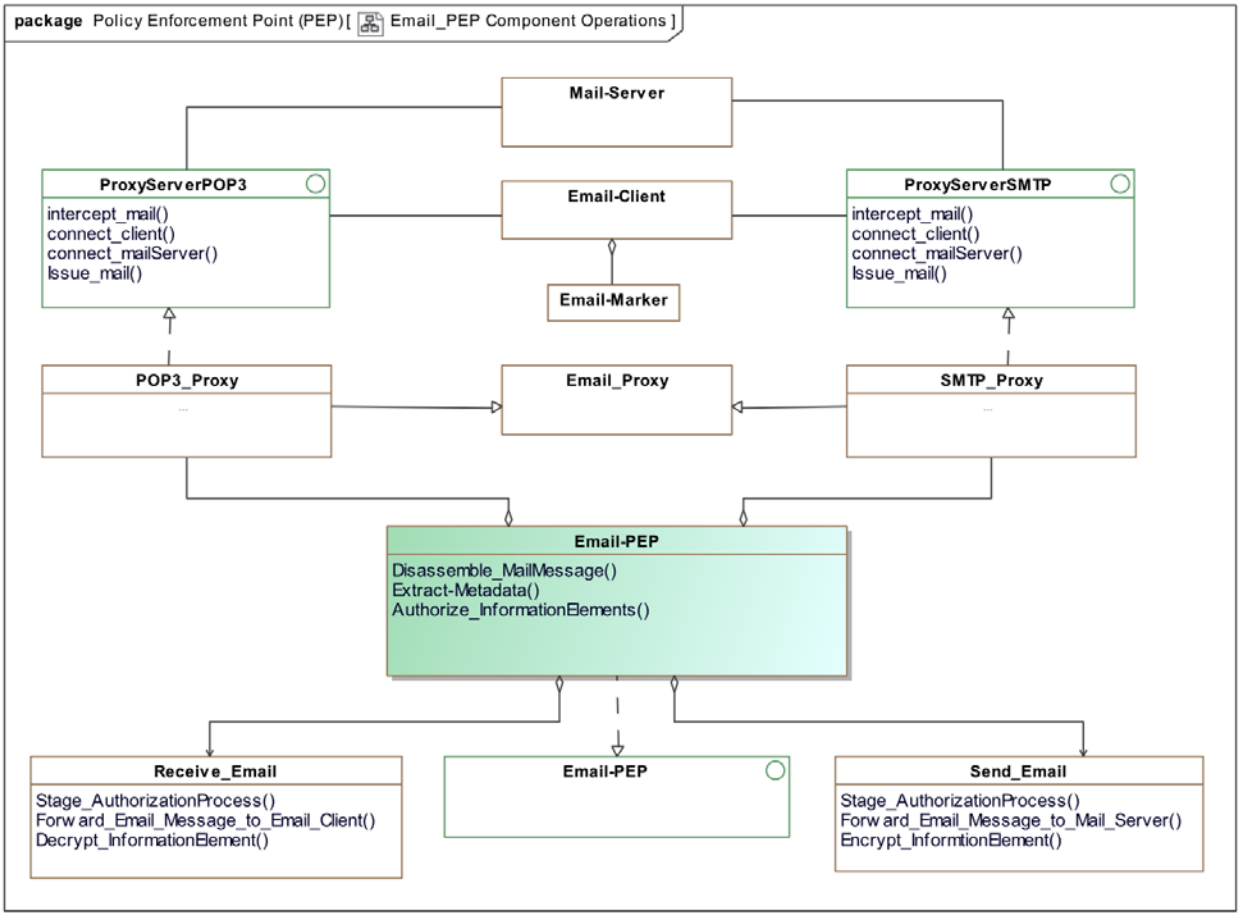

package Policy Enforcement Point (PEP) [Email_PEP Component Operations]
Mail-Server
ProxyServerPOP3
intercept_mail()
connect_client()
connect_mailServer()
Issue_mail()
Email-Client
ProxyServerSMTP
intercept_mail()
connect_client()
connect_mailServer()
Issue_mail()
Email-Marker
POP3_Proxy
Email_Proxy
SMTP_Proxy
Email-PEP
Disassemble_MailMessage()
Extract-Metadata()
Authorize_InformationElements()
Receive_Email
Stage_AuthorizationProcess()
Forward_Email_Message_to_Email_Client()
Decrypt_InformationElement()
Email-PEP
Send_Email
Stage_AuthorizationProcess()
Forward_Email_Message_to_Mail_Server()
Encrypt_InformtionElement()

Figure 23 -Email_PEP Component Operations

The following table identifies and describes the elements and operations illustrated in diagram "Email_PEP Component Operations".

Table 9 - Email_PEP Component Operations

Element Name	Operations
Email-Client	The email client is a standard commercial of the shelf application, selected by the user, for accessing and generating e-mail messages. The user is responsible for assuring that email messages and their attachments are effectively marked with their security level and caveats.

Table 9 - Email_PEP Component Operations	
Element Name	**Operations**
	Element Type: *Class*
Email-Marker	The Email-Marker is a software service that integrates with off-the-shelf Email clients that obliges and prompts the authorized user to appropriately mark the e-mail and if not already marked, its attachments. The IEF requires that all elements of the email are appropriately marked (Tagged or labeled) in order to apply access and release policies.
	Element Type: *Class*
Email-PEP	The E-mail PEP operates between the e-mail client and the Mail server. The PEP intercepts each request, determines if the user is authorized to perform the requested action given sensitivities of the information assets (email body and each attachment) involved.
	Element Type: *Class* ***Owned Operations:*** *Disassemble_MailMessage:* The Email-PEP provides features that disassemble a mail message and extracts each of the enclosed information elements (body and attachment). *Extract-Metadata:* The PEP provides features that gather *Metadata* from the email header and each of the attachments contained in the Email message. The data collected includes: • InformationElement(s) metadata; and • User (Producer & recipient) Identifier (Email Address). *Authorize_InformationElements:* The Email-PEP provides features that communicate with the PDP to determine if the requested action is authorized. As part of this operation the PEP: • gathers and assembles the data elements needed by the PDP to adjudicate the releaseability of the information elements and render a decision; • Package a PDP-Request message; • Issues the PDP-Request to the PDP; • Receives the PDP-Response message.

Table 9 - Email_PEP Component Operations	
Element Name	**Operations**
	Inherited Operations: *Package_AuthorizationRequest* inherited from *PEP* *Parse_AuthorizationResponse* inherited from *PEP* *Enforce-AuthorizationDecisions* inherited from *PEP* *Package_SecurityServiceRequest* inherited from *PEP* *Parse_SecurityServiceResponse* inherited from *PEP* *Package-CTS-Request* inherited from *PEP* *Parse-CTS-Response* inherited from *PEP* *Execute-AdministrationFunctions* inherited from *PEP* *GatherSACMetadata* inherited from *PEP* *ExtractSACMetadata* inherited from *PEP* *Start_Operations* inherited from *IEF_Component* *Maintain-OperatingState* inherited from *IEF_Component* *Recover_Operations* inherited from *IEF_Component* *Track_RequestResponse* inherited from *IEF_Component* *Authorize_ActionRequest* inherited from *IEF_Component* *Package_AuthorizationRequest* inherited from *IEF_Component* *Package-AdministrativeCommandResponse* inherited from *IEF_Component* *Package_EventLog* inherited from *IEF_Component* *Package_AlertWarningData* inherited from *IEF_Component* *Process_AdministrationCommand* inherited from *IEF_Component* *Configure_Properties* inherited from *IEF_Component* *Archive_Properties* inherited from *IEF_Component*
Email_Proxy	The Email proxy server acts as an intermediary for requests from clients seeking resources from other servers. The Proxy server intercepts each message and assures that the PEP is engaged to validate and verify that: 1. The sender is authorized to release the included content (body and each attachments) using email; and 2. The recipients are authorized to receive the included content.

Table 9 - Email_PEP Component Operations

Element Name	**Operations**
	Element Type: *Class*
Mail-Server	A mail server is a computer application that serves as an electronic post office for e-mail. The email application is built around agreed-upon, standardized protocols for handling mail messages and any data files (such as images, multimedia or documents) that might be attached to them. Email servers provide features such as: • Simple Mail Transfer Protocol (SMTP) is an Internet standard for e-mail transmission (i.e., RFC 5321); • Post Office Protocol (POP) is an application-layer Internet standard protocol used by local e-mail clients to retrieve e-mail from a remote server over a TCP/IP. POP3 is the current version; • Internet Message Access Protocol (IMAP) is a protocol for e-mail retrieval and storage; • Webmail (or web-based e-mail) is any e-mail client implemented as a web application running on a web server. The web server also provides for the storage of the e-mail using a database, file system or other form of storage. The IEF services do not alter the standard e-mail headers or protocols, however, the content (body and attachments) of the e-mail message are encrypted. The IEF ensures that all messages are only provided to authorized recipients, appropriately marked, and encrypted; both at rest and in motion.
	Element Type: *Class*
POP3_Proxy	The POP3 proxy is an intermediary service connecting to the mail server. The e-mail client connects to the proxy server, which requests service from the mail server. The proxy service adds the IEF structures and encapsulates the mail server. The proxy service adds the structures that enable the IEF to: • Intercept user requests to receive their e-mails from the server. Before the proxy delivers an e-mail message it hands off the e-mail to a staging area where the PEP can apply the necessary information protection logic. The security markings from the encrypted messages are

Table 9 - Email_PEP Component Operations

Element Name	Operations
	extracted and used to determine if the security policy allows the user to receive the e-mail message. If the security policy allows the user to receive the email message it is then decrypted and forwarded to the user's email client. - Intercept user requests to send an email to the server. Before the proxy sends the mail message it hands off the email to a staging area where the PEP can again apply the necessary information protection logic. The security attributes from the e-mail message are extracted and used to determine if the sender can send the email message. If the security policy allows the user to send the email message it is then encrypted and forwarded to the mail server for delivery.
	Element Type: *Class*
Receive_Email	Email-PEP features that process email messages transiting from the email server to the User's email client application.
	Element Type: *Class* ***Owned Operations:*** *Stage_AuthorizationProcess:* The Email-PEP provides features that queue each received mail message for authorization and decryption. As part of this process, the PEP: 1. Parses the mail message and extracts the enclosed information elements (body and Attachment(s)); 2. Verifies the mail message is transiting to the intended recipient - if not alert the administrator (see PAP-AlertWarning message) and terminate processing of the message. 3. Collects the metadata describing the sensitivity (e.g., Security Level and Caveats) from each of the enclosed information elements; 4. (optional) Requests additional information (e.g., Operational Context, and/or User Location); 5. Requests the recipient's authorizations and privileges from the users' security services and infrastructure (via the ISSG); 6. Gathers metadata (e.g., security level, and caveats) from each of the information elements in the message; and

Table 9 - Email_PEP Component Operations

Element Name	**Operations**
	7. Packages a PDP-Request message and issues it to the PDP for adjudication and determination. The PDP determines if the recipient is authorized to receive each information element based on the recipients' authorizations and the markings on each information element. Optionally, policies may reflect the operational context in adjudicating its response. Upon receipt of the PDP-Response message, the PEP: 1. Extracts the PDP decisions and instructions from the PDP-Response. 2. Verifies the recipient is authorized to receive the content - if not: executes the processing instruction contained in the PDP-Response message. 3. Stages the decryption of the information element authorized for release to the recipient. ○ The PEP prepares the CTS-Request; and ○ Gathers transformed element from the CTS-Response. 4. Repackages the mail message with only the elements that the PDP has authorized for receipt, 5. Issues the repackaged mail message to the User's Email Client Application. 6. Prepares a TLS-LogMessage documenting the transaction and issues it to the TLS. This process is repeated for each received email. *Forward_Email_Message_to_Email_Client:* The Email-PEP provides features that forward authorized mail messages to the user`s email client application. *Decrypt_InformationElement:* The PEP provides features that stage the cryptographic transformation of each PDP authorized information element (body and attachments). For each authorized information element enclosed in the mail Message, the PEP: • Requests the unique symmetric cryptographic key from the user`s Key Management services through a request to the ISSG - the elements key is identified using the token bound to the information elements; • Package a CTS-Request containing the encryption key and the information element and issues the message to the CTS; and

Table 9 - Email_PEP Component Operations

Element Name	**Operations**
	- Receives the decrypted Information Element from the CTS-Response.
Send_Email	Email-PEP features that process email messages transiting from the User's email client application to the email server.
	Element Type: *Class* ***Owned Operations:*** *Stage_AuthorizationProcess:* The Email-PEP provides features that queue the (sent) mail message for authorization and encryption. As part of this process, the PEP: 1. Parses the mail message and extracts the enclosed information elements (body and Attachment(s)); 2. Verifies the mail message is transiting to the intended recipient - if not alert the administrator (see PAP-AlertWarning message) and terminate processing of the message. 3. Collects the metadata describing the sensitivity (e.g., Security Level and Caveats) from each of the enclosed information elements; 4. (optional) Requests additional information (e.g., Operational Context, and/or User Location); 5. Requests each recipient's authorizations and privileges from the user's security services and infrastructure (via the ISSG); 6. Gathers metadata (e.g., security level, and caveats) from each of the information elements in the message; and 7. Packages a PDP-Request message and issues it to the PDP for adjudication and determination. The PDP determines if each recipient is authorized to receive each information element based on the recipients' authorizations and the markings on each information element. Optionally, policies may reflect the operational context in adjudicating its response. Upon receipt of the PDP-Response message, the PEP: 1. Extracts the PDP decisions and instructions from the PDP-Response. 2. Verifies that each recipient is authorized to receive each of the information elements - if not applies the PDPs instructions*. 3. Stages** the decryption of the information elements authorized for release to the recipient. The PEP:

Table 9 - Email_PEP Component Operations

Element Name	Operations
	○ Prepares the CTS-Request; and ○ Gathers transformed elements from the CTS-Response. 4. Repackages the mail message with only the elements that the PDP has authorized for receipt, 5. Issues the repackaged mail message to the User's Email Client Application. 6. Prepares a TLS-LogMessage documenting the transaction and issues it to the TLS. The PDP determines if each recipient is authorized to receive each information element based on the recipients' authorizations and the markings on each information element. Optionally, policies may reflect the operational context in adjudicating it response. Upon receipt of the PDP-Response message, the PEP: 1. Extracts the PDP decisions and instructions from the PDP-Response. 2. Verifies that each recipient is authorized to receive each of the information elements - if not applies the PDPs instructions*. 3. Stages the decryption of the information element authorized for release to the recipient. The PEP: ○ Prepares the CTS-Request; and ○ Gathers transformed element from the CTS-Response. 4. Repackages the mail message with only the elements that the PDP has authorized for receipt, 5. Issues the repackaged mail message to the User's Email Client Application. 6. Prepares a TLS-LogMessage documenting the transaction and issues it to the TLS. * there are several processing options if the sender or a recipient does not have the appropriate authorizations, e.g.: • Terminate processing and return the email to the sender with an error message and have the sender correct the issue. • Extract the offending information element and send the remaining information elements to the recipients - send an error message to sender identifying the issue. • Send the email to only the recipients authorized to get all the information elements -

Table 9 - Email_PEP Component Operations	
Element Name	**Operations**
	• Send an error message to sender identifying the issue. It is user policy that dictates PEP action. These instructions should be provided by the PDP as part of its response. ** Note that the SAC may have to be removed from an IEF protected file prior to release because the recipient may not be able to handle the SAC structure or access the user's Key Escrow Service. This removal of the SAC protections may also require adjudication by the PDP. *Forward_Email_Message_to_Mail_Server:* The Email-PEP provides features that forward authorized mail messages to the email server application. *Encrypt_InformtionElement:* The PEP provides features that stage the cryptographic transformation of an email body and its attachments. For each informationElement the PEP enforces PDP instructions pertaining to the encryption of information elements within a mail message. For each information element in the email message, the encrypted information element operations perform the following functions: • Requests a unique symmetric cryptographic key from the Key Generation Service (KGS); • Packages an ISMB message containing the encryption key and the information element contained in the Email message; • Sends the message to the message specified CryptographicTransformationService (CTS); • Receives the encrypted Information Element; • Sends the information element data and the cryptographic key to the Key Escrow Service (KES); • Receives the Key Token from the KES; and • Binds the Key Token to the encrypted information element - as part of its SAC.
SMTP_Proxy	The Email PEP SMTP proxy is an intermediary service connecting the e-mail client to the mail server. The e-mail client connects to the proxy server, which requests service from the mail server. The proxy service adds the IEF structures and encapsulates the mail server. The proxy service adds the structures that enable the IEF to:

Table 9 - Email_PEP Component Operations	
Element Name	**Operations**
	• When the user sends an e-mail message, the e-mail client connects to the SMTP proxy. The proxy software receives the e-mail message and places the contents in a staging area. The proxy software then calls the Email PEP information protection logic, that is, the software validation routine that evaluates the message to ensure it meets policy and information protection requirements. • Once it is determined that the message can be sent and the message has been properly protected, it is forwarded on to the mail server for delivery.
	Element Type: *Class*

10.3 File Sharing PEP

The File Sharing PEP protects individual files as they are stored to and accessed from the IEF protected file share. The file sharing approach assures that each file stored and retrieved from the protected store is appropriately marked with security, privacy and caveat (warning orders) and supporting metadata. The markings are permanently bound to the files while they are protected by the IEF.

The IEF operates between the file management client and the protected-file share. The IEF intercepts all user requests and brokers the requested action (e.g., open, store, move, copy, cut, paste, and delete). The IEF verifies that for each action the user has the policy-rights to affect the requested operation on an information asset with the declared (/marked) sensitivities.

The IEF supports policy enforcement for the following files operations.

10.3.1 File_PEP Component Operations

The following figure identifies the key features of a File-PEP. This PEP intercepts each request from a user application to interact with a file or file-share protected by an IEF installation. The PEP validates and verifies that: each participant is authorized to access the specified files and perform the requested function in the specified location of the IEF protected file-share. The File-PEP also ensures that the file is marked by the user with the sensitively labels, the key token, and other supporting information.

The File-PEP is implemented as a proxy architecture where file requests are directed through the proxies to the IEF services that enforces user policy and applies needed information protection logic. The File-PEP provides information level protection for each individual file.

As a simple process, the PEP:

- Extracts the pertinent information (e.g., user Identification, request type, source, and/or target location) from the request message;
- Gathers the user's authorizations and privileges, role, responsibilities, location and any other information required for the PDP to adjudicate the request;
- Requests the file(s) and extracts its metadata (tags and labels) from the SAC header;
- Packages a PDP request and issues it to the PDP to be adjudicated;
- Receives the PDP's determination and enforces that decision;
- If the user is authorized to access the file(s), the PEP:
    - Requests the cryptographic key(s) for the file(s);
    - Orchestrates the decryption of the file(s);
    - Issues the file(s) to the User application.
    - Packages and sends a transaction report to the TLS.

This generic pattern will vary, based on the type to request (e.g., Get, Open, Copy, Cut, Paste, move, and Save), as well as the designs of the PDP, CTS, and ISSG.

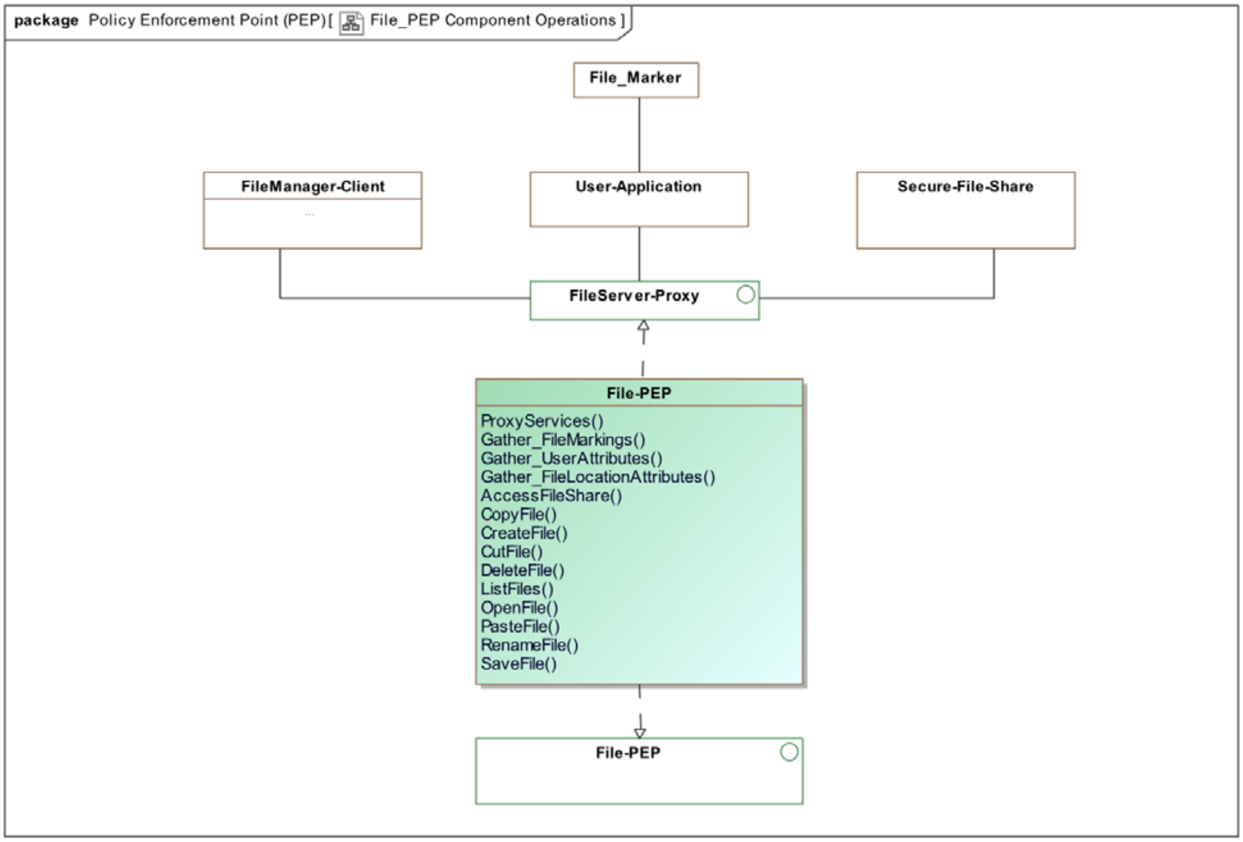


Figure 24 -File_PEP Component Operations

The following table identifies and describes the elements and operations illustrated in diagram "File_PEP Component Operations".

Table 10 - File_PEP Component Operations

Element Name	Operations
File-PEP	The File-PEP operates between a user application and the file server. The PEP intercepts each request, determines if the user is authorized to perform the requested action given sensitivities of each file.
	Element Type: *Class* ***Owned Operations:*** *ProxyServices:* Proxy services that intercepts file requests to and from a protected file-share. *Gather_FileMarkings:* The File-PEP provides features that extract files metadata attributes from the SAC data fields. *Gather_UserAttributes:* The File-PEP provides features that retrieve the user's identity from the requesting application, and uses that identifier to request the user's authorizations (/privileges / attributes) from the user's security infrastructure via the ISSG. *Gather_FileLocationAttributes:* The File-PEP provides features that extract the file location information from the application request, and requests the location access restrictions from the user's directory services (e.g., LDAP) via the ISSG. *AccessFileShare:* The File-PEP provides features that intercept a user application's request to access a specified file/share (device/folder). A request to access a file-share results in the provision of the contents of that file share to the user: a listing of resources (i.e., file and folders) that the user is authorized to access. The PEP: • Intercepts the request; • Extracts the users identity from the request; • Packages and issues an ISSG-Request for the user's authorizations and the authorization requirements for the specified file-share; • Packages and issues a PDP-Request; and • If authorized (PDP-Response), ListFiles().

Table 10 - File_PEP Component Operations

Element Name	**Operations**
	• Packages and sends a transaction report to the TLS. *CopyFile:* The File-PEP provides features that intercept each request to copy a file or files from one location to another (duplicate the file(s) in the later location), the PEP: • Intercepts the request; • Extracts the users identity from the request; • Extracts the markings from the SAC envelope header; • Packages and issues an ISSG-request for user authorizations; • If SAC is not present: ○ The PEP requests that the User provide sensitivity marking and other supporting metadata; ○ The PEP then generates a CTS-Request for encryption of the file and creation of its SAC. • Packages and issues a PDP-Request. • Verifies that the user is authorized to access the source device, directory and file. (PDP-Response). • Verifies that the user is authorized to access the target device and folder and write new file(s) with their individual sensitivities and restrictions. (PDP-Response) • If the operation is authorized, the PEP determines if the security policies necessitate that the duplicate file(s) requires a new cryptographic key. This obligation is provided to the PEP by the PDP within the PDP-responseMessage. • If a key (new SAC), the PEP packages a CTS-Request and issues the request to the CTS for processing. • Holds the encrypted file in a temporary storage location until the paste operation is requested; and • Packages and sends a transaction report to the TLS. Note: As part of its temporary store, the PEP maintains the identity of the requester, if the operation is a cut or copy (affecting the paste operation). *CreateFile:*

Table 10 - File_PEP Component Operations

Element Name	Operations
	The File-PEP provides features that intercept each request to create a file (of type) at a specified location in the IEF protected file share, the PEP: • Intercepts the request; • Extracts the user's identity from the request. • Creates the informationElement (file); • Requests the sensitivity level and restriction marking to be placed on the file. (ISSG-Request) • Packages and issues a PDP-Request. • Verifies that the user is authorized to create a file with the specified sensitivities and restrictions on that device and folder. (PDP-Response) • If the operation is authorized, packages a CTS-Request to generate a SAC for the information element. • Sends the request to the Secure-File-Share to store the SAC to the specified device and location. • Packages and sends a transaction report to the TLS. *CutFile:* The File-PEP provides features that intercept each request to cut a file or files from one location to another (move the file to the later location), the PEP: • Intercepts the request; • Extracts the user's identity from the request. • Extracts the markings contained in the SAC Envelope Header; • Packages and issue an ISSG-Request for required user authorizations; • If the marking are not present: ○ The PEP requests that the User provide sensitivity marking and other supporting metadata; ○ The PEP then generates a CTS-Request for encryption of the file and creation of its SAC. • Packages and issues a PDP-Request.Verifies that the user is authorized to access and remove the source device, directory and file. (PDP-Request).

Table 10 - File_PEP Component Operations

Element Name	**Operations**
	• Verifies that the user is authorized to access the target device and folder and write new file(s) with their individual sensitivities and restrictions. (PDP-Request) • If the operation is authorized, the PEP determines if the security policies necessitate that the duplicate file(s) requires a new cryptographic key. This obligation is provided to the PEP by the PDP as an instruction within the PDP-responseMessage. • If a new key (new SAC), the PEP packages a CTS-Request and issues the request to the CTS for processing. • Holds the encrypted file in a temporary store until the paste operation is requested. • Packages and sends a transaction report to the TLS. *DeleteFile:* The File-PEP provides features that intercept each request to delete (remove) a file or files from a specified device and folder. When a user requests that a file be deleted, PEP: • Intercepts the request; • Extracts the user's identity from the request. • Prepares and issues an ISSG-request for user authorizations to the Device and folder. • If the SAC is present, the PEP, extracts the files markings from the envelop header. • Packages and issues a PDP-Request with the available information. • Verifies that the user is authorized to access and remove the file from the device and directory. (PDP-Response). • If authorized, directs the secure file share to remove the files. • Packages and sends a transaction report to the TLS. *ListFiles:* The File-PEP provides features that intercept each request to list the contents of a device/folder. When a user requests a folder listing, PEP: • Intercepts the request; • Extracts the user's identity from the request.

Table 10 - File_PEP Component Operations	
Element Name	**Operations**
	• Prepares and issues an ISSG-Request for the users authorizations; • Requests the list of file names and attributes from the secure-file-share; • Packages and issues an ISSG-Request for additional resource authorization requirements and restrictions that may apply; • Packages and issues a PDP-Request to authorize access to the files; • For each file: ○ Verifies the user is authorized to see the file; (PDP-Response) ○ If not authorized, redacts the file names and attributes from the dataset. • Sends the redacted dataset to the File Manager Client; • Packages and sends a transaction report to the TLS. *OpenFile:* The File-PEP provides features that intercept a user application's request to open a specified file (device:folder/file). On receipt the PEP: • Intercepts the request; • Extracts the user's identity from the request. • Prepares and issues an ISSG-Request for the users' authorizations; • Packages and issues an ISSG-Request for additional resource authorization requirements and restrictions that may apply; • Requests the SAC from the files-share; • Extracts the file metadata from the Envelop header; • Packages and issues a PDP-Request to authorize access to the files; • If access is authorized (PDP-Response), ○ Packages and issues a CTS-Request containing the SAC. ○ Sends the decrypted file (CTS-Response) to the User Application; • Packages and sends a transaction report to the TLS.

Table 10 - File_PEP Component Operations

Element Name	**Operations**
	PasteFile: The File-PEP provides features that intercept each request to paste a file to a device/folder. When a user requests that a file is pasted to a device/folder, the PEP: • Intercepts the request; • Extracts the user's identity from the request; • Prepares and issues an ISSG-Request for the users authorizations; • Packages and issues an ISSG-Request for additional resource authorization requirements and restrictions that may apply to the target Device:/Folder; • Packages and issues a PDP-Request to authorize the operation on the files; • If the operation is authorized, copies and stores the file to the specified location (device:/folder); • If the operation that placed the SAC in the temporary store was "Cut" operation, delete the file from the original location; and • Packages and sends a transaction report to the TLS. *RenameFile:* The File-PEP provides features that intercept each request to rename a file. When a user requests that a file is renamed, the PEP: • Intercepts the request; • Extracts the user's identity from the request; • Prepares and issues an ISSG-Request for the users' authorizations; • Packages and issues an ISSG-Request for additional file authorization requirements and restrictions that may apply to the file; • Packages and issues a PDP-Request to authorize the operation; • If the operation is authorized, packages a CTS-Request to create a new SAC with the file name changed; • Stores the renamed SAC to the specified location; • Deletes the original SAC from the specified location; and • Packages and sends a transaction report to the TLS.

Table 10 - File_PEP Component Operations	
Element Name	**Operations**
	SaveFile: The File-PEP provides features that intercept each request to save a file to a device/folder. When a user (/user application) requests that a file is saved to a device/folder, the PEP: • Intercepts the request; • Extracts the users' identity from the request; • Packages and issues an ISSG-Request for the users' authorizations. • Packages and issues an ISSG-Request for additional resource authorization requirements and restrictions that may apply to the target Device:/Folder; • Packages and issues a PDP-Request to authorize the operation on the files; • If the operation is authorized, copies and stores the file to the specified location (device:/folder); and • If the operation that placed the SAC in the temporary store was "Cut" operation, delete the file from the original location. • Packages and sends a transaction report to the TLS. ***Inherited Operations:*** *Package_AuthorizationRequest* inherited from *PEP* *Parse_AuthorizationResponse* inherited from *PEP* *Enforce-AuthorizationDecisions* inherited from *PEP* *Package_SecurityServiceRequest* inherited from *PEP* *Parse_SecurityServiceResponse* inherited from *PEP* *Package-CTS-Request* inherited from *PEP* *Parse-CTS-Response* inherited from *PEP* *Execute-AdministrationFunctions* inherited from *PEP* *GatherSACMetadata* inherited from *PEP* *ExtractSACMetadata* inherited from *PEP* *Start_Operations* inherited from *IEF_Component* *Maintain-OperatingState* inherited from *IEF_Component* *Recover_Operations* inherited from *IEF_Component* *Track_RequestResponse* inherited from *IEF_Component* *Authorize_ActionRequest* inherited from *IEF_Component*

Table 10 - File_PEP Component Operations	
Element Name	**Operations**
	Package_AuthorizationRequest inherited from *IEF_Component* *Package-AdministrativeCommandResponse* inherited from *IEF_Component* *Package_EventLog* inherited from *IEF_Component* *Package_AlertWarningData* inherited from *IEF_Component* *Process_AdministrationCommand* inherited from *IEF_Component* *Configure_Properties* inherited from *IEF_Component* *Archive_Properties* inherited from *IEF_Component*
File_Marker	The File-Marker is a software service that integrates with off-the-shelf user applications that obliges and prompts the authorized user to appropriately mark each file with the required markings, The IEF requires that all elements of the email are appropriately marked (Tagged or labeled) in order to apply access and release policies.
	Element Type: *Class*
FileManager-Client	Off-the-shelf file manager or file browser that provides an application and interface which provides users with the ability to manage files and folders. Common operations performed on files or groups of files include listing/displaying, creating, opening, renaming, moving or copying, deleting and searching for files, as well as modifying file attributes, properties and file permissions.
	Element Type: *Class* ***Owned Operations:***
Secure-File-Share	The file share location that is to be protected by the File Sharing PEP is mounted in a staging area on the PEP's host file system. A file stored here is protected in a Secure Asset Container. Only an authorized user may retrieve the container and request that the file be decrypted.
	Element Type: *Class*
User-Application	Any user application that: • Access files stored by the IEF protected File-share; or

Table 10 - File_PEP Component Operations

Element Name	Operations
	- Access data sharable using structured messages.
	Element Type: *Class*

10.4 Instant Messaging PEP

The Instant Messaging (IM) PEP intercepts each message as it transits between the IM client and the IM server. The IM PEP enables users to establish a secure chat room that protects each message level. Each chat room is assigned a set of markings and a unique symmetric key for that room. Each participant in the chat room must have the policy rights to access the chat-room.

The IM PEP can also protect an individual message. A user can mark a message for special handling and limit access to selected users. These messages are assigned separate markings and are protected with their own unique symmetric key.

The IM PEP provides the following user operations:

- Chat Room Listing;
- Join Chat Room;
- Receiving a message;
- Sending a message;
- Sending a special message; and
- Receiving a marked-up message.

10.4.1 IM_PEP Component Operations

The following figure identifies the key features of an Instant Messaging-PEP. This configuration of a PEP intercepts each instant or text message transiting between an IM-Client and an IM-Server. The PEP validates and verifies that each participant is authorized to send or receive informationElements at the sensitivity and protection levels assigned a dedicated chat or chat room. The IM-PEP is implemented as a proxy architecture where the IM-mail traffic is redirected by the proxy to the IM-PEP that enforces the information policy and protection logic.

The IM-PEP features intercept each transaction between the IM client and server and ensures that the sender and receivers are authorized to participate in the exchange. The IM-PEP enforces the user defined policies for IM communications, including: chat-room identification and listing, chat-room creation; and chat-room participation.

Each chat-room has a specified set of sensitivity markings and is assigned its own cryptographic key that is applied to each message by the IM-PEP as in transits to the IM-Server. The IM-PEP also provides features that enable a user to “mark up” or flag a message for special handling. These messages are assigned separate security attributes and are protected with their own unique cryptographic key.

Users must have the policy-right to access a chat room, given the chat room’s sensitivity markings, before access to the room or message is granted. Once in the chat room, all normal messages are delivered to the user, but marked-up messages are delivered only if the user has the policy-right to see data with the associated security attributes.

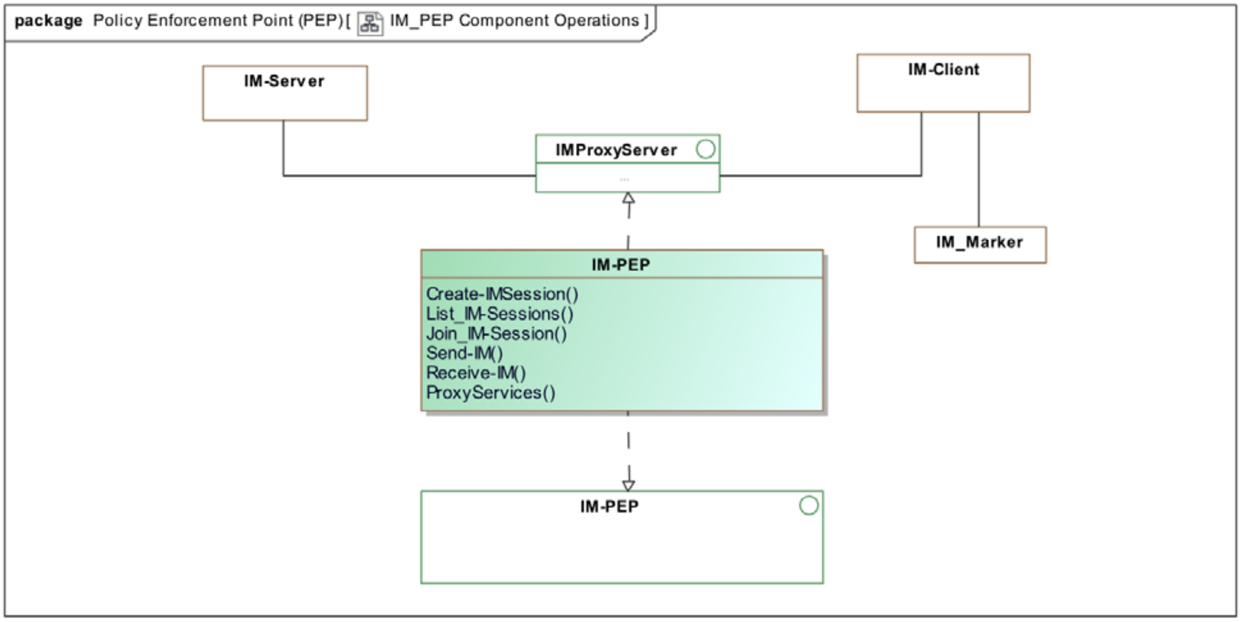


Figure 25 -IM_PEP Component Operations

The following table identifies and describes the elements and operations illustrated in diagram “IM_PEP Component Operations”.

Table 11 - IM_PEP Component Operations	
Element Name	**Operations**
IM-Client	User specified/provided off-the-shelf application that connects users to on-line chat rooms, which offer near real-time text transmission over the Internet or Intranet.
	Element Type: *Class*
IM-PEP	The IM-PEP provides features that intercept all interactions between the users’ Instant or Text Massaging client and the Instant Messaging (IM) server. The PEP intercepts each interaction between an IM client and server to determine if the

Table 11 - IM_PEP Component Operations

Element Name	**Operations**
	user (sender and receiver) has the appropriate privileges (/authorizations) to perform the requested action.
	Element Type: *Class* ***Owned Operations:*** *Create-IMSession:* The IM-PEP provides features that enable a user to set-up a secure chat-room. As part of this process the IM-PEP: • Intercepts the user request to create or establish a chat-room; • Parses and extracts the pertinent information from the request message; • If markings are not enclosed in the request message: ○ Request chat-room sensitivity markings from the user; and ○ Parse and extract the markings from the response message; • Request Authorization determinations from the PDP that the User is authorized to create a chat-room at the specified sensitivity level; • If authorized: ○ Request a Key from the User's Key management Services via the ISSG. ○ Request the chat-room from the IM-Server; and ○ Send notification to the requested participants; and ○ Notify the user that the chat-room has been created; and • Packages and sends a transaction report to the TLS. *List_IM-Sessions:* The IM-PEP provides features that provide a redacted list of active chat-rooms to a user's request to list the active chat-rooms. To provide this feature the IM-PEP: • Intercepts the user request to list available community/public chat-rooms; • Parses and extracts the pertinent information from the request message. • Requests the list from the IM Server;

Table 11 - IM_PEP Component Operations

Element Name	**Operations**
	• Requests release authorization determinations from the PDP. • Redacts the names and identifiers of the chat-rooms that user is not authorized; • Packages and sends the redacted list to the user; and • Packages and sends a transaction report to the TLS. *Join_IM-Session:* The IM-PEP provides features that enable a user to join a chat-room they are authorized to access. As part of this process the IM-PEP: • Intercepts the user request to create or establish a chat-room; • Parses and extracts the pertinent information from the request message. • Gathers the Sensitivity Markings for the chat-room; • Requests the Users Authorization from the user's privilege (/authorization / attribute) management services via the ISSG. • Request Authorization determination from the PDP that the user is authorized to join the chat-room. • If authorized, adds the user to the distribution list for the chat-room. • Packages and sends a transaction report to the TLS. *Send-IM:* The IM-PEP provides features that validate and verify a message sent to a chat-room is authorized for release to that chat-room. As part of this process the IM-PEP: • Intercepts the message; • Parses and extracts the pertinent information from the send message; • Requests authorization determination from the PEP that the user is authorized to send information of that sensitivity level to the specified chat-room • If authorized: ○ Requests the cryptographic key for the chat-room. ○ Sends Encryption Request (Package SAC) with the cryptographic key and

Table 11 - IM_PEP Component Operations

Element Name	Operations
	informationElement to the CTS; (CTS-Request Message) ○ Receives and processes the CTS-Response message; ○ Packages and sends the IM Message (SAC) to the IM-Server; and • Packages and sends a transaction report to the TLS. When a user sends an IM message, the message is intercepted by the IM PEP and encrypted so that it is stored in a protected form at the IM server. The room's cryptographic key is retrieved from escrow and used to encrypt the message that is then sent to the IM server for delivery to all room participants. Sending a marked-up message: A marked up message must be protected inside a virtual private room with the room having its own access control and key management. When the IM PEP receives a marked-up message it extracts the caveat from the message. The IEF ensures that the user has the policy-right to send data with this security attribute. If the action in permitted, the IEF requests a new key for this chat room/caveat combination, encrypts the message and sends the protected message to the IM server for delivery to all authorized chat room participant(s). *Receive-IM:* The IM-PEP provides features to intercept an IM-message from the server, and validate and verify that the recipient is authorized to receive the contents of the message. As part of this process the IM-PEP: • Intercepts the message; • Parses and extracts the pertinent information from the send message; • If packaged as a SAC, processes the SAC to extract sensitivity markings of a special message and the InformationElement Token. • Requests authorization determination from the PEP that the user is authorized to receive information of that sensitivity level from the specified chat-room; • If authorized: ○ Request the cryptographic key for the chat-room and/or the SAC. ○ Send decryption Request with the InformationElement and cryptographic key to the CTS; (CTS-Request Message)

Table 11 - IM_PEP Component Operations

Element Name	**Operations**
	○ Receive and process the CTS-Response message; ○ Package and send the IM Message (Decrypted) to the IM-Client; and • Package and send a transaction report to the TLS. *ProxyServices:* Proxy services that intercepts instant/text messages to and from a protected IM service. ***Inherited Operations:*** *Package_AuthorizationRequest* inherited from *PEP* *Parse_AuthorizationResponse* inherited from *PEP* *Enforce-AuthorizationDecisions* inherited from *PEP* *Package_SecurityServiceRequest* inherited from *PEP* *Parse_SecurityServiceResponse* inherited from *PEP* *Package-CTS-Request* inherited from *PEP* *Parse-CTS-Response* inherited from *PEP* *Execute-AdministrationFunctions* inherited from *PEP* *GatherSACMetadata* inherited from *PEP* *ExtractSACMetadata* inherited from *PEP* *Start_Operations* inherited from *IEF_Component* *Maintain-OperatingState* inherited from *IEF_Component* *Recover_Operations* inherited from *IEF_Component* *Track_RequestResponse* inherited from *IEF_Component* *Authorize_ActionRequest* inherited from *IEF_Component* *Package_AuthorizationRequest* inherited from *IEF_Component* *Package-AdministrativeCommandResponse* inherited from *IEF_Component* *Package_EventLog* inherited from *IEF_Component* *Package_AlertWarningData* inherited from *IEF_Component* *Process_AdministrationCommand* inherited from *IEF_Component* *Configure_Properties* inherited from *IEF_Component* *Archive_Properties* inherited from *IEF_Component*

Table 11 - IM_PEP Component Operations

Element Name	**Operations**
IM-Server	User specified/provided off-the-shelf server that retransmits Instant Messages from the sender to the receiver. IEF services do not alter the standard headers or protocols, however, the content of the message is encrypted both at rest and in motion. The IEF ensures that all messages are authorized, appropriately marked, and encrypted.
	Element Type: *Class*
IM_Marker	The IM-Marker is a software service that integrates with off-the-shelf IM clients that obliges and prompts the authorized user to appropriately mark the text messages. The IEF requires that text messages are appropriately marked (Tagged or labeled) in order to apply access and release policies.
	Element Type: *Class*

10.5 Messaging PEP

A Messaging PEP intercepts each message as it transits between User Application and a PPS data provisioning service over a messaging bus or data bus. The PEP verifies that:

- The content (message body and all attachments) is encrypted while at rest and in transit;
- The sender of the message has the policy-rights to send the specific message content; and
- Each recipient is individually authorized (has the policy-rights) to receive and access the content of the message body and each of the attachments.

The Messaging PEP applies protections to each element (email body and all attachment) included in the email message. Each element must therefore be bound with all relevant sensitivity markings (e.g., privacy, confidentiality, classification, legal significance), and caveats (warning orders and restrictions) necessary for the IEF services to determine the policy defined protections appropriate to the emails content.

The Messaging PEP can be implemented in two different configurations:

1. As a proxy service between the User Application and PPS in the same manner as the Email, File Share, and IM PEPs; or
2. As a component on the ISMB.

The following clauses illustrate and describe the two Messaging PEP configurations.

10.5.1 Messaging_PEP Integrated Data Service

The following figure depicts a Messaging PEP configuration where the user's information services are situated within the IEF environment (directly connected to the ISMB). The figure identifies the key capabilities of the Messaging-PEP. The Messaging-PEP intercepts messages transiting between the users specified Information Exchange Service(s) and the PPS. The proxy service intercepts each message, routes the InformationElements (/messages), and validates and verifies that each participant is authorized to send, and/or receive the information content contained in that message. The PEP provides both outgoing message authorization (determining authorization to release) and protection for local data stores (authorization to receive). The former features assure the producer is authorized to send the information content to the specified recipients, and that the specified recipients are authorized to access (/use) the information. The latter features assure that the receiving system is authorized to receive, access, use and store the content of the message. If sensitivity of the information exceeds the authorizations of the system and data stores - the integration of the information could raise the security requirements for the entire environment.

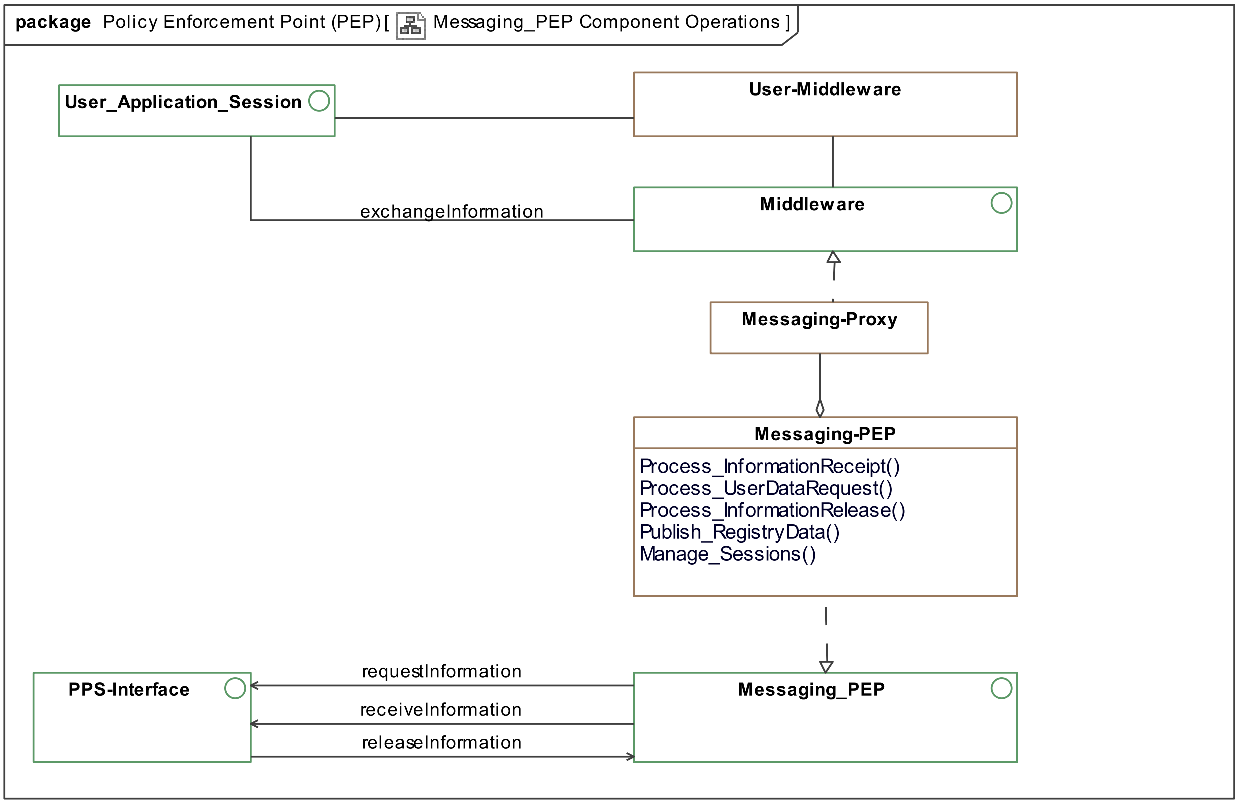


Figure 26 -Messaging_PEP Component Operations

The following table identifies and describes the elements and operations illustrated in diagram "Messaging_PEP Component Operations".

Table 12 - Messaging_PEP Component Operations

Element Name	Operations
Messaging-PEP	The Messaging-PEP provides access and release control for: • User applications requesting information from a PPS, and • PPS pushing information to a user application or middleware. The Messaging-PEP authorizes release and receipt based on the users (sender and receiver) privileges and current user specified policy. On receipt, the Messaging-PEP verifies that the local PPS and data-store(s) are authorized to access, process and store the contents of the Message. On Release, the Messaging-PEP ensures that each recipient to the message is authorized to receive the contents of the message, and that the specific communication mechanism (e.g., topic, queue, communication channel) is authorized to transport the contents of the message.
	Element Type: *Class* ***Owned Operations:*** *Process_InformationReceipt:* The Messaging-PEP provides features that stage the processing of an information message received from the user specified middleware. The messaging protocol is stripped by the interface prior to the commencement of processing. The Messaging-PEP then: • Identifies the message type. • Gathers the rules (message protocol) governing the structure and content of the message type. • De-constructs the message to collect the embedded InformationElements. • Collects the metadata about each InformationElement. Message metadata may include: ○ Message Metadata - data elements describing the content of the InformationElement (/Message); ○ Data Owner Metadata - data elements identifying the owner/steward of the content of the InformationElement (/Message); ○ Privacy Metadata - data elements describing the private content and release restriction

Table 12 - Messaging_PEP Component Operations	
Element Name	**Operations**
	associated with the InformationElement (/Message); ○ Security Metadata - data elements describing the classified content and release restriction associated with the InformationElement (/Message); and ○ Handling Instructions - data elements describing any specialized data/information access, processing or storage instructions for the InformationElement (/Message). • Requests the privileges and/or authorizations of the recipient (i.e., target PPS); • (Optional) Requests the operational context for the exchange from the user's SA or incident management system. • Packages and issues an authorization request to the PDP. • If authorized, decrypts the InformationElement(s) provided in the message: ○ Gathers Cryptographic Key Tokens from the Information element metadata, ○ Requests Cryptographic Keys for authorized information elements from the user's Key Escrow Service (ISSG-Request), and ○ Packages a CTS-Request for the decryption of each authorized information element. • If authorized, packages and issues the Metadata and decrypted InformationElement to the PPS for processing. • Logs the transaction to the TLS. *Process_UserDataRequest:* The Messaging-PEP provides features that orchestrate the authorization or rejection of a user data request and issue the request to the PPS. A user request to the PPS must include: • User's Identity; • References to the SemanticElement or FilteredSemanticElement and the object(s) to be reported on. These references should be discoverable from the users data registry (used to discover data and information elements in the environment); • Whether this is a one-time request or a request for all available updates on the objects requested; and

Table 12 - Messaging_PEP Component Operations

Element Name	Operations
	• Release instructions (e.g. communication channel, and messaging protocol). The PEP gathers the users' identity and privilege information, sensitivity of the information being requested, and optionally, location and SA information. These elements are packaged (PDP-Request), if Authorized by the PDP, The PEP packages and issues a PPS-Request message to the PPS for processing. The PEP then logs the transaction to the TLS. *Process_InformationRelease:* The Messaging-PEP provides features that stage the processing of an InformationElement(s) from a PPS for release to the user specified middleware. The ISMB messaging protocol is stripped by the interface prior to the commencement of processing. The Messaging-PEP then: • Extracts the message-metadata, which includes: ○ Message Type, ○ Security Level, ○ Warning Orders or Caveats, ○ Privacy Indicators, ○ Sender Identification; ○ Recipient(s) Identification; ○ Target Communication Channel; and ○ Target Protocol. • Requests the privileges and/or authorizations of the recipient(s); • (Optional) Requests the operational context for the exchange from the user's SA or incident management system; • Packages and issues an authorization request to the PDP; • If authorized, encrypts the authorized InformationElement(s): ○ Requests Cryptographic Keys and tokens from the users Key Management Service(s) (ISSG-Request), and ○ Packages a CTS-Request for the encryption of each authorized information element.

Table 12 - Messaging_PEP Component Operations

Element Name	**Operations**
	• Packages and issues the Message to the middleware for dissemination. • Logs the transaction. *Publish_RegistryData:* (Optional) The Messaging-PEP provides features that gather Metadata about InformationElement(s) available through the attached PPS. The Registry may include information on: • Supported Information Topics; • Supported Message Queues; • PPS supported SemanticElements, FilteredSemanticElements & Messages; and • Historical InformationElements stored to disk. These entries enable the discovery of IEF protected information holdings. The Data Registry is not an IEF defined component, it is assumed to be part of the user defined infrastructure and the format of the registry data message will be determined by the registry selected. The PEP provisions registry information using the user specified middleware services. These features may also be used to issue an InformationElement to a records and document management system (RDMS). In this case the InformationElement is included as an encrypted attachment to the registry message. *Manage_Sessions:* The Messaging-PEP provides features that execute PAP commands directing it to administer and manage communication channels (e.g., topics, and queues). The PAP can direct the Messaging-PEP to create or modify the available communication channels. ***Inherited Operations:*** *Package_AuthorizationRequest* inherited from *PEP* *Parse_AuthorizationResponse* inherited from *PEP* *Enforce-AuthorizationDecisions* inherited from *PEP* *Package_SecurityServiceRequest* inherited from *PEP* *Parse_SecurityServiceResponse* inherited from *PEP* *Package-CTS-Request* inherited from *PEP* *Parse-CTS-Response* inherited from *PEP* *Execute-AdministrationFunctions* inherited from *PEP* *GatherSACMetadata* inherited from *PEP*

Table 12 - Messaging_PEP Component Operations

Element Name	**Operations**
	ExtractSACMetadata inherited from *PEP* *Start_Operations* inherited from *IEF_Component* *Maintain-OperatingState* inherited from *IEF_Component* *Recover_Operations* inherited from *IEF_Component* *Track_RequestResponse* inherited from *IEF_Component* *Authorize_ActionRequest* inherited from *IEF_Component* *Package_AuthorizationRequest* inherited from *IEF_Component* *Package-AdministrativeCommandResponse* inherited from *IEF_Component* *Package_EventLog* inherited from *IEF_Component* *Package_AlertWarningData* inherited from *IEF_Component* *Process_AdministrationCommand* inherited from *IEF_Component* *Configure_Properties* inherited from *IEF_Component* *Archive_Properties* inherited from *IEF_Component*
Messaging-Proxy	The User-Application proxy is an intermediary service connecting a user application client to the user's Information service (PPS data provisioning). The user application connects to the proxy server, which requests service from the mail information services. The proxy service adds the IEF information protection features to the information services. The proxy service adds the structures that enable the IEF to: • When the user requests information, the proxy service intercepts the message and places the contents in a staging or processing area. The proxy service then calls the Messaging PEP to validate and verify whether or not the user is authorized to access: ○ The Information Service; and ○ The Information content being requested. • When the user sends information, the proxy service intercepts the message and places the contents in a staging or processing area. The proxy service then calls the Messaging PEP to apply the IEF protection logic that validates whether or not the user is authorized to access: ○ The Information Service; ○ Once it is determined that the message can be sent and the message has been properly

Table 12 - Messaging_PEP Component Operations

Element Name	**Operations**
	protected, it is forwarded on to the mail server for delivery.
	Element Type: *Class*
User-Middleware	User selected and implemented software that enables information services, application and/or systems to share information and data elements (e.g., DDS, Enterprise Services Bus, or Web Services).
	Element Type: *Class*

10.5.2 Messaging_PEP Proxy Component Operations

The following figure depicts a Messaging PEP configuration where the user's information services are situated outside the IEF environment. The figure identifies the key capabilities of the Messaging-PEP which do not functionally change between configurations beyond the addition of the Proxies.

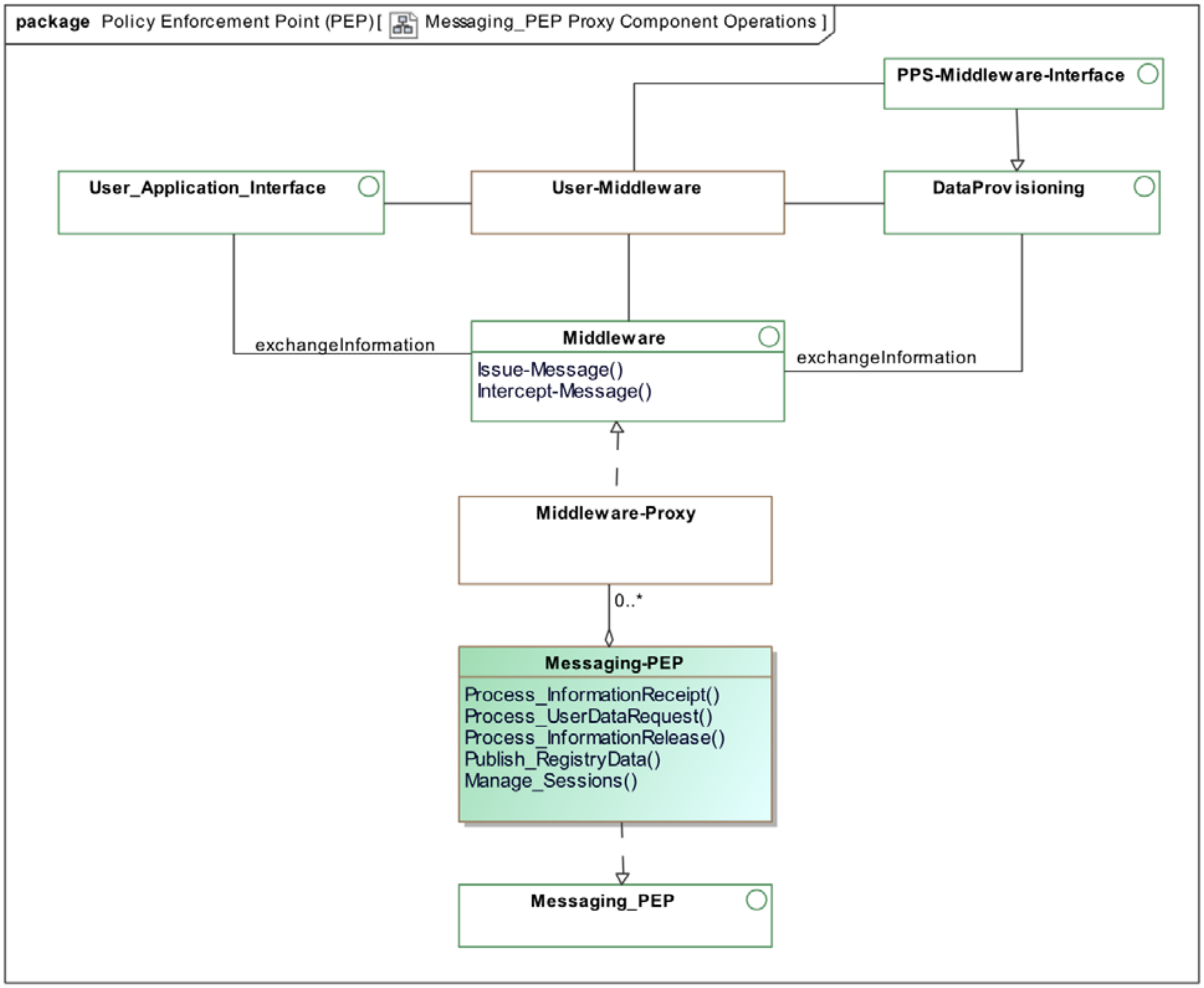


Figure 27 -Messaging_PEP Proxy Component Operations

The following table identifies and describes the elements and operations illustrated in diagram “Messaging_PEP Proxy Component Operations”.

Table 13 - Messaging_PEP Proxy Component Operations

Element Name	**Operations**
Messaging-PEP	The Messaging-PEP provides access and release control for: • User applications requesting information from a PPS, and • PPS pushing information to a user application or middleware. The Messaging-PEP authorizes release and receipt based on the users (sender and receiver) privileges and current user specified policy. On receipt, the Messaging-PEP verifies that the local PPS and data-store(s) are authorized to access, process and store the contents of the Message. On Release, the Messaging-PEP ensures that each recipient to the message is authorized to receive the contents of the message, and that the specific communication mechanism (e.g., topic, queue, communication channel) is authorized to transport the contents of the message.
	Element Type: *Class* ***Owned Operations:*** *Process_InformationReceipt:* The Messaging-PEP provides features that stage the processing of an information message received from the user specified middleware. The messaging protocol is stripped by the interface prior to the commencement of processing. The Messaging-PEP then: • Identifies the message type. • Gathers the rules (message protocol) governing the structure and content of the message type. • De-constructs the message to collect the embedded InformationElements. • Collects the metadata about each InformationElement. Message metadata may include: ○ MessageMetadata - data elements describing the content of the InformationElement (/Message); ○ Data Owner Metadata - data elements identifying the owner/steward of the content of the InformationElement (/Message); ○ Privacy Metadata - data elements describing the private content and release restriction associated with the InformationElement (/Message); ○ Security Metadata - data elements describing the classified content and release restriction

Table 13 - Messaging_PEP Proxy Component Operations

Element Name	Operations
	associated with the InformationElement (/Message); and ○ Handling Instructions - data elements describing any specialized data/information access, processing or storage instructions for the InformationElement (/Message). • Requests the privileges and/or authorizations of the recipient (i.e., target PPS); • (Optional) Requests the operational context for the exchange from the user's SA or incident management system. • Packages and issues an authorization request to the PDP. • If authorized, decrypts the InformationElement(s) provided in the message: ○ Gathers Cryptographic Key Tokens from the Information element metadata, ○ Requests Cryptographic Keys for authorized information elements from the user's Key Escrow Service (ISSG-Request), and ○ Packages a CTS-Request for the decryption of each authorized information element. • If authorized, packages and issues the Metadata and decrypted InformationElement to the PPS for processing. • Logs the transaction to the TLS. *Process_UserDataRequest:* The Messaging-PEP provides features that orchestrate the authorization or rejection of a user data request and issue the request to the PPS. A user request to the PPS must include: • User's Identity; • References to the SemanticElement or FilteredSemanticElement and the object(s) to be reported on. These references should be discoverable from the users data registry (used to discover data and information elements in the environment); • Whether this is a one-time request or a request for all available updates on the objects requested; and • Release instructions (e.g. communication channel, and messaging protocol).

Table 13 - Messaging_PEP Proxy Component Operations

Element Name	**Operations**
	The PEP gathers the users' identity and privilege information, sensitivity of the information being requested, and optionally, location and SA information. These elements are packaged (PDP-Request), if Authorized by the PDP, The PEP packages and issues a PPS-Request message to the PPS for processing. The PEP then logs the transaction to the TLS. *Process_InformationRelease:* The Messaging-PEP provides features that stage the processing of an InformationElement(s) from a PPS for release to the user specified middleware. The ISMB messaging protocol is stripped by the interface prior to the commencement of processing. The Messaging-PEP then: • Extracts the message-metadata, which includes: ○ Message Type, ○ Security Level, ○ Warning Orders or Caveats, ○ Privacy Indicators, ○ Sender Identification; ○ Recipient(s) Identification; ○ Target Communication Channel; and ○ Target Protocol. • Requests the privileges and/or authorizations of the recipient(s); • (Optional) Requests the operational context for the exchange from the user's SA or incident management system; • Packages and issues an authorization request to the PDP; • If authorized, encrypts the authorized InformationElement(s): ○ Requests Cryptographic Keys and tokens from the users Key Management Service(s) (ISSG-Request), and ○ Packages a CTS-Request for the encryption of each authorized information element. • Packages and issues the Message to the middleware for dissemination.

Table 13 - Messaging_PEP Proxy Component Operations

Element Name	Operations
	• Logs the transaction. *Publish_RegistryData:* (Optional) The Messaging-PEP provides features that gather Metadata about InformationElement(s) available through the attached PPS. The Registry may include information on: • Supported Information Topics; • Supported Message Queues; • PPS supported SemanticElements, FilteredSemanticElements & Messages; and • Historical InformationElements stored to disk. These entries enable the discovery of IEF protected information holdings. The Data Registry is not an IEF defined component, it is assumed to be part of the user defined infrastructure and the format of the registry data message will be determined by the registry selected. The PEP provisions registry information using the user specified middleware services. These features may also be used to issue an InformationElement to a records and document management system (RDMS). In this case the InformationElement is included as an encrypted attachment to the registry message. *Manage_Sessions:* The Messaging-PEP provides features that execute PAP commands directing it to administer and manage communication channels (e.g., topics, and queues). The PAP can direct the Messaging-PEP to create or modify the available communication channels. ***Inherited Operations:*** *Package_AuthorizationRequest* inherited from *PEP* *Parse_AuthorizationResponse* inherited from *PEP* *Enforce-AuthorizationDecisions* inherited from *PEP* *Package_SecurityServiceRequest* inherited from *PEP* *Parse_SecurityServiceResponse* inherited from *PEP* *Package-CTS-Request* inherited from *PEP* *Parse-CTS-Response* inherited from *PEP* *Execute-AdministrationFunctions* inherited from *PEP* *GatherSACMetadata* inherited from *PEP* *ExtractSACMetadata* inherited from *PEP*

Table 13 - Messaging_PEP Proxy Component Operations	
Element Name	**Operations**
	Start_Operations inherited from *IEF_Component* *Maintain-OperatingState* inherited from *IEF_Component* *Recover_Operations* inherited from *IEF_Component* *Track_RequestResponse* inherited from *IEF_Component* *Authorize_ActionRequest* inherited from *IEF_Component* *Package_AuthorizationRequest* inherited from *IEF_Component* *Package-AdministrativeCommandResponse* inherited from *IEF_Component* *Package_EventLog* inherited from *IEF_Component* *Package_AlertWarningData* inherited from *IEF_Component* *Process_AdministrationCommand* inherited from *IEF_Component* *Configure_Properties* inherited from *IEF_Component* *Archive_Properties* inherited from *IEF_Component*
Middleware-Proxy	A Proxy Service that intercepts messages to and from a protected structured data source.
	Element Type: *Class*
User-Middleware	User selected and implemented software that enables information services, application and/or systems to share information and data elements (e.g., DDS, Enterprise Services Bus, or Web Services).
	Element Type: *Class*

11 Data Packaging and Processing Services

The Packaging and Processing Service (PPS) is an integrated policy decision and enforcement point for the packaging and processing of structured messages.

11.1 Policy-based Packaging and Processing Service (PPS)

The Policy-based packaging service (PPS) executes and enforces policy (rules and constraints) governing the packaging (aggregation, transformation, marking, filtering, structuring and formatting) of messages for release to authorized recipients and process (parse, transform, and marshal) messages received by the messaging-PEP. The PPS provides the ability, based on policy, to transition data and information elements between canonical exchange models (e.g., NIEM, EDXL, and HL7) and user specified data stores (e.g., RDBMS).

The PPS ingests Information sharing and safeguarding policy conforming to the Information Exchange Packaging Policy Vocabulary (IEPPV) and executes the packaging and processing rules and constraints defined by its semantics.

11.1.1 PPS Component Operations

The following figure identifies the core features and functions provided by a Policy-based Packaging and Processing Service.

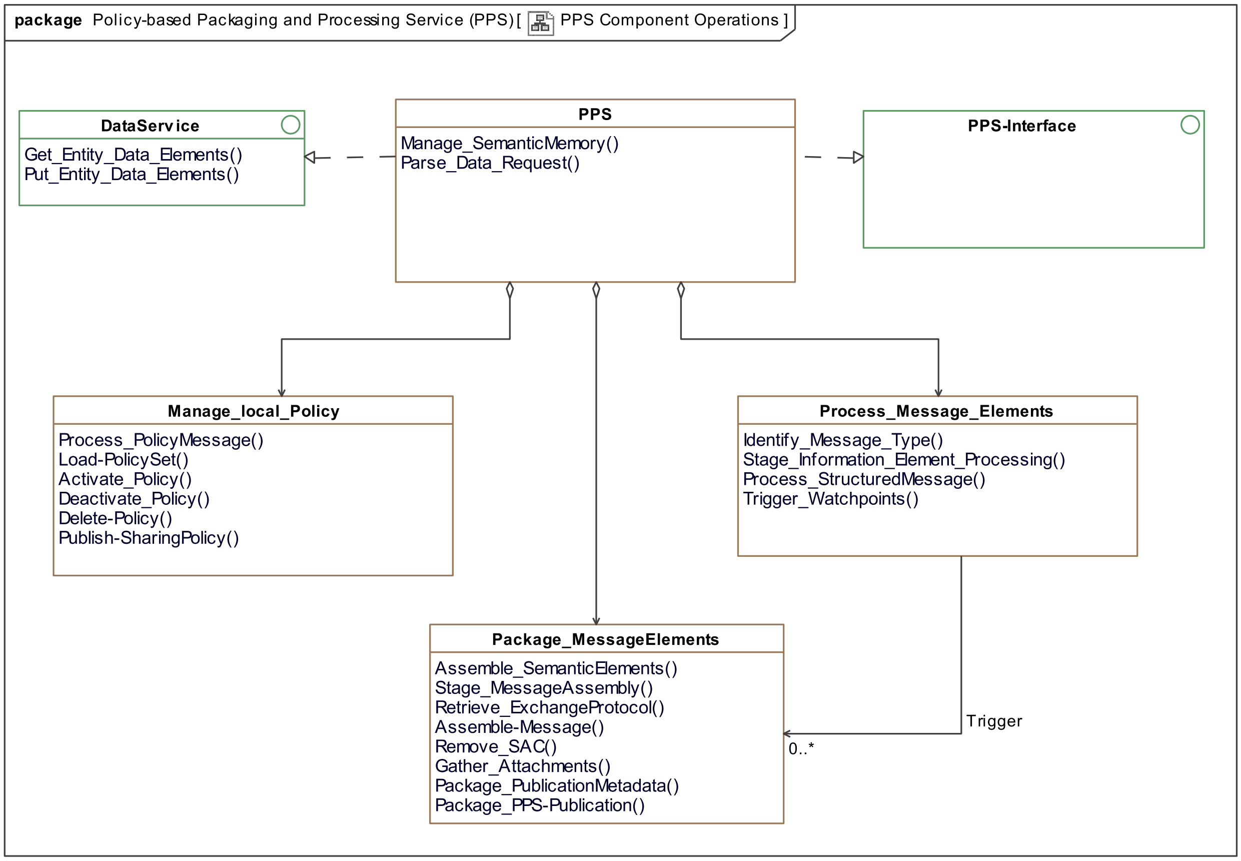


Figure 28 -PPS Component Operations

The following table identifies and describes the elements and operations illustrated in diagram “PPS Component Operations”.

Table 14 - PPS Component Operations

Element Name	**Operations**
Manage_local_Policy	The PPS provides features that manage and administer policies in accordance with PAP instructions (see PAP-Command message).
	Element Type: *Class* ***Owned Operations:*** *Process_PolicyMessage:* The PPS provides features that extract IEPPV policy statements and their relationships from a PAP-Command message, verify that the policies conform to IEPPV requirements and prepare* the statement for use by the PPS. * Prepare may include a reformatting and restructuring of the policy statements to conform to the storage and processing requirements of the PPS. *Load-PolicySet:* The PPS provides features that integrate a policy, or set of policies that have been processed into its set of operational policies. *Activate_Policy:* The PPS provides features that in response to an authorized PAP-Command, activate one or more of its operational policies. *Deactivate_Policy:* The PPS provides features that in response to an authorized PAP-Command, deactivate one or more of its operational policies. *Delete-Policy:* The PPS provides features that in response to an authorized PAP-Command, remove one or more of its operational policies. *Publish-SharingPolicy:* The PPS provides features that in response to an authorized PAP-Command, packages one or more of its operational policies and: • Sends it to the PAP as part of the PAP-CommandResponse message;

Table 14 - PPS Component Operations

Element Name	**Operations**
	• Creates a SecureAssetContainer for the policies and marshal the container to the Protected Information Store location also specified in the PAP-Command message.
Package_MessageElements	Identifies the set of capabilities required by the PPS to execute the Information Exchange Specification elements of the policy. (See IEPPV specification for details).
	Element Type: *Class* ***Owned Operations:*** *Assemble_SemanticElements:* The PPS provides features that assemble data and information elements tailored to authorizing policy for the specified recipient(s) of the publication. InformationElements* that may be assembled include: • Digest; • Information Package(s); • Information Payload(s); and • Message Metadata. According to the IEPPV, a structured message must contain message metadata, and one information payload, but may contain a Digest, [0..*] payloads and [0..*] attachments. These semantic elements are assembled and formatted individually, as required, and then combined into the message protocol. * See IEPPV for descriptions of these Information Elements. *Stage_MessageAssembly:* The PPS provides features that manage the assembly of message elements (e.g., metadata, digest, packages, payloads and attachments) in accordance with policy and established protocols (e.g., LEXS). *Retrieve_ExchangeProtocol:* PPS features that retrieves the exchange protocol* (e.g., XML schema) from a location in the IEF protected information store or request it through the ISSG. * An exchange protocol may be required for each element in the message structure. *Assemble-Message:*

Table 14 - PPS Component Operations

Element Name	Operations
	The PPS provides features that gather the releasable datasets for each of the message elements (digest, packages, payloads and attachments) as specified in the InformationExchangeSpecification, transform a releasable (as per data policy) dataset to the specified message protocol (e.g., XSD), and packages the overall message in accordance with its protocol (e.g., LEXS). *Remove_SAC:* The PPS provides features to remove an information element from a SAC prior to its packaging for release. *Gather_Attachments:* PPS features that gather Attachments specified for inclusion in the message. Attachments are identified as specialized WrapperElements in the Information Policy Sets. *Package_PublicationMetadata:* The PPS provides features that package message metadata from the releasable dataset, the InformationExchangeSpecification and policy store. This metadata includes message metadata, plus release and handling instructions. *Package_PPS-Publication:* The PPS provides features that package the message (payload) and the Message metadata for release to the messaging PEP as a PPS-Publication message.
PPS	The Policy-based Packaging and Processing Service (PPS) transitions structured InformationElements (e.g., NIEM, EDXL, and HL7) between data stores and information exchange services in accordance with local information sharing and safeguarding policies conforming to the Information Exchange Packaging Policy Vocabulary (IEPPV). The PPS provides the ability to selectively package (aggregate, transform, mark, filter, structure and format) informationElements for publication to authorized recipients. It also provides the ability to process (parse, transform, and marshal) structured messages and integrate the data elements into the user's data stores.
	Element Type: *Class* ***Owned Operations:*** *Manage_SemanticMemory:* The PPS provides features that manage the memory it uses to package and process its data elements. The PPS provides the ability to operate completely in volatile memory (without

Table 14 - PPS Component Operations

Element Name	**Operations**
	persistence) or in a hybrid mode with use of both volatile memory and a persistent data store. *Parse_Data_Request:* The PPS provides features that decompose a request for information, extracting the pertinent information and triggering the appropriate semantic packaging process - as specified in the PPS policies. ***Inherited Operations:*** *Start_Operations* inherited from *IEF_Component* *Maintain-OperatingState* inherited from *IEF_Component* *Recover_Operations* inherited from *IEF_Component* *Track_RequestResponse* inherited from *IEF_Component* *Authorize_ActionRequest* inherited from *IEF_Component* *Package_AuthorizationRequest* inherited from *IEF_Component* *Package-AdministrativeCommandResponse* inherited from *IEF_Component* *Package_EventLog* inherited from *IEF_Component* *Package_AlertWarningData* inherited from *IEF_Component* *Process_AdministrationCommand* inherited from *IEF_Component* *Configure_Properties* inherited from *IEF_Component* *Archive_Properties* inherited from *IEF_Component*
Process_Message_Elements	Identifies the set of capabilities required by the PPS to manage local policies in accordance with PAP instructions (see PAP-Command message).
	Element Type: *Class* ***Owned Operations:*** *Identify_Message_Type:* The PPS provides features that identify the type of information message being received from the PEP. *Stage_Information_Element_Processing:* The PPS provides features that stage the processing of each of the message elements (digest, packages, payload, and attachments) in accordance with data policy. *Process_StructuredMessage:*

Table 14 - PPS Component Operations	
Element Name	**Operations**
	The PPS provides features that decompose the InformationElement (Payload) into its parts (data elements) and interrelationships, populate the appropriate semantic patterns* permitted by PPS policy and (optionally) marshal the data elements to the user's data store(s). Marshaling and persisting data elements to the specified data store is optional, a user may direct the PPS to operate only in volatile memory and not persist the information. * Semantic (data) patterns are defined by Semantic, Transactional and Wrapper Elements in PPS policy conforming to the IEPPV. *Trigger_Watchpoints:* The PPS provides features that, on a specified data change, the PPS triggers the packaging and publication of messages to all recipients requiring and authorized to receive updates. (See IEPPV for details).

12 Security Services Gateway

12.1 IEF Security Service Gateway (ISSG)

The Security Services Gateway (ISSG) provides a single secure interface between IEF components and the user's specified security services and infrastructure. The ISSG intercepts all communication between IEF components and the user's security services (e.g., Identity Management, Privilege Management, and Key Management) infrastructure and the ISS supporting services (e.g., situational Awareness)). The ISSG:

- Provides messaging interfaces for both:
    - The ISMB; and
    - The Users messaging infrastructure;
- Authorizes each request (gains authorization from the PDP); and
- Translates the requests and responses between IEF protocols and user networking protocols.

12.2 ISSG Component Operations

The following figure identifies the IEF Security Services Gateway interfaces that provide the integration point between IEF components and user specified security services, including: Identity Management, Privilege/Attribute Management, Cryptographic, TrustMark Provider, and Policy Development and Management Environments.

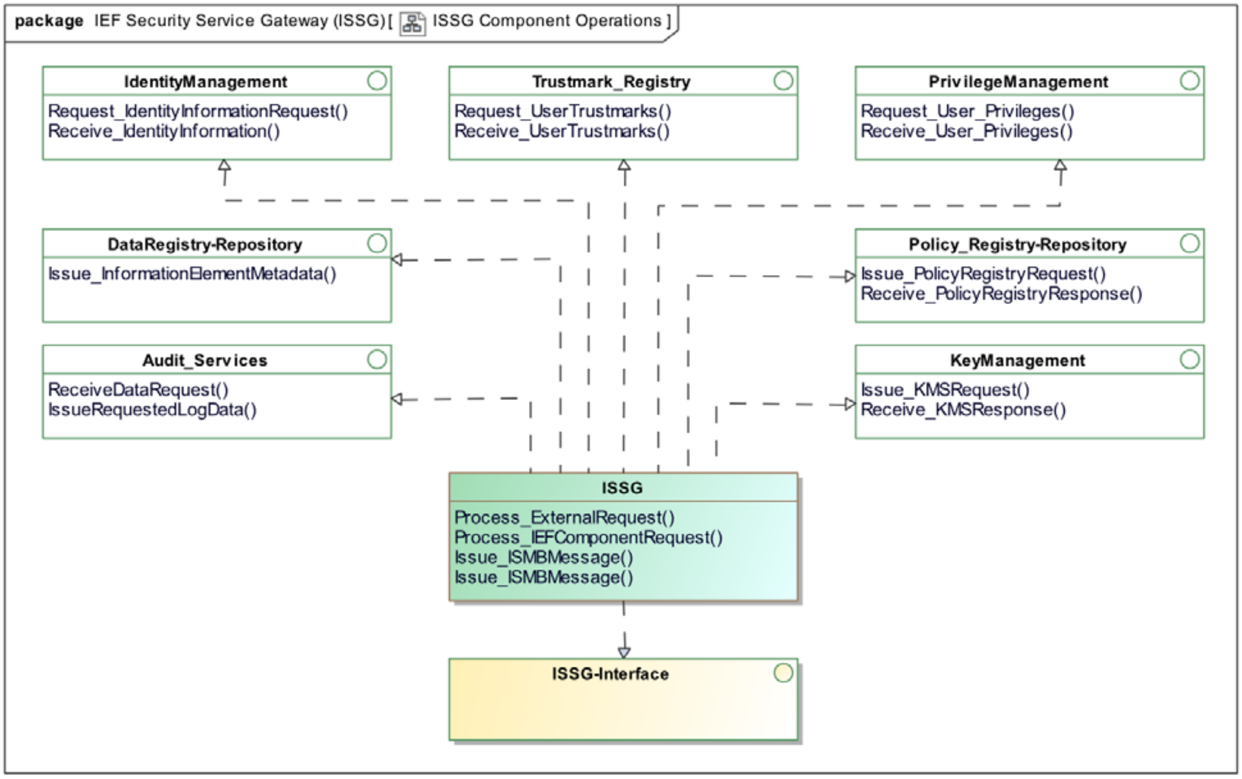

package IEF Security Service Gateway (ISSG)[ISSG Component Operations]
IdentityManagement
Request_IdentityInformationRequest()
Receive_IdentityInformation()
Trustmark_Registry
Request_UserTrustmarks()
Receive_UserTrustmarks()
PrivilegeManagement
Request_User_Privileges()
Receive_User_Privileges()
DataRegistry-Repository
Issue_InformationElementMetadata()
Policy_Registry-Repository
Issue_PolicyRegistryRequest()
Receive_PolicyRegistryResponse()
Audit_Services
ReceiveDataRequest()
IssueRequestedLogData()
KeyManagement
Issue_KMSRequest()
Receive_KMSResponse()
ISSG
Process_ExternalRequest()
Process_IEFComponentRequest()
Issue_ISMBMessage()
Issue_ISMBMessage()
ISSG-Interface

Figure 29 -ISSG Component Operations

The following table identifies and describes the elements and operations illustrated in diagram "ISSG Component Operations".

Table 15 - ISSG Component Operations

Element Name	**Operations**
ISSG	The IEF Security Services Gateway (ISSG) provides a single point for users (vendors and integrators) to integrate IEF components with the users' own security services (e.g., Identity, credential, access-control, and key management) and infrastructure. The ISSG provides the interface to all User specified (delivered) services and infrastructure (e.g., Identity Management).
	Element Type: *Class* ***Owned Operations:*** *Process_ExternalRequest:* The ISSG provides features that receive a resource request from a user (or user service), verifies the user is authorized to access the IEF resource, translates the request into an ISSG-Service Request message and issues the message to the PAP for resolution. For version 1.0 of the specification, these requests are limited to: • Access to PAP situational and context data; and • Access to TLS data. Upon receipt of the information from the IEF component, the ISSG transforms the information into a form that conforms to the users' own environment. *Process_IEFComponentRequest:* The ISSG provides features that receive a resource request from an IEF component, verifies the component is authorized to access the external resource, translates the request into a form that conforms to the users' security service interface and issues the message to the external service for resolution. For version 1.0 of the specification, the ISSG should provide the ability to integrate with the users': • Identity Management Services; • Privilege Management Services; and • Key Management Services. Optional integrations include: • TrustMark Registry; • Data Registry; • Policy-Registry-Repository: ○ Policy Development Environment, and/or ○ Policy Management Environment; and/or

Table 15 - ISSG Component Operations	
Element Name	**Operations**
	- Audit Services. Upon receipt of the information from the Security Service, the ISSG transforms the information into a form that conforms to the IEF defined ISSG-Response message. *Issue_ISMBMessage:* *Issue_ISMBMessage:* ***Inherited Operations:*** *Start_Operations* inherited from *IEF_Component* *Maintain-OperatingState* inherited from *IEF_Component* *Recover_Operations* inherited from *IEF_Component* *Track_RequestResponse* inherited from *IEF_Component* *Authorize_ActionRequest* inherited from *IEF_Component* *Package_AuthorizationRequest* inherited from *IEF_Component* *Package-AdministrativeCommandResponse* inherited from *IEF_Component* *Package_EventLog* inherited from *IEF_Component* *Package_AlertWarningData* inherited from *IEF_Component* *Process_AdministrationCommand* inherited from *IEF_Component* *Configure_Properties* inherited from *IEF_Component* *Archive_Properties* inherited from *IEF_Component*

13 Cryptographic Transformation Services

13.1 Cryptographic Transformation Service (CTS)

The Cryptographic Transformation Service (CTS) performs the actual cryptographic operations in the information assets being protected by the IEF. When an information asset becomes subject to IEF protection, the information is encrypted so that it cannot be subsequently disclosed except through the IEF access control protection. The means by which information assets are protected depends on the nature of the asset itself, for example:

- Data files being copied to a IEF protected file share are encrypted prior to storage;
- Email messages sent through a SAMSON protected mail server are encrypted for storage at the mail server while awaiting delivery;
- Instant messages are stored in encrypted form while hosted at a SAMSON protected chat server.
- Structured messages are encrypted prior to their release to the external user / community messaging services (e.g., AMQP and DDS).

Similarly, when data assets (files, emails, instant messages) are released to an authorized client application, the CTS decrypts the assets prior to delivery. In all cases, it is the PEP that call upon the CTS for cryptographic operations.

The symmetric key to protect an information asset is provided by the PEP that leverages the users own Key Management Services (KMS) to create, store or retrieve the needed key.

When creating encrypted objects, the CTS creates Secure Asset Container (SAC) objects that hold not only the encrypted Information Asset but also object properties including:

- A copy of the security label associated with the original object so that it is not necessary to decrypt the object to acquire the security metadata on the object;
- The key token so that the CTS can retrieve the necessary key to decrypt the object; and
- A hashed value across the object and the security metadata to ensure that the object and/or the security label have not been altered since its original creation.

The operations that can be performed through the CTS include the following:

- Encrypt a file object using a new unique symmetric key and place the new encrypted asset inside a SAC with the key token and supporting properties.
- Decrypt a Secure Asset Container using the symmetric key provided by the PEP from the key token inside the SAC.
- Encrypt a file using a provided key but do not generate a SAC for the encrypted object. It is the responsibility of the calling process to ensure that a key has been created and can be retrieved using a unique key token.
- Decrypt a file using the supplied key. Again, it is the responsibility of the calling process to ensure that the symmetric key can be retrieved with the supplied token.

13.1.1 CTS Component Operations

The Cryptographic-Transformation-Service (CTS) provides the interface between a PEP and the user specified/provided cryptographic application(s), services or appliance(s) that:

- Encrypt information assets; and
- Decrypt information assets.

The CTS must provide the protection capability for at least one of the following data types:

- Files (e.g., data, image, audio, and video) to be encrypted at rest and in transit, and decrypted for use within an authorized client-service;
- E-mail messages, including attachments, to be encrypted when they are stored/prepared for delivery and decrypted when a user accesses the e-mail using an authorized e-mail-client.
- IM messages to be encrypted as they are exchanged by an IM server. IM messages are decrypted when viewed through an authorized IM-client.
- Structured payloads (messages, including attachments) to be encrypted for storage and transmission, and decrypted for use within an authorized client-service.

The CTS returns an error code to the PEP if any of the following conditions are encountered:

- Message cannot be parsed due to a malformed request;
- No action element was specified in the request or the action is unsupported;
- The source file for the operation cannot be accessed; or
- An error occurred in the external Key Management services.

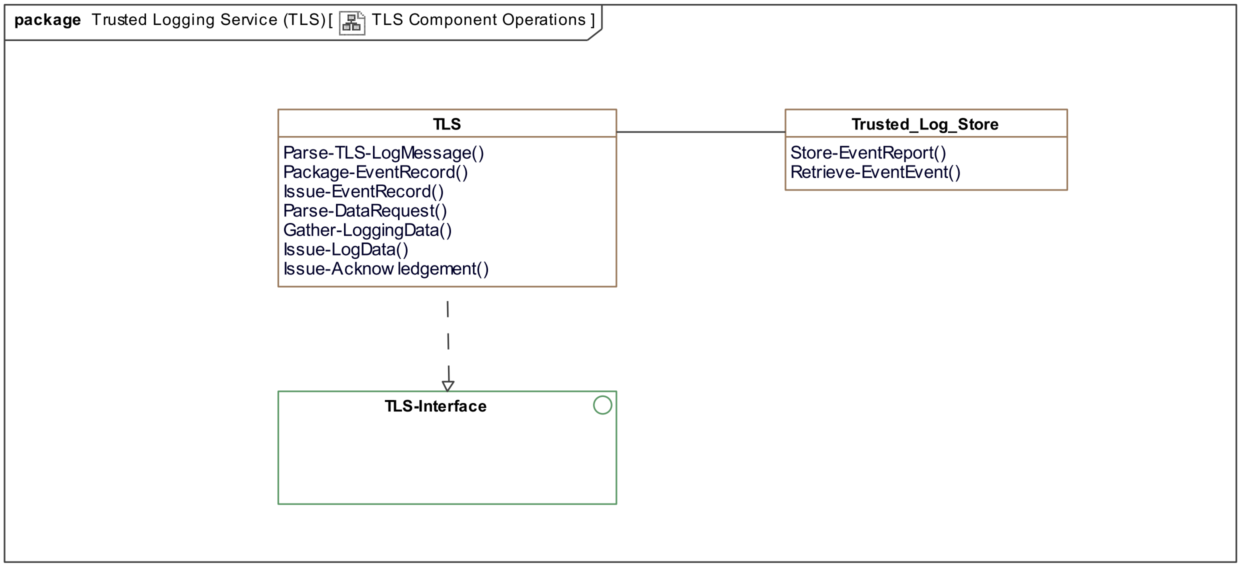


Figure 30 -CTS Component Operations

The following table identifies and describes the elements and operations illustrated in diagram "CTS Component Operations".

Table 16 - CTS Component Operations

Element Name	Operations
Cryptographic_Service	This reference architecture illustrates the direct integration of cryptographic services into the IEF CTS service offering. CTS services may also be accessed through the ISSG if the services are maintained elsewhere in the user infrastructure. User specified, FIPS-complaint software module, service or appliance that is integrated into the CTS to provide cryptographic transformations on IEF protected information elements. These services provide encryption/decryption of data and information elements managed in an IEF environment. When data and information elements become subject to IEF protection, they are individually encrypted using a symmetric key. Encrypted information elements carry a "key Token" provisioned by the Key Escrow service. This token is used by authorized services (/authorized PEP) to request the key from the escrow service.
	Element Type: *Class* ***Owned Operations:*** *Encrypt_InformationElement:* The Cryptographic service provides features that transform an InformationElement into an unintelligible form using the CryptographicKey provided by the PEP as part of the CTS-Request message. *Decrypt_InformationElement:* User specified Cryptographic service features that transform the information content of an information element back into its original form using the unique symmetric key retrieved from escrow by the PEP as part of the CTS-request message.
CTS	The Cryptographic Transformation Service (CTS) is the IEF component that encrypts and decrypts InformationElements as authorized by policy. The CTS is a bridging component that will link cryptographic action requests from the PEP to a FIPS-complaint software module for execution.
	Element Type: *Class* ***Owned Operations:*** *ProcessTransformationRequest:* The CTS provides features that engage the user specified cryptographic services to encrypt or decrypt the PEP provided Information element. *Package_CTS-ResponseMessage:*

Table 16 - CTS Component Operations

Element Name	**Operations**
	The CTS provides features that gather the transformed InformationElement and formats the element into a CTS-Response message. *PackageSAC:* The CTS provides features that gather and process the required Secure Access Container (SAC) elements and packages the SAC containing the encrypted information element and Envelope Header. *ProcessSAC:* The CTS provides features that extract and verify the SAC InformationElements has not been tampered with or modified, and extract the informationElements for processing. *Package_SecurityServiceRequest:* The PEP provides features that package ServiceRequestData as an ISMB-Message directed to ISSG in order to request information from the user's security infrastructure: Request Cryptographic Key / Token from the user specified Key Management system or service. *Parse_SecurtyServiceResponse:* The PEP provides features that parse an ISSG-Response message and extract the data elements required by the PEP enforcement features. ServiceResponseData includes: • CryptographicKey and KeyToken; • SenderIdentityInformation; • ReceiverIdentityInformation; • SenderPrivileges; • ReceiverPrivileges ; and • OperationalContextData. ***Inherited Operations:*** *Start_Operations* inherited from *IEF_Component* *Maintain-OperatingState* inherited from *IEF_Component* *Recover_Operations* inherited from *IEF_Component* *Track_RequestResponse* inherited from *IEF_Component* *Authorize_ActionRequest* inherited from *IEF_Component* *Package_AuthorizationRequest* inherited from *IEF_Component*

Table 16 - CTS Component Operations

Element Name	**Operations**
	Package-AdministrativeCommandResponse inherited from *IEF_Component* *Package_EventLog* inherited from *IEF_Component* *Package_AlertWarningData* inherited from *IEF_Component* *Process_AdministrationCommand* inherited from *IEF_Component* *Configure_Properties* inherited from *IEF_Component* *Archive_Properties* inherited from *IEF_Component*

14 Secure Messaging Bus

14.1 IEF Secure Messaging Bus (ISMB)

The IEF is specified as a service-oriented architecture that uses standardized messaging to provide a set of interconnected services that provide policy-driven data-centric information sharing and safeguarding services for file-shares, email, instant messaging and structured messaging.

The information exchanges between services utilize industry accepted, open standards that are based on XML. It is the responsibility of the IEF Secure Messaging Bus (ISMB) to securely deliver these messages between IEF components. Although the specific protocol or format of the message content depends on the nature of the service being used, all messages are delivered through the same communications mechanism. The ISMB has responsibility for providing a robust, secure and trusted delivery of security messages between IEF components. The messaging infrastructure forms the critical core of the IEF architecture.

Technology demonstration projects (TDP) have been conducted using two different standards-based messaging solutions:

- XMPP network integrating IEF components that deliver IEF services (components) for file-share, email and Instant (Text) Messaging. This TDP was conducted by Defence Research and Development Canada.
- DDS network integrating IEF components that deliver IEF services for structured messaging for standardized canonical models (e.g., CAP, NIEM, and MIEM). This TDP was conducted by Shared Services Canada (SSC) and the Centre for Security Sciences (CSS).

The results of these two TDPs form the basis for this Policy-drive Data-centric information and safeguarding architecture.

15 Trusted Logging Service

15.1 Trusted Logging Service (TLS)

The Trusted Logging Service (TLS) persists event/transaction reports from the IEF components in order to enable both short-term security incident and event monitoring and reporting, and long-term forensic auditing.

Transaction logging is a feature of every IEF component: PAP; PDP; PEP; PPS; Secure Messaging Bus; and Security Services Gateway.

The level of logging to be performed is set by a configuration parameter within the PAP profile (configuration file) based on the user defined resource utilization and performance specifications. The level of logging may be set for each component, or for the environment as a whole.

15.2 TLS Component Operations

The following figure identified the core features and functions provided by a Trusted Logging Service.

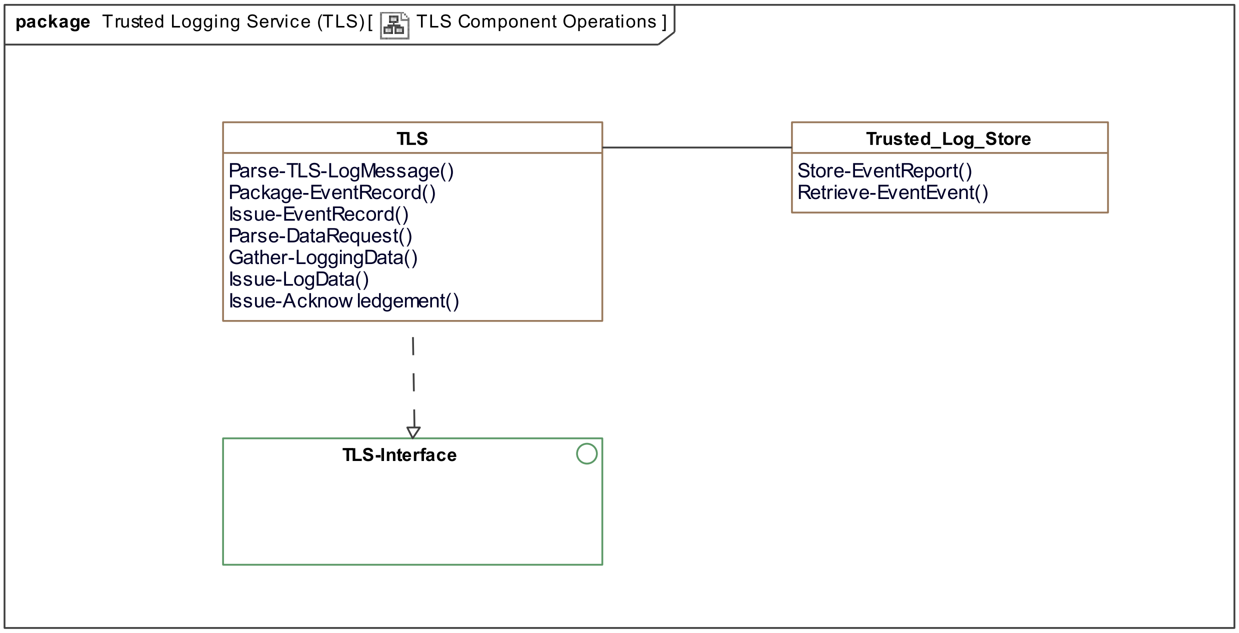


Figure 31 -TLS Component Operations

The following table identifies and describes the elements and operations illustrated in diagram “TLS Component Operations”.

Table 17 - TLS Component Operations

Element Name	**Operations**
TLS	The Trusted Logging Service (TLS) securely records IEF component activity as a transactional history of the policy decisions and access control enforcement. The information stored by the TLS can be used to actively monitor IEF activity, or as a forensic audit of the information sharing conducted as part of a mission or operation.
	Element Type: *Class* ***Owned Operations:*** *Parse-TLS-LogMessage:* The TLS provides features that parse an IEF Component Log Report. *Package-EventRecord:* The TLS provides features that package an event record to be stored in the data store. *Issue-EventRecord:* The TLS provides features that issue the Event Record to the data store. *Parse-DataRequest:* The TLS provides features that parse a data request from the user specified monitoring or auditing services. *Gather-LoggingData:* The TLS provides features that gather requested log data from the data store. *Issue-LogData:* The TLS provides features that issue log data to a requesting auditing or monitoring service. *Issue-Acknowledgement:* The TLS provides features that package and issue an acknowledgment to a component logging an event.
Trusted_Log_Store	A special-purpose database designed to provide a tamper-resistant record of IEF events and transactions (e.g., policy decision activities) as recorded by the Trusted Logging Service in a manner that enables both real-time Security Incident and Event Monitoring and longer-term forensic activities.

Table 17 - TLS Component Operations

Element Name	**Operations**
	Element Type: *Class* ***Owned Operations:*** *Store-EventReport:* The Trusted Log store provides features that write a record to a tamper resistant physical data store. *Retrieve-EventEvent:* The Trusted Log store provides features that retrieve a record to a tamper resistant physical data store.

16 ISMB Messages

The following clauses describe the messages used by the IEF components to communicate over the Secure Messaging Bus (ISMB). XSDs for these messages are provided in Annex A.

16.1 ISMB Message Attributes

This clause identifies and describes the core attributes included in all ISMB-Messages.

Table 18 - ISMB Message Attributes

Element Name	**Attributes**
ISMB-Message	Messages passed between IEF elements across the Secure Messaging Bus (ISMB) to direct and report on IEF operations. ***Owned Attributes:*** *Issue-DateTime* (type: String) [1..1]: Identifies when the Message was issued. *MessageID* (type: OctetSeq) [1..1]: Unique identifier for the message. *MessageSourceID* (type: OctetSeq) [1..1]: Unique identifier for the component issuing the message. *MessageTargetID* (type: OctetSeq) [1..1]: Unique identifier for the component expected to receive the message. *MessageType* (type: ISMB-MessageType) [1..1]: Identifies the type of message in order for the receiving component to initiate the appropriate processing sequence. *ProcessStatus* (type: OperationStatusType) [1..1]: Returns the operational status of the component.

16.2 ISMB Messages

The following figure identifies the messages and the core metadata elements exchanged by each of these messages. The prefix to the message identifies the primary actor in the message exchange.

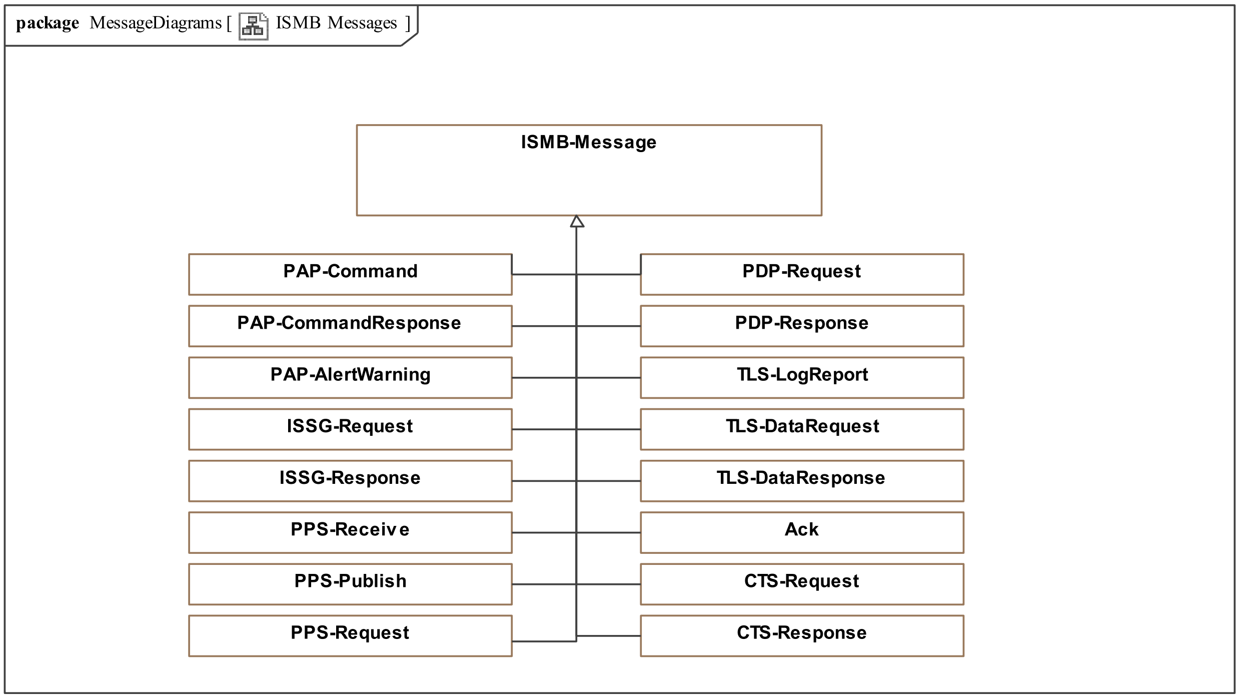


Figure 32 -ISMB Messages

The following table describes the message elements depicted in diagram “ISMB Messages”.

Table 19 - ISMB Messages Attributes	
Element Name	**Attributes**
Ack	An ISMB-Message from a message recipient to the message sender reporting that a specific message was received.
CTS-Request	An ISMB-message to the CTS that requests the transformation (encryption or decryption) of the specified InformationElement.
CTS-Response	An ISMB-message from a CTS to an IEF component responding to a cryptographic transformation request.
ISMB-Message	Messages passed between IEF elements across the Secure Messaging Bus (ISMB) to direct and report on IEF operations.

ISSG-Request	An *ISMB-message* issued by an IEF component to the **ISSG** requesting data / information from a user specified and provisioned security service, e.g.: • Identity Management; • Privilege (/attribute / authorization) Management; • Cryptographic Key Management; or • TrustMark Registry.
ISSG-Response	An ISMB-message from the ISSG to the IEF Component providing the requested data or Information elements.
PAP-AlertWarning	An *ISMB-Message* from an IEF Component to the **PAP** (and the **TLS**) that informs the user (/administrator) of an unauthorized request to access or release information, unauthorized request to perform an operation, or other error conditions that are being generated through these requests.
PAP-Command	An ISMB-Message from the PAP to an IEF component directing the component to perform a specified operation or action.
PAP-CommandResponse	An ISMB-Message from an IEF component to the PAP providing the results to a PAP's command message.
PDP-Request	An ISMB-Message from an IEF component to a PDP requesting authorization to perform a specific action, on specified InformationElement(s), targeting a specified set of recipients.
PDP-Response	An ISMB-Message from a PDP to a PEP providing the result(s) of its policy adjudication.
PPS-Publish	An ISMB-message from a PPS to a Messaging-PEP directing the PEP to validate and verify that the PPS is authorized to issue the InformationElement(s) and that the specified recipient(s) are authorized to receive and access the enclosed content. The message also provides direction on the communication channels and protocols to be applied.
PPS-Receive	An ISMB-message from a Messaging-PEP to a PPS forwarding authorized InformationElements from the integrated messaging services or middleware.
PPS-Request	An ISMB-message from a PEP to the PPS to request the release of information based on a specified SemanticElement. The request contains the base (unique) identifier for the information elements being requested, and the identifier for semantic policies to be applied.

TLS-DataRequest	An ISMB-Message issued by a User Monitoring Application or Auditing Application to the TLS requesting a log entry or set of log entries for an IEF component.
TLS-DataResponse	An ISMB-Message issued from the TLS to the User Auditing or Monitoring Application.
TLS-LogReport	An ISMB-Message issued by an IEF component to the Trusted Logging Service(s) describing: • Operations on InformationElements protected by the IEF; • Changes to the operating characteristics of an IEF Component; and • Changes to the Data Policies or Access & Release Control policies. These messages enable the tamper-resistant recording of IEF operations. The log(s) supports both Security Incident and Event Monitoring (SIEM) and forensic auditing of the environment. Each IEF transaction must be recorded in a manner that resists tampering and alteration of the log records. The log is intended to maintain a chain-of-custody record for all information elements protected by an IEF implementation. Each log record is encrypted in motion and storage and assigned a chained digital signature.

16.3 ISMB Message Details

The following clauses identify and describe the attributes for each of the IMSB Messages.

16.3.1 PAP Command Message

The following figure identifies the data and information elements issued by the PAP to manage and administer the operations of IEF components.

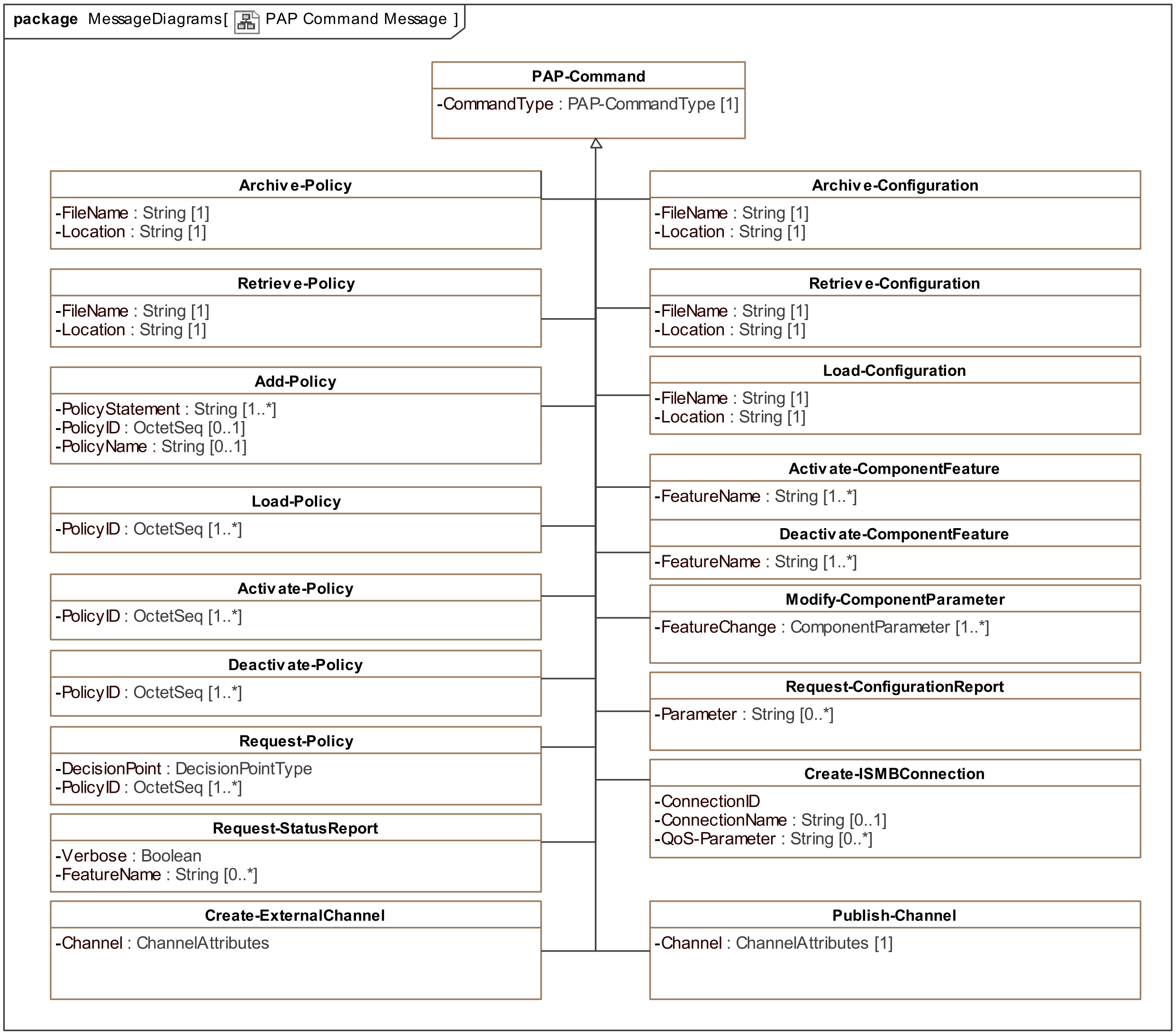

package MessageDiagrams[PAP Command Message]
PAP-Command
-CommandType : PAP-CommandType [1]
Archive-Policy
-FileName : String [1]
-Location : String [1]
Retrieve-Policy
-FileName : String [1]
-Location : String [1]
Add-Policy
-PolicyStatement : String [1..*]
-PolicyID : OctetSeq [0..1]
-PolicyName : String [0..1]
Load-Policy
-PolicyID : OctetSeq [1..*]
Activate-Policy
-PolicyID : OctetSeq [1..*]
Deactivate-Policy
-PolicyID : OctetSeq [1..*]
Request-Policy
-DecisionPoint : DecisionPointType
-PolicyID : OctetSeq [1..*]
Request-StatusReport
-Verbose : Boolean
-FeatureName : String [0..*]
Create-ExternalChannel
-Channel : ChannelAttributes
Archive-Configuration
-FileName : String [1]
-Location : String [1]
Retrieve-Configuration
-FileName : String [1]
-Location : String [1]
Load-Configuration
-FileName : String [1]
-Location : String [1]
Activate-ComponentFeature
-FeatureName : String [1..*]
Deactivate-ComponentFeature
-FeatureName : String [1..*]
Modify-ComponentParameter
-FeatureChange : ComponentParameter [1..*]
Request-ConfigurationReport
-Parameter : String [0..*]
Create-ISMBConnection
-ConnectionID
-ConnectionName : String [0..1]
-QoS-Parameter : String [0..*]
Publish-Channel
-Channel : ChannelAttributes [1]

Figure 33 -PAP Command Message

The following table describes the message elements depicted in diagram “PAP Command Message”.

Table 20 - PAP Command Message Attributes

Element Name	**Attributes**
Activate-Policy	ISMB-Message from a PAP that directs an IEF Component (i.e., PDP and PPS) to change the state of a specified set of policies in its environment from inactive to active. ***Owned Attributes:*** *PolicyID* (type: OctetSeq) [1..*]: Unique identifier for the policy statement. **Inherited Attributes:** *CommandType* inherited from *PAP-Command* *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message*
Activate-_ComponentFeature	ISMB-Message from a PAP that directs an IEF Component to activate one or more of its features. ***Owned Attributes:*** *FeatureName* (type: String) [1..*]: Name of the feature to be Activated. "All" directs the component to activate itself and/or all internal features. **Inherited Attributes:** *CommandType* inherited from *PAP-Command* *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message*

Add-Policy	ISMB-Message from a PAP that directs an IEF Component (i.e., **PDP** or **PPS**) to add the policies in the message or from a specified file to its policy environment. These policies are held in a temporary processing area until the user (/administrator) directs the component to load the policies. ***Owned Attributes:*** *PolicyID* (type: OctetSeq) [0..1]: Unique identifier for the policy statement. *PolicyName* (type: String) [0..1]: Name of the policy statement. *PolicyStatement* (type: String) [1..*]: Executable expression of the policy. **Inherited Attributes:** *CommandType* inherited from *PAP-Command* *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message*
Archive-Configuration	ISMB-Message issued by the PAP that directs an IEF component to persist its current operating configuration to a specified location. ***Owned Attributes:*** *FileName* (type: String) [1..1]: Name of the file that contains the archived configuration. *Location* (type: String) [1..1]: Location of a file within the IEF persistent store. **Inherited Attributes:** *CommandType* inherited from *PAP-Command* *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *q* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message*

Archive-Policy	ISMB-Message issued by the PAP that directs an IEF component (i.e., PPS and PDP) to persist its current policies to a file in the specified location. The policy file (archive) stores as a SecureAssetContainer at the specified location. ***Owned Attributes:*** *FileName* (type: String) [1..1]: Name of the file that contains the archived Policy. *Location* (type: String) [1..1]: Location of a file within the IEF persistent store. **Inherited Attributes:** *CommandType* inherited from *PAP-Command* *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message*
Create-ExternalChannel	ISMB-Message issued by the PAP that directs a Messaging-**PEP** to create a connection with the user specified messaging infrastructure. ***Owned Attributes:*** *Channel* (type: ChannelAttributes) [1..1]: Unique identifier for the information exchange connection. **Inherited Attributes:** *CommandType* inherited from *PAP-Command* *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message*

Create-ISMBConnection	ISMB-Message issued by the PAP that directs an IEF component to establish and register a connection to the ISMB. ***Owned Attributes:*** *ConnectionID* (type:) [1..1]: Unique identifier for the information exchange connection on the ISMB. *ConnectionName* (type: String) [0..1]: Human readable name for the information sharing connection on the ISMB. *QoS-Parameter* (type: String) [0..*]: Quality of Service Parameter for the connection of the ISMB. Provided as "ParameterName: ParameterValue". **Inherited Attributes:** *CommandType* inherited from *PAP-Command* *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message*
Deactivate-Policy	ISMB-Message from a PAP that directs an IEF Component (i.e., **PDP** and **PPS**) to change the state of a specified set of policies in its environment from active to inactive. ***Owned Attributes:*** *PolicyID* (type: OctetSeq) [1..*]: Unique identifier for the policy statement. **Inherited Attributes:** *CommandType* inherited from *PAP-Command* *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message*

Deactivate-ComponentFeature	ISMB-Message from a PAP that directs an IEF Component to deactivate one or more component features. ***Owned Attributes:*** *FeatureName* (type: String) [1..*]: Name of the feature to be deactivated. "All" directs the component to deactivate itself and/or all internal features. **Inherited Attributes:** *CommandType* inherited from *PAP-Command* *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message*
Load-Configuration	ISMB-Message issued by the PAP that directs an IEF component to load (reset) parameters held in its processing area to its operational settings. ***Owned Attributes:*** *FileName* (type: String) [1..1]: Name of the file that contains the configuration. *Location* (type: String) [1..1]: Location of a file containing the configuration. **Inherited Attributes:** *CommandType* inherited from *PAP-Command* *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message*
Load-Policy	ISMB-Message issued by the PAP that directs a PDP or a PPS to load one or more policies from its processing area to its policy environment. ***Owned Attributes:*** *PolicyID* (type: OctetSeq) [1..*]: Unique identifier for the policy statement.

	Inherited Attributes: *CommandType* inherited from *PAP-Command* *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message*
Modify-ComponentParameter	ISMB-Message from a PAP that directs an IEF component to change the value of one or more of its configuration parameters. ***Owned Attributes:*** *FeatureChange* (type: ComponentParameter) [1..*]: **Inherited Attributes:** *CommandType* inherited from *PAP-Command* *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message*
PAP-Command	An ISMB-Message from the PAP to an IEF component directing the component to perform a specified operation or action. ***Owned Attributes:*** *CommandType* (type: PAP-CommandType) [1..1]: Type of command being issued. **Inherited Attributes:** *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message*

Publish-Channel	***Owned Attributes:*** *Channel* (type: ChannelAttributes) [1..1]: **Inherited Attributes:** *CommandType* inherited from *PAP-Command* *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message*
Request-ConfigurationReport	ISMB-Message from a PAP that directs an IEF component to report on one or more of its operating characteristics (/parameters/properties) to the PAP. ***Owned Attributes:*** *Parameter* (type: String) [0..*]: Identifies the parameter (and value) to be included in the report. "All" directs the component to report on all operating parameters. **Inherited Attributes:** *CommandType* inherited from *PAP-Command* *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message*
Request-Policy	ISMB-Message from a PAP that directs an IEF component (i.e., **PPS** or **PDP**) to provide information on one or more policies. ***Owned Attributes:*** *DecisionPoint* (type: DecisionPointType) [1..1]: Identifies the type of Decision Point. *PolicyID* (type: OctetSeq) [1..*]: Unique identifier for the policy or a set of policies to be reported on. **Inherited Attributes:** *CommandType* inherited from *PAP-Command* *Issue-DateTime* inherited from *ISMB-Message*

	MessageID inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message*
Request-StatusReport	ISMB-Message from a PAP that directs an IEF Component to report the current operating status of one or more of its features to the PAP. ***Owned Attributes:*** *FeatureName* (type: String) [0..*]: Name of the feature to be reported on. "All" refers to all features and sub-features. *Verbose* (type: Boolean) [1..1]: Indicates whether or not a verbose status report is being requested. A verbose report would identify the operating status of each feature and sub-feature individually. The basic report provides an overall status for the component. **Inherited Attributes:** *CommandType* inherited from *PAP-Command* *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message*
Retrieve-Configuration	ISMB-Message issued by the PAP that directs an IEF component to get a configuration file (or archive) from a specified location in the IEF environment. The component extracts the configuration file from the SAC, ingest the configuration policies environment and wait for the instruction to activate the configuration. ***Owned Attributes:*** *FileName* (type: String) [1..1]: Name of the file that contains the archived configuration. *Location* (type: String) [1..1]: Location of a file within the IEF persistent store. **Inherited Attributes:** *CommandType* inherited from *PAP-Command*

	Issue-DateTime inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message*
Retrieve-Policy	ISMB-Message issued by the PAP that directs an IEF component (i.e., PPS and PDP) to get a set of policies from a specified file stored at a specified location in the IEF environment. The component extracts the policy file from the SAC, ingest the policies environment and wait for the instruction to activate the policies. ***Owned Attributes:*** *FileName* (type: String) [1..1]: Name of the file that contains the archived configuration. *Location* (type: String) [1..1]: Location of a file within the IEF persistent store. **Inherited Attributes:** *CommandType* inherited from *PAP-Command* *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message*

16.3.2 PAP Command Response Message

The following figure identifies the data and information elements issued by an IEF component to the PAP in response to a PAP-Command.

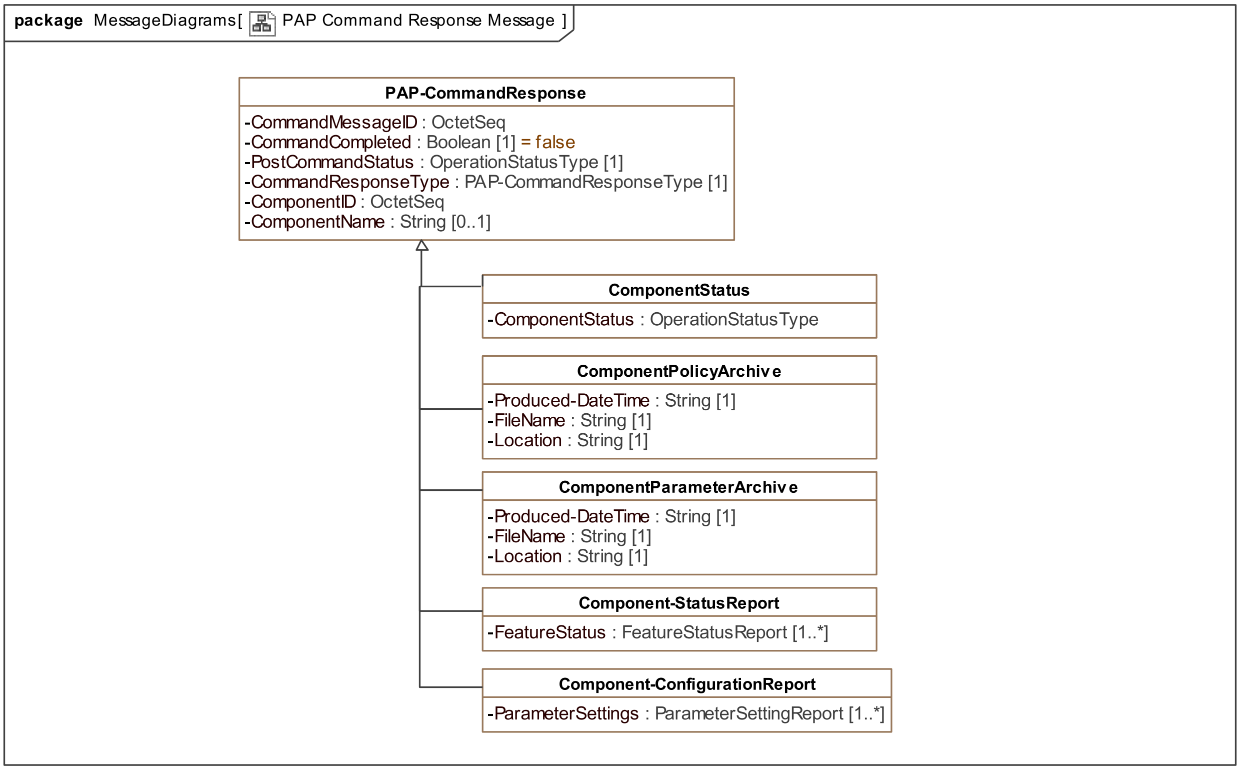

package MessageDiagrams[PAP Command Response Message]
PAP-CommandResponse
-CommandMessageID : OctetSeq
-CommandCompleted : Boolean [1] = false
-PostCommandStatus : OperationStatusType [1]
-CommandResponseType : PAP-CommandResponseType [1]
-ComponentID : OctetSeq
-ComponentName : String [0..1]
ComponentStatus
-ComponentStatus : OperationStatusType
ComponentPolicyArchive
-Produced-DateTime : String [1]
-FileName : String [1]
-Location : String [1]
ComponentParameterArchive
-Produced-DateTime : String [1]
-FileName : String [1]
-Location : String [1]
Component-StatusReport
-FeatureStatus : FeatureStatusReport [1..*]
Component-ConfigurationReport
-ParameterSettings : ParameterSettingReport [1..*]

Figure 34 -PAP Command Response Message

The following table describes the message elements depicted in diagram "PAP Command Response Message".

Table 21 - PAP Command Response Message Attributes	
Element Name	**Attributes**
Component-ConfigurationReport	ISMB-Message that contains the IEF Component configuration for the **PAP**. ***Owned Attributes:*** *ParameterSettings* (type: ParameterSettingReport) [1..*]: Array of Parameter settings. **Inherited Attributes:** *CommandCompleted* inherited from *PAP-CommandResponse* *CommandMessageID* inherited from *PAP-CommandResponse* *CommandResponseType* inherited from *PAP-CommandResponse*

	ComponentID inherited from *PAP-CommandResponse* *ComponentName* inherited from *PAP-CommandResponse* *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *PostCommandStatus* inherited from *PAP-CommandResponse* *ProcessStatus* inherited from *ISMB-Message*
Component-StatusReport	ISMB-Message that contains a reporting of the current status of the overall component and each of its features. ***Owned Attributes:*** *FeatureStatus* (type: FeatureStatusReport) [1..*]: Array of Feature Status values. **Inherited Attributes:** *CommandCompleted* inherited from *PAP-CommandResponse* *CommandMessageID* inherited from *PAP-CommandResponse* *CommandResponseType* inherited from *PAP-CommandResponse* *ComponentID* inherited from *PAP-CommandResponse* *ComponentName* inherited from *PAP-CommandResponse* *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *PostCommandStatus* inherited from *PAP-CommandResponse* *ProcessStatus* inherited from *ISMB-Message*
ComponentParameterArchive	ISMB-Message that contains the name and location of the file containing the current component configuration parameters. ***Owned Attributes:*** *FileName* (type: String) [1..1]: Name of the file that contains the archived configuration. *Location* (type: String) [1..1]: Location of the file within the IEF persistent store.

	Produced-DateTime (type: String) [1..1]: Date and time the parameter set was generated. **Inherited Attributes:** *CommandCompleted* inherited from *PAP-CommandResponse* *CommandMessageID* inherited from *PAP-CommandResponse* *CommandResponseType* inherited from *PAP-CommandResponse* *ComponentID* inherited from *PAP-CommandResponse* *ComponentName* inherited from *PAP-CommandResponse* *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *PostCommandStatus* inherited from *PAP-CommandResponse* *ProcessStatus* inherited from *ISMB-Message*
ComponentPolicyArchive	ISMB-Message issued by an IEF component to the PAP that contains the name and location of a file containing the current set of component policies. Applicable to the PDP and PPS. ***Owned Attributes:*** *FileName* (type: String) [1..1]: Name of the file that contains the archived policies. *Location* (type: String) [1..1]: Location of the file within the IEF persistent store. *Produced-DateTime* (type: String) [1..1]: Date and time the policy set was generated. **Inherited Attributes:** *CommandCompleted* inherited from *PAP-CommandResponse* *CommandMessageID* inherited from *PAP-CommandResponse* *CommandResponseType* inherited from *PAP-CommandResponse* *ComponentID* inherited from *PAP-CommandResponse* *ComponentName* inherited from *PAP-CommandResponse* *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message*

	MessageType inherited from *ISMB-Message* *PostCommandStatus* inherited from *PAP-CommandResponse* *ProcessStatus* inherited from *ISMB-Message*
ComponentStatus	ISMB-Message issued by an IEF component to the PAP that contains the overall status of the component. ***Owned Attributes:*** *ComponentStatus* (type: OperationStatusType) [1..1]: The overall status of the component. **Inherited Attributes:** *CommandCompleted* inherited from *PAP-CommandResponse* *CommandMessageID* inherited from *PAP-CommandResponse* *CommandResponseType* inherited from *PAP-CommandResponse* *ComponentID* inherited from *PAP-CommandResponse* *ComponentName* inherited from *PAP-CommandResponse* *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *PostCommandStatus* inherited from *PAP-CommandResponse* *ProcessStatus* inherited from *ISMB-Message*
PAP-CommandResponse	An ISMB-Message from an IEF component to the PAP providing the results to a PAP's command message. ***Owned Attributes:*** *CommandCompleted* (type: Boolean) [1..1]: Indicates whether or not the command executed correctly. *CommandMessageID* (type: OctetSeq) [1..1]: Unique identifier for the PAP-CommandMessage resulting in this response. *CommandResponseType* (type: PAP-CommandResponseType) [1..1]: Identifies the message type being issued. *ComponentID* (type: OctetSeq) [1..1]:

	The identifier of the responding IEF component. *ComponentName* (type: String) [0..1]: Human readable name for the IEF component issuing the message. *PostCommandStatus* (type: OperationStatusType) [1..1]: Provides the post command operating status of the IEF component. **Inherited Attributes:** *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message*

16.3.3 PAP AlertWarning Message

The following figure identifies the data and information elements issued by an IEF component to provide operational alerts, warnings or error conditions to the PAP (i.e., IEF Administrator). PAP-AlertWarning messages are also sent to the TLS as a record of the event.

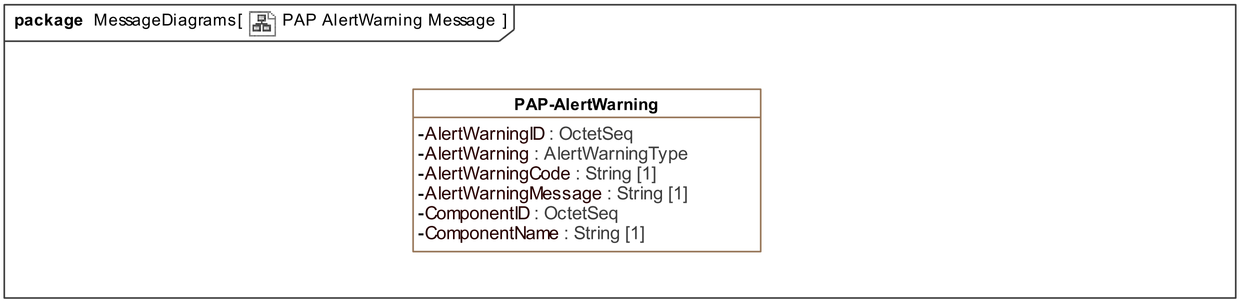

package MessageDiagrams[PAP AlertWarning Message]
PAP-AlertWarning
-AlertWarningID : OctetSeq
-AlertWarning : AlertWarningType
-AlertWarningCode : String [1]
-AlertWarningMessage : String [1]
-ComponentID : OctetSeq
-ComponentName : String [1]

Figure 35 -PAP AlertWarning Message

The following table describes the message elements depicted in diagram "PAP AlertWarning Message".

Table 22 - PAP AlertWarning Message Attributes

Element Name	Attributes
PAP-AlertWarning	An *ISMB-Message* from an IEF Component to the **PAP** (and the **TLS**) that informs the user (/administrator) of an unauthorized request to access or release information, unauthorized request to perform an operation, or other error conditions that are being generated through these requests. ***Owned Attributes:*** *AlertWarning* (type: AlertWarningType) [1..1]: Type of AlertWarning being sent. *AlertWarningCode* (type: String) [1..1]: Unique identifier for the Alert Warning Message. *AlertWarningID* (type: OctetSeq) [1..1]: Unique identifier for the AlertWarning. *AlertWarningMessage* (type: String) [1..1]: Human readable text for the AlertWarning Message. *ComponentID* (type: OctetSeq) [1..1]: Unique identifier for the component sending the AlertWarning Message. *ComponentName* (type: String) [1..1]: Name of the component sending the AlertWarning Message. **Inherited Attributes:** *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message*

16.3.4 PDP Request Message

PDP Request Message conforms to a XACML request message (http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.html).

16.3.5 PDP Response Message

The PDP Response Message conforms to a XACML response message (http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.html).

16.3.6 PPS Receive Message

The following figure identifies the data and information elements issued by a Messaging-PEP to a PPS in order to transfer InformationElements for processing and marshaling.

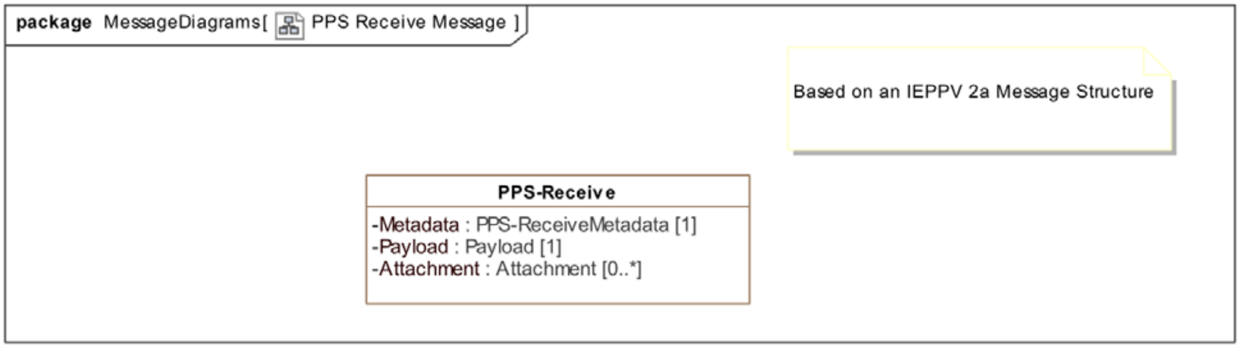

package MessageDiagrams[PPS Receive Message]
Based on an IEPPV 2a Message Structure
PPS-Receive
-Metadata : PPS-ReceiveMetadata [1]
-Payload : Payload [1]
-Attachment : Attachment [0..*]

Figure 36 -PPS Receive Message

The following table describes the message elements depicted in diagram "PPS Receive Message".

Table 23 - PPS Receive Message Attributes

Element Name	Attributes
PPS-Receive	An ISMB-message from a Messaging-PEP to a PPS forwarding authorized InformationElements from the integrated messaging services or middleware. ***Owned Attributes:*** *Attachment* (type: Attachment) [0..*]: *Metadata* (type: PPS-ReceiveMetadata) [1..1]: *Payload* (type: Payload) [1..1]: **Inherited Attributes:** *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message*

16.3.7 PPS Publish Message

The following figure identifies the data and information elements issued by the PPS to the Messaging-PEP requesting the release (dissemination) of the included InformationElements to a communication channel or specified group of recipients.

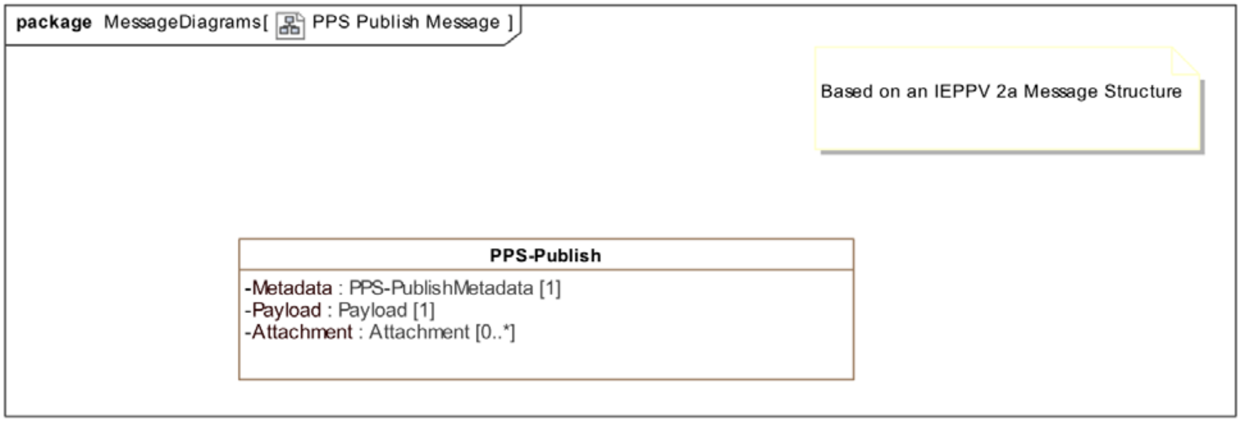

package MessageDiagrams[PPS Publish Message]
Based on an IEPPV 2a Message Structure
PPS-Publish
-Metadata : PPS-PublishMetadata [1]
-Payload : Payload [1]
-Attachment : Attachment [0..*]

Figure 37 -PPS Publish Message

The following table describes the message elements depicted in diagram "PPS Publish Message".

Table 24 - PPS Publish Message Attributes	
Element Name	**Attributes**
PPS-Publish	An ISMB-message from a PPS to a Messaging-PEP directing the PEP to validate and verify that the PPS is authorized to issue the InformationElement(s) and that the specified recipient(s) are authorized to receive and access the enclosed content. The message also provides direction on the communication channels and protocols to be applied. ***Owned Attributes:*** *Attachment* (type: Attachment) [0..*]: *Metadata* (type: PPS-PublishMetadata) [1..1]: *Payload* (type: Payload) [1..1]: **Inherited Attributes:** *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message*

	MessageTargetID inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message*

16.3.8 PPS Request Message

The following figure identifies the data and information elements sent to the PPS to request the release (dissemination) of information to the requesting user.

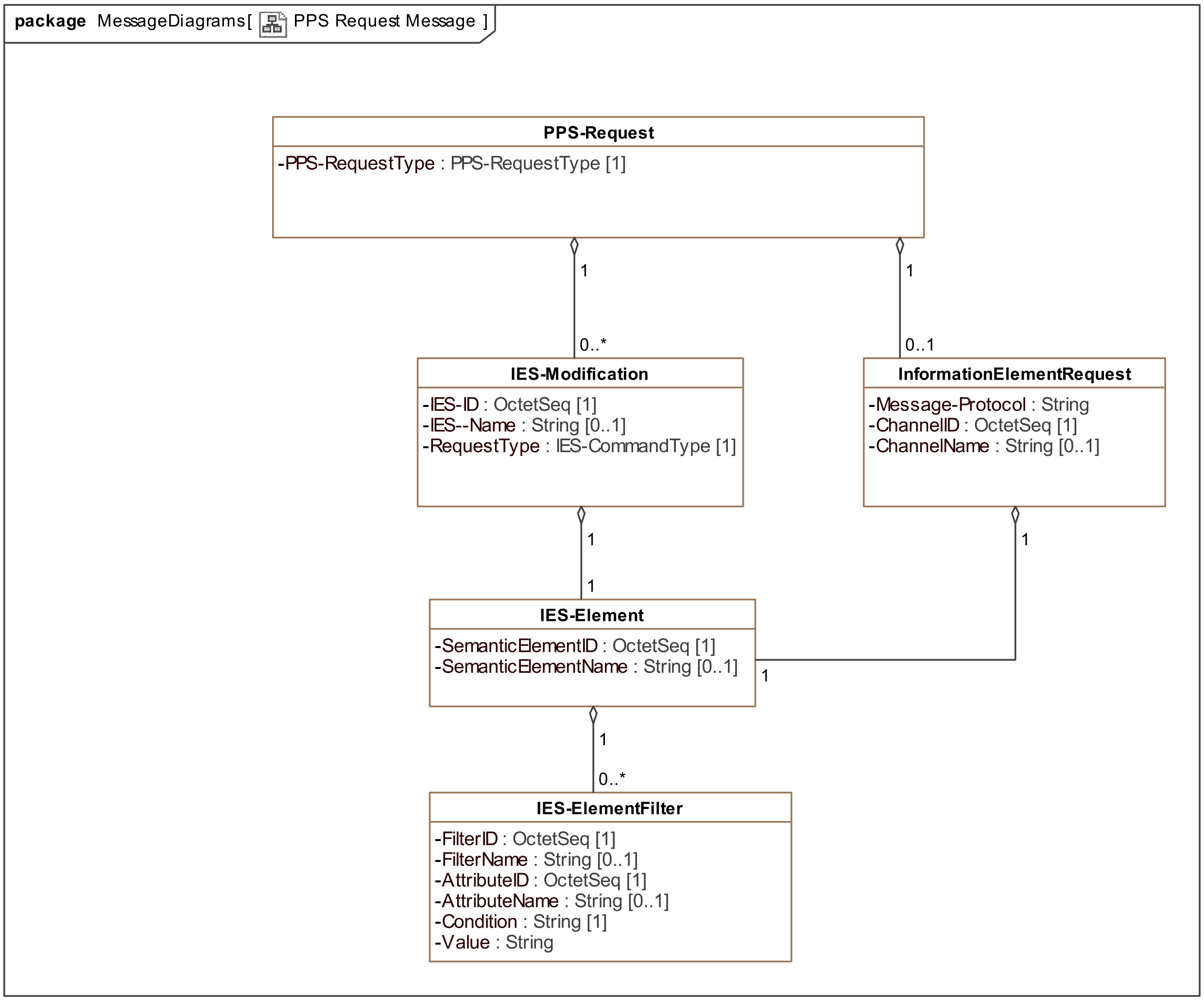

package MessageDiagrams[PPS Request Message]
PPS-Request
-PPS-RequestType : PPS-RequestType [1]
1
0..*
1
0..1
IES-Modification
-IES-ID : OctetSeq [1]
-IES--Name : String [0..1]
-RequestType : IES-CommandType [1]
InformationElementRequest
-Message-Protocol : String
-ChannelID : OctetSeq [1]
-ChannelName : String [0..1]
1
1
1
1
IES-Element
-SemanticElementID : OctetSeq [1]
-SemanticElementName : String [0..1]
1
0..*
IES-ElementFilter
-FilterID : OctetSeq [1]
-FilterName : String [0..1]
-AttributeID : OctetSeq [1]
-AttributeName : String [0..1]
-Condition : String [1]
-Value : String

Figure 38 -PPS Request Message

The following table describes the message elements depicted in diagram “PPS Request Message”.

Table 25 - PPS Request Message Attributes	
Element Name	**Attributes**
IES-Element	Meta description of an IES informationElement. ***Owned Attributes:*** *SemanticElementID* (type: OctetSeq) [1..1]: The Unique Identifier for the base semantic Element. *SemanticElementName* (type: String) [0..1]: The Unique Identifier for the base semantic Element.
IES-ElementFilter	Characteristics for the filter assigned to the base SemanticElement to produce the required data. ***Owned Attributes:*** *AttributeID* (type: OctetSeq) [1..1]: Unique identifier for the attribute used in the filter. *AttributeName* (type: String) [0..1]: Name of the attribute used in the filter. *Condition* (type: String) [1..1]: Boolean condition that satisfies the filter. *FilterID* (type: OctetSeq) [1..1]: The unique identifier for the filter. *FilterName* (type: String) [0..1]: The name of the filter. *Value* (type: String) [1..1]: The value for the filter.
IES-Modification	Request for the modification to an existing Information Exchange Specification. ***Owned Attributes:*** *IES--Name* (type: String) [0..1]: The name of the contract (information exchange specification) that the user is seeking to participate in. *IES-ID* (type: OctetSeq) [1..1]: The unique identifier for the contract (information exchange specification) that the user is seeking to participate in. *RequestType* (type: IES-CommandType) [1..1]:

InformationElementRequest	A request for information being maintained by a PPS. The request identifies the semantic, filters, and protocols for the packaging and release of the information. ***Owned Attributes:*** *ChannelID* (type: OctetSeq) [1..1]: The unique identifier for the channel to be used to exchange the information elements. *ChannelName* (type: String) [0..1]: The name of the channel to be used to exchange the information elements. *Message-Protocol* (type: String) [1..1]:
PPS-Request	An ISMB-message from a PEP to the PPS to request the release of information based on a specified SemanticElement. The request contains the base (unique) identifier for the information elements being requested, and the identifier for semantic policies to be applied. ***Owned Attributes:*** *PPS-RequestType* (type: PPS-RequestType) [1..1]: **Inherited Attributes:** *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message*

16.3.9 ISSG Request Message

The following figure identifies the data and information elements issued by an IEF component to request information and data elements from the users' own security services and infrastructure, including services related to:

- Identity Management;
- Privilege (authorization / attribute) Management;
- Cryptographic Key Management; or
- TrustMark Registry.

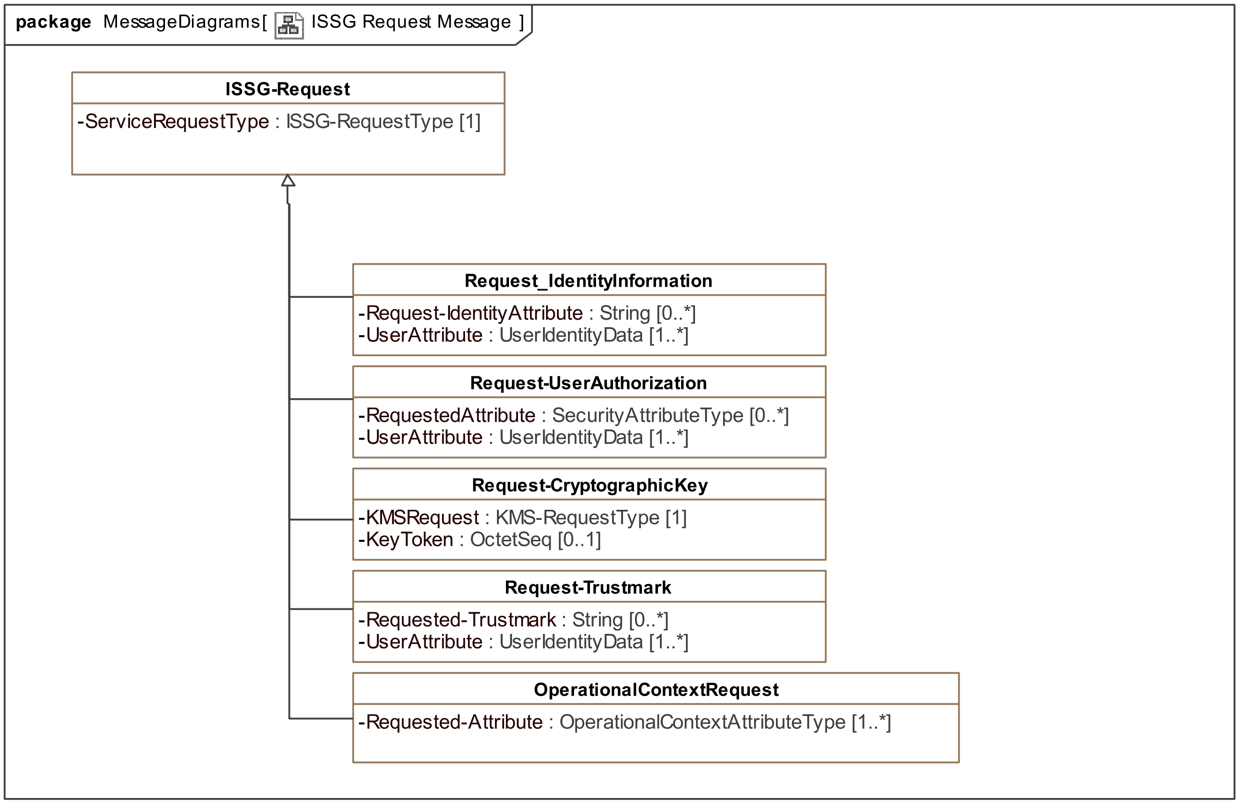

package MessageDiagrams[ISSG Request Message]
ISSG-Request
-ServiceRequestType : ISSG-RequestType [1]
Request_IdentityInformation
-Request-IdentityAttribute : String [0..*]
-UserAttribute : UserIdentityData [1..*]
Request-UserAuthorization
-RequestedAttribute : SecurityAttributeType [0..*]
-UserAttribute : UserIdentityData [1..*]
Request-CryptographicKey
-KMSRequest : KMS-RequestType [1]
-KeyToken : OctetSeq [0..1]
Request-Trustmark
-Requested-Trustmark : String [0..*]
-UserAttribute : UserIdentityData [1..*]
OperationalContextRequest
-Requested-Attribute : OperationalContextAttributeType [1..*]

Figure 39 -ISSG Request Message

The following table describes the message elements depicted in diagram “ISSG Request Message”.

Table 26 - ISSG Request Message Attributes

Element Name	Attributes
ISSG-Request	An *ISMB-message* issued by an IEF component to the **ISSG** requesting data / information from a user specified and provisioned security service, e.g.: • Identity Management; • Privilege (/attribute / authorization) Management; • Cryptographic Key Management; or • TrustMark Registry. ***Owned Attributes:*** *ServiceRequestType* (type: ISSG-RequestType) [1..1]:

	Identifies the type of information being requested from an external user specified security service. **Inherited Attributes:** *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message*
OperationalContextRequest	(Optional) Component request for information about the current operational context / situation. Used in environment where ISS policies can be tailored to address changes in operational context. ***Owned Attributes:*** *Requested-Attribute* (type: OperationalContextAttributeType) [1..*]: Identities the operational context attribute being requested. **Inherited Attributes:** *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message* *ServiceRequestType* inherited from *ISSG-Request*
Request-CryptographicKey	Component request for a cryptographic key from the user specified and provisioned key management services. This request can request the generation of a new key and token pair, or the retrieval of an existing key based on the token provided. ***Owned Attributes:*** *KeyToken* (type: OctetSeq) [0..1]: Unique identifier for a cryptographic key which is stored by the user specified and provisioned key escrow service. *KMSRequest* (type: KMS-RequestType) [1..1]: Identifies the type of request being made to the KMS. **Inherited Attributes:** *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message*

	MessageSourceID inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message* *ServiceRequestType* inherited from *ISSG-Request*
Request-Trustmark	(Optional) Component request to a user specified TrustMark Registry for a user's authorizations (/trustmarks/policies). Placeholder for the integration of a TrustMark framework. ***Owned Attributes:*** *Requested-Trustmark* (type: String) [0..*]: Identifies the type(s) of Trustmarks being requested. If the attribute is not sent - the **ISSG** requests all authorized user Trustmarks. *UserAttribute* (type: UserIdentityData) [1..*]: The user attributed used to obtain information. **Inherited Attributes:** *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message* *ServiceRequestType* inherited from *ISSG-Request*
Request-UserAuthorization	Component requests for users' authorizations (/privileges / attributes). ***Owned Attributes:*** *RequestedAttribute* (type: SecurityAttributeType) [0..*]: Identifies the type of attributes (authorizations, privileges) being requested. If the attribute is not sent - the **ISSG** requests all user attributes. *UserAttribute* (type: UserIdentityData) [1..*]: **Inherited Attributes:** *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message*

	ProcessStatus inherited from *ISMB-Message* *ServiceRequestType* inherited from *ISSG-Request*
Request_IdentityInformation	Component request for identity information for one or more users. ***Owned Attributes:*** *Request-IdentityAttribute* (type: String) [0..*]: Identifies the type of identity information being requested. If an attribute is not sent - the ISSG requests all user attributes. It is up to the user to identify the types of identity that is available to the PDP for adjudication. *UserAttribute* (type: UserIdentityData) [1..*]: **Inherited Attributes:** *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message* *ServiceRequestType* inherited from *ISSG-Request*

16.3.10 ISSG Response Message

The following figure identifies the data and information elements provided by the ISSG to an IEF component in response to a request to the user's security services and infrastructure.

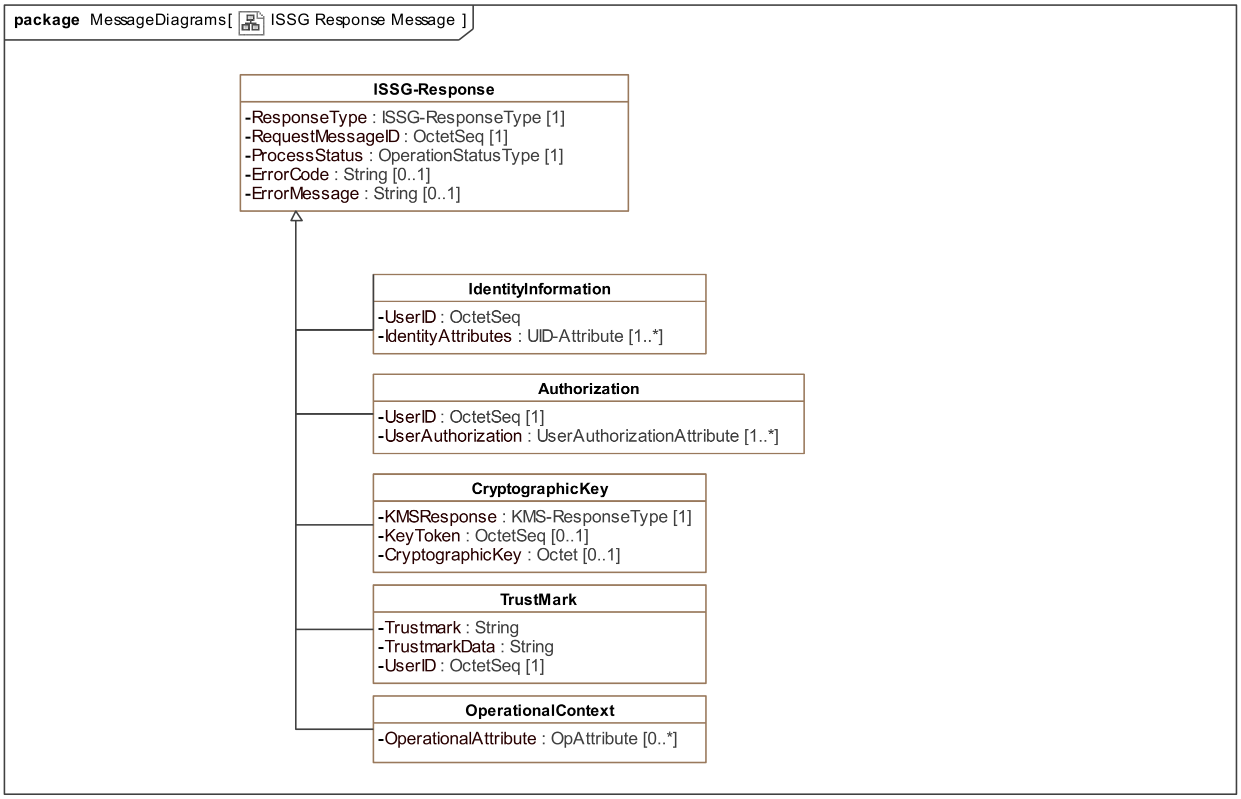

package MessageDiagrams[ISSG Response Message]
ISSG-Response
-ResponseType : ISSG-ResponseType [1]
-RequestMessageID : OctetSeq [1]
-ProcessStatus : OperationStatusType [1]
-ErrorCode : String [0..1]
-ErrorMessage : String [0..1]
IdentityInformation
-UserID : OctetSeq
-IdentityAttributes : UID-Attribute [1..*]
Authorization
-UserID : OctetSeq [1]
-UserAuthorization : UserAuthorizationAttribute [1..*]
CryptographicKey
-KMSResponse : KMS-ResponseType [1]
-KeyToken : OctetSeq [0..1]
-CryptographicKey : Octet [0..1]
TrustMark
-Trustmark : String
-TrustmarkData : String
-UserID : OctetSeq [1]
OperationalContext
-OperationalAttribute : OpAttribute [0..*]

Figure 40 -ISSG Response Message

The following table describes the message elements depicted in diagram "ISSG Response Message".

Table 27 - ISSG Response Message Attributes

Element Name	Attributes
Authorization	**ISSG** message that returns the user authorizations, rights or privileges to the requesting component. ***Owned Attributes:*** *UserAuthorization* (type: UserAuthorizationAttribute) [1..*]: Array of privileges/authorizations for a specified set of users. *UserID* (type: OctetSeq) [1..1]: **Inherited Attributes:** *ErrorCode* inherited from *ISSG-Response* *ErrorMessage* inherited from *ISSG-Response*

	Issue-DateTime inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISSG-Response* *ProcessStatus* inherited from *ISMB-Message* *RequestMessageID* inherited from *ISSG-Response* *ResponseType* inherited from *ISSG-Response*
CryptographicKey	**ISSG** message that returns a requested cryptographic key to the requesting component. ***Owned Attributes:*** *CryptographicKey* (type: Octet) [0..1]: Value of the Cryptographic Key. *KeyToken* (type: OctetSeq) [0..1]: Token for retrieving the cryptographic key from escrow. *KMSResponse* (type: KMS-ResponseType) [1..1]: Response type from the key management service. **Inherited Attributes:** *ErrorCode* inherited from *ISSG-Response* *ErrorMessage* inherited from *ISSG-Response* *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISSG-Response* *ProcessStatus* inherited from *ISMB-Message* *RequestMessageID* inherited from *ISSG-Response* *ResponseType* inherited from *ISSG-Response*

IdentityInformation	ISSG-message that returns user identity information, for one or more users, to the requesting component. ***Owned Attributes:*** *IdentityAttributes* (type: UID-Attribute) [1..*]: List of user identity attributes. *UserID* (type: OctetSeq) [1..1]: The user's unique identifier. **Inherited Attributes:** *ErrorCode* inherited from *ISSG-Response* *ErrorMessage* inherited from *ISSG-Response* *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISSG-Response* *ProcessStatus* inherited from *ISMB-Message* *RequestMessageID* inherited from *ISSG-Response* *ResponseType* inherited from *ISSG-Response*
ISSG-Response	An ISMB-message from the ISSG to the IEF Component providing the requested data or Information elements. ***Owned Attributes:*** *ErrorCode* (type: String) [0..1]: If unable to complete the request, provide the unique ErrorCode for the issue encountered. *ErrorMessage* (type: String) [0..1]: If unable to complete the request, provide the text describing the issue encountered. *ProcessStatus* (type: OperationStatusType) [1..1]: *RequestMessageID* (type: OctetSeq) [1..1]: Unique identifier for the message that requested the information. *ResponseType* (type: ISSG-ResponseType) [1..1]: Identifies the ISSG-message type which is used to stage the appropriate processing sequence for the message. **Inherited Attributes:**

	Issue-DateTime inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message*
OperationalContext	**ISSG-**message that returns situational awareness or incident data to the requesting component. ***Owned Attributes:*** *OperationalAttribute* (type: OpAttribute) [0..*]: Data describing operational context. **Inherited Attributes:** *ErrorCode* inherited from *ISSG-Response* *ErrorMessage* inherited from *ISSG-Response* *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISSG-Response* *ProcessStatus* inherited from *ISMB-Message* *RequestMessageID* inherited from *ISSG-Response* *ResponseType* inherited from *ISSG-Response*
TrustMark	**ISSG** message that returns TrustMark attributes to the requesting component. Placeholder for a specialized integration with a Trustmark Registry. ***Owned Attributes:*** *Trustmark* (type: String) [1..1]: *TrustmarkData* (type: String) [1..1]: *UserID* (type: OctetSeq) [1..1]: **Inherited Attributes:** *ErrorCode* inherited from *ISSG-Response* *ErrorMessage* inherited from *ISSG-Response* *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message*

	MessageSourceID inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISSG-Response* *ProcessStatus* inherited from *ISMB-Message* *RequestMessageID* inherited from *ISSG-Response* *ResponseType* inherited from *ISSG-Response*

16.3.11 CTS Request Message

The following figure identifies the data and information elements sent by an IEF component to the Cryptographic Transformation Service (CTS) to request the transformation (i.e., encryption or decryption) of an InformationElement.

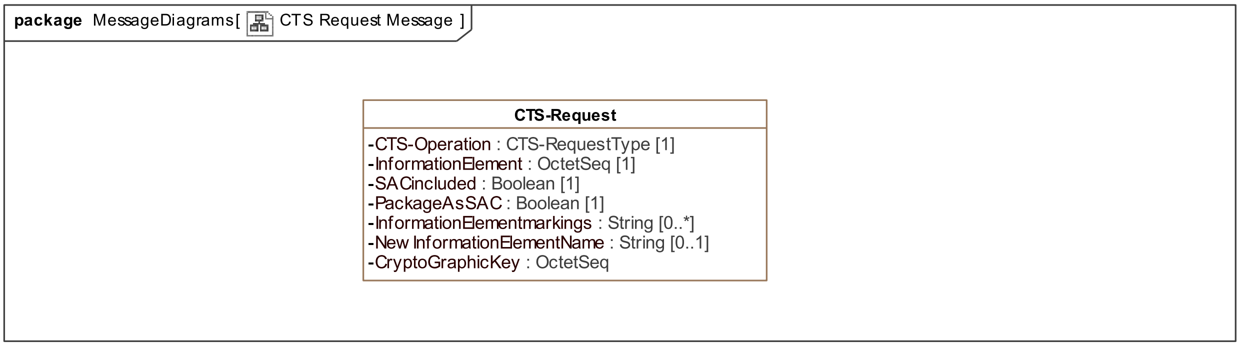

package MessageDiagrams[CTS Request Message]
CTS-Request
-CTS-Operation : CTS-RequestType [1]
-InformationElement : OctetSeq [1]
-SACincluded : Boolean [1]
-PackageAsSAC : Boolean [1]
-InformationElementmarkings : String [0..*]
-New InformationElementName : String [0..1]
-CryptoGraphicKey : OctetSeq

Figure 41 -CTS Request Message

The following table describes the message elements depicted in diagram "CTS Request Message".

Table 28 - CTS Request Message Attributes	
Element Name	**Attributes**
CTS-Request	An ISMB-message to the CTS that requests the transformation (encryption or decryption) of the specified InformationElement. ***Owned Attributes:*** *CryptoGraphicKey* (type: OctetSeq) [1..1]: The key to encrypt or decrypt the information element. *CTS-Operation* (type: CTS-RequestType) [1..1]: The transformation being requested (i.e., encryption or decryption). *InformationElement* (type: OctetSeq) [1..1]: The InformationElement being transformed by the CTS. *InformationElementmarkings* (type: String) [0..*]: Marking to be included within the SAC for the informationElement. (See Secure Access Container). Markings are only required when a SAC is being prepared and the SAC is not already provided. Each of the strings has the name value pair for the marking. *NewInformationElementName* (type: String) [0..1]: Provides a new File name for the InformationElement in the SAC. Requires the CTS to unpackage the original SAC, and create a new SAC with the new file name applied. *PackageAsSAC* (type: Boolean) [1..1]: Directs the CTS to return the encrypted informationElement within a Secure Access Container (SAC). *SACincluded* (type: Boolean) [1..1]: Identifies that the InformationElement contained in the message is a SAC. **Inherited Attributes:** *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message*

16.3.12 CTS Response Message

The following figure identifies the data and information elements issued by the Cryptographic Transformation Services (CTS) in response to a request to transform an InformationElement.

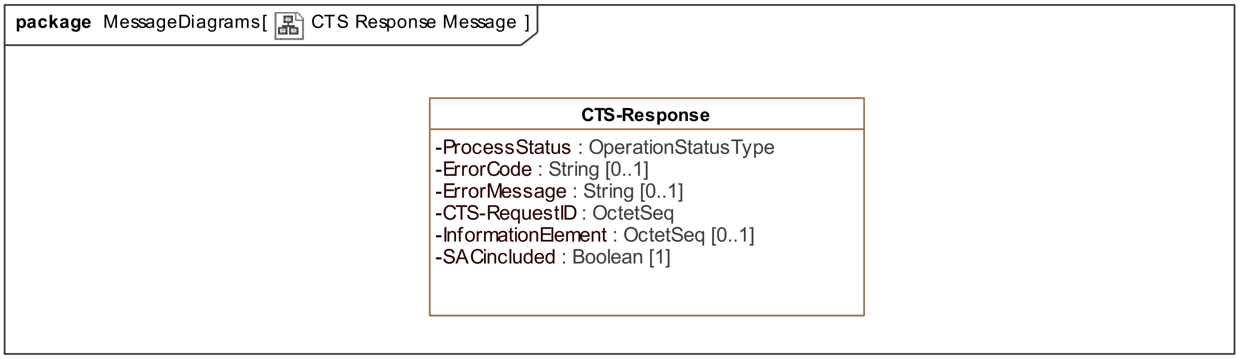

package MessageDiagrams[CTS Response Message]
CTS-Response
-ProcessStatus : OperationStatusType
-ErrorCode : String [0..1]
-ErrorMessage : String [0..1]
-CTS-RequestID : OctetSeq
-InformationElement : OctetSeq [0..1]
-SACincluded : Boolean [1]

Figure 42 -CTS Response Message

The following table describes the message elements depicted in diagram "CTS Response Message".

Table 29 - CTS Response Message Attributes

Element Name	Attributes
CTS-Response	An ISMB-message from a CTS to an IEF component responding to a cryptographic transformation request. ***Owned Attributes:*** *CTS-RequestID* (type: OctetSeq) [1..1]: The unique identifier for the corresponding CTS-Request Message. *ErrorCode* (type: String) [0..1]: If unable to transform the InformationElement as requested, provide the unique ErrorCode for the issue encountered. *ErrorMessage* (type: String) [0..1]: If unable to transform the InformationElement as requested, provide the text describing the issue encountered. *InformationElement* (type: OctetSeq) [0..1]: Transformed InformationElement. This element is not included if an error is encountered during the transformation. *ProcessStatus* (type: OperationStatusType) [1..1]: *SACincluded* (type: Boolean) [1..1]: Identifies that the informationElement included in the message is a SAC. **Inherited Attributes:**

	Issue-DateTime inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message*

16.3.13 TLS Log Report Message

The following figure identifies the data and information elements issued by an IEF component to log a transaction or event.

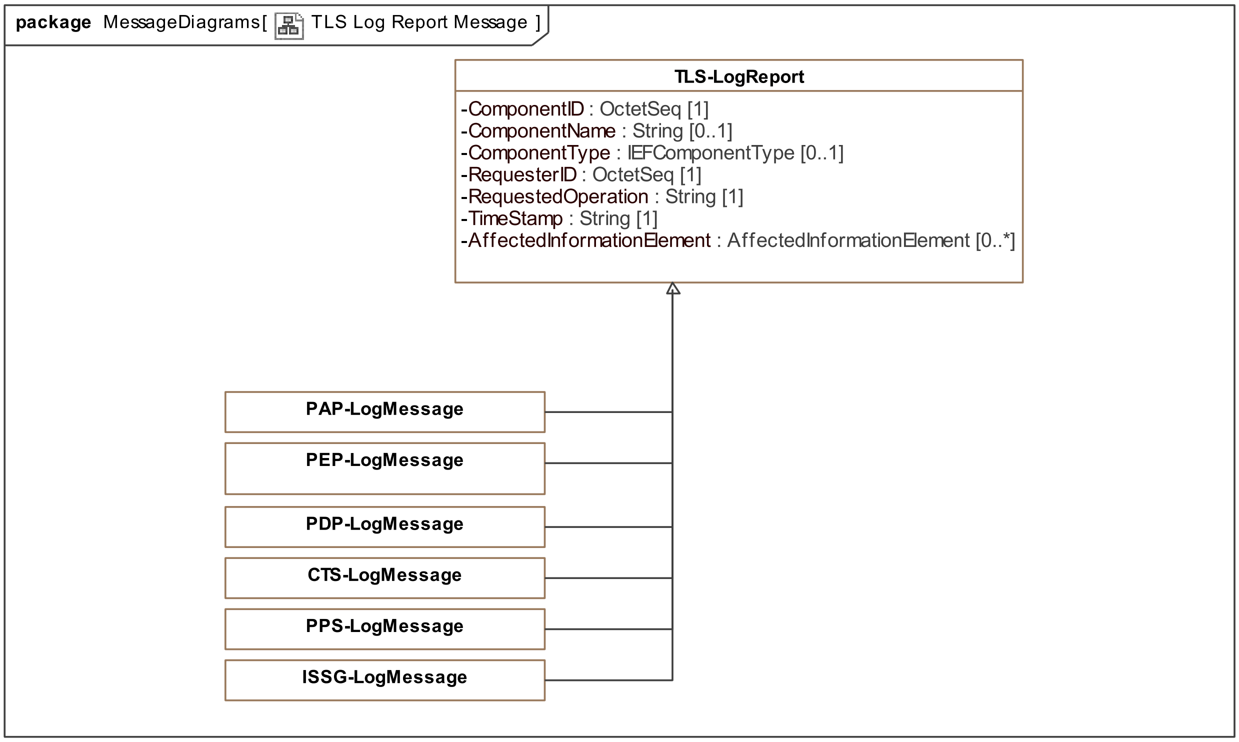

package MessageDiagrams[TLS Log Report Message]
TLS-LogReport
-ComponentID : OctetSeq [1]
-ComponentName : String [0..1]
-ComponentType : IEFComponentType [0..1]
-RequesterID : OctetSeq [1]
-RequestedOperation : String [1]
-TimeStamp : String [1]
-AffectedInformationElement : AffectedInformationElement [0..*]
PAP-LogMessage
PEP-LogMessage
PDP-LogMessage
CTS-LogMessage
PPS-LogMessage
ISSG-LogMessage

Figure 43 -TLS Log Report Message

The following table describes the message elements depicted in diagram “TLS Log Report Message”.

Table 30 - TLS Log Report Message Attributes

Element Name	**Attributes**
CTS-LogMessage	User specified elements of a CTS log message to the TLS. ***Owned Attributes:*** **Inherited Attributes:** *AffectedInformationElement* inherited from *TLS-LogReport* *ComponentID* inherited from *TLS-LogReport* *ComponentName* inherited from *TLS-LogReport* *ComponentType* inherited from *TLS-LogReport* *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message* *RequestedOperation* inherited from *TLS-LogReport* *RequesterID* inherited from *TLS-LogReport* *TimeStamp* inherited from *TLS-LogReport*
ISSG-LogMessage	User specified elements of a ISSG log message to the TLS. ***Owned Attributes:*** **Inherited Attributes:** *AffectedInformationElement* inherited from *TLS-LogReport* *ComponentID* inherited from *TLS-LogReport* *ComponentName* inherited from *TLS-LogReport* *ComponentType* inherited from *TLS-LogReport* *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message* *RequestedOperation* inherited from *TLS-LogReport* *RequesterID* inherited from *TLS-LogReport*

	TimeStamp inherited from *TLS-LogReport*
PAP-LogMessage	User specified elements of a PAP log message to the TLS. ***Owned Attributes:*** **Inherited Attributes:** *AffectedInformationElement* inherited from *TLS-LogReport* *ComponentID* inherited from *TLS-LogReport* *ComponentName* inherited from *TLS-LogReport* *ComponentType* inherited from *TLS-LogReport* *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message* *RequestedOperation* inherited from *TLS-LogReport* *RequesterID* inherited from *TLS-LogReport* *TimeStamp* inherited from *TLS-LogReport*
PDP-LogMessage	User specified elements of a PDP log message to the TLS. ***Owned Attributes:*** **Inherited Attributes:** *AffectedInformationElement* inherited from *TLS-LogReport* *ComponentID* inherited from *TLS-LogReport* *ComponentName* inherited from *TLS-LogReport* *ComponentType* inherited from *TLS-LogReport* *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message*

	ProcessStatus inherited from *ISMB-Message* *RequestedOperation* inherited from *TLS-LogReport* *RequesterID* inherited from *TLS-LogReport* *TimeStamp* inherited from *TLS-LogReport*
PEP-LogMessage	User specified elements of a PEP log message to the TLS. ***Owned Attributes:*** **Inherited Attributes:** *AffectedInformationElement* inherited from *TLS-LogReport* *ComponentID* inherited from *TLS-LogReport* *ComponentName* inherited from *TLS-LogReport* *ComponentType* inherited from *TLS-LogReport* *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message* *RequestedOperation* inherited from *TLS-LogReport* *RequesterID* inherited from *TLS-LogReport* *TimeStamp* inherited from *TLS-LogReport*
PPS-LogMessage	User specified elements of a PPS log message to the TLS. ***Owned Attributes:*** *InformationExchangeSpecification* (type: String) [1..1]: Identifies the InformationExchangeSpecification exercised during the transaction. **Inherited Attributes:** *AffectedInformationElement* inherited from *TLS-LogReport* *ComponentID* inherited from *TLS-LogReport* *ComponentName* inherited from *TLS-LogReport* *ComponentType* inherited from *TLS-LogReport* *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message*

	MessageSourceID inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message* *RequestedOperation* inherited from *TLS-LogReport* *RequesterID* inherited from *TLS-LogReport* *TimeStamp* inherited from *TLS-LogReport*
TLS-LogReport	An ISMB-Message issued by an IEF component to the Trusted Logging Service(s) describing: • Operations on InformationElements protected by the IEF; • Changes to the operating characteristics of an IEF Component; and • Changes to the Data Policies or Access & Release Control policies. These messages enable the tamper-resistant recording of IEF operations. The log(s) supports both Security Incident and Event Monitoring (SIEM) and forensic auditing of the environment. Each IEF transaction must be recorded in a manner that resists tampering and alteration of the log records. The log is intended to maintain a chain-of-custody record for all information elements protected by an IEF implementation. Each log record is encrypted in motion and storage and assigned a chained digital signature. ***Owned Attributes:*** *AffectedInformationElement* (type: AffectedInformationElement) [0..*]: Identifies the information elements affected by the transaction being logged. *ComponentID* (type: OctetSeq) [1..1]: The unique identifier of the IEF Component that handled the transaction. *ComponentName* (type: String) [0..1]: The name/type of PEP that handled the transaction. *ComponentType* (type: IEFComponentType) [0..1]: IEF Component type logging an operation or transaction. *RequestedOperation* (type: String) [1..1]: The operation requested on the data asset. *RequesterID* (type: OctetSeq) [1..1]: The identity of the user that requested the data transaction. *TimeStamp* (type: String) [1..1]: A time stamp (date and time) generated by the IEF components (typically system time), indicating to when each transaction was handled.

	Inherited Attributes: *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message*

16.3.14 TLS Data Request Message

The following figure identifies the data and information elements sent to the TLS to request the release (dissemination) of information log records to a monitoring or auditing application.

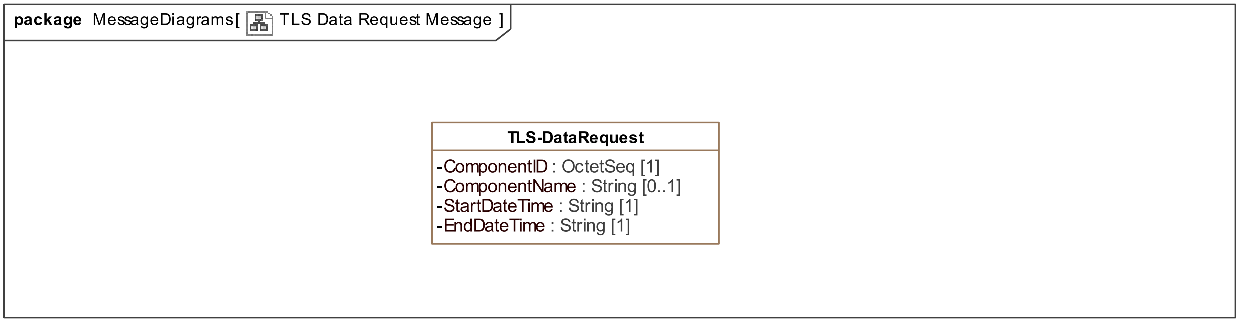

package MessageDiagrams[TLS Data Request Message]
TLS-DataRequest
-ComponentID : OctetSeq [1]
-ComponentName : String [0..1]
-StartDateTime : String [1]
-EndDateTime : String [1]

Figure 44 -TLS Data Request Message

The following table describes the message elements depicted in diagram “TLS Data Request Message”.

Table 31 - TLS Data Request Message Attributes

Element Name	Attributes
TLS-DataRequest	An ISMB-Message issued by a User Monitoring Application or Auditing Application to the TLS requesting a log entry or set of log entries for an IEF component. ***Owned Attributes:*** *ComponentID* (type: OctetSeq) [1..1]: The unique identifier of the IEF Component for which the log data is being requested. *ComponentName* (type: String) [0..1]: The name of the IEF Component for which the log data is being requested. *EndDateTime* (type: String) [1..1]: The ending date and time the log reports being requested. *StartDateTime* (type: String) [1..1]: The starting date and time the log reports being requested. **Inherited Attributes:** *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message*

16.3.15 TLS Data Response Message

The following figure identifies the data and information elements sent by the TLS to a monitoring or auditing application in response to a TLS-DataRequest.

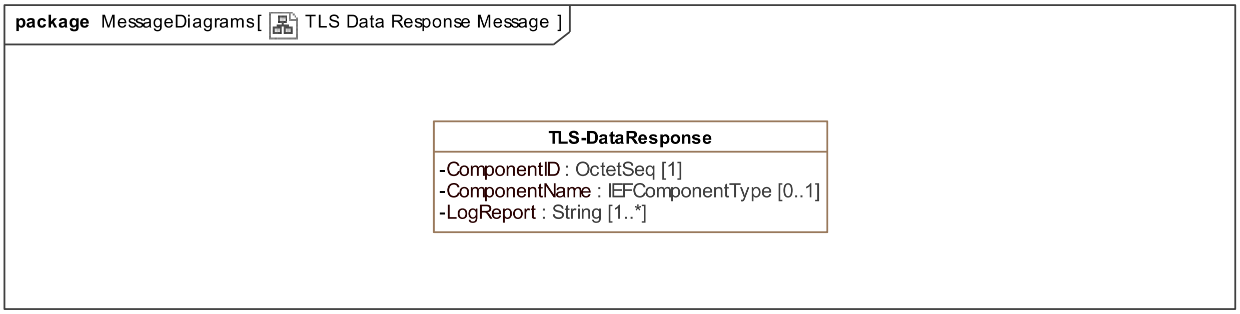

package MessageDiagrams[TLS Data Response Message]
TLS-DataResponse
-ComponentID : OctetSeq [1]
-ComponentName : IEFComponentType [0..1]
-LogReport : String [1..*]

Figure 45 -TLS Data Response Message

The following table describes the message elements depicted in diagram "TLS Data Response Message".

Table 32 - TLS Data Response Message Attributes

Element Name	**Attributes**
TLS-DataResponse	An ISMB-Message issued from the TLS to the User Auditing or Monitoring Application. ***Owned Attributes:*** *ComponentID* (type: OctetSeq) [1..1]: Unique identifier for the component that issues the LogReport. *ComponentName* (type: IEFComponentType) [0..1]: Name of the component that issues the LogReport. *LogReport* (type: String) [1..*]: Requested Log Report. **Inherited Attributes:** *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message* *ProcessStatus* inherited from *ISMB-Message*

16.3.16 Ack Message

The following figure identifies the data and information elements sent to the originating component in order to acknowledge the receipt of message.

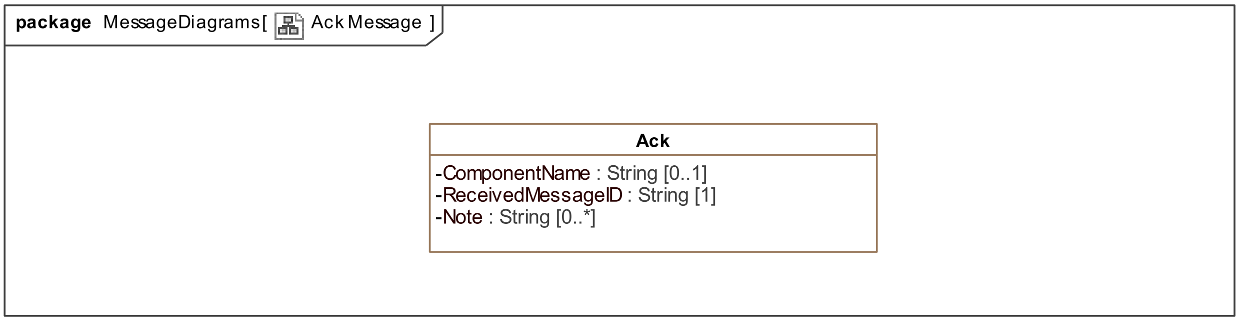

package MessageDiagrams[Ack Message]
Ack
-ComponentName : String [0..1]
-ReceivedMessageID : String [1]
-Note : String [0..*]

Figure 46 -Ack Message

The following table describes the message elements depicted in diagram "Ack Message".

Table 33 - Ack Message Attributes	
Element Name	**Attributes**
Ack	An ISMB-Message from a message recipient to the message sender reporting that a specific message was received. ***Owned Attributes:*** *ComponentName* (type: String) [0..1]: Name of the acknowledging component. *Note* (type: String) [0..*]: Other supporting information included in the acknowledgment message. *ReceivedMessageID* (type: String) [1..1]: Unique identifier for the message being acknowledged. **Inherited Attributes:** *Issue-DateTime* inherited from *ISMB-Message* *MessageID* inherited from *ISMB-Message* *MessageSourceID* inherited from *ISMB-Message* *MessageTargetID* inherited from *ISMB-Message* *MessageType* inherited from *ISMB-Message*

	ProcessStatus inherited from *ISMB-Message*

16.4 Additional Data Patterns

16.4.1 Metadata Patterns

The following clauses are provided as guidance to users, developers, and integrators during their development of metadata patterns for message elements (e.g., messages, information packages, and information payload) and information elements (e.g., digests, information payloads, and files). The IEF is specified in a manner that enables the user to adopt the metadata standards that best suit their organizations and/or community.

16.4.1.1 Message Metadata

The following figure identifies the Metadata elements defined for a Message structure. These data elements are used by the Messaging-PEP to authorize its release and route the message to the communication channel.

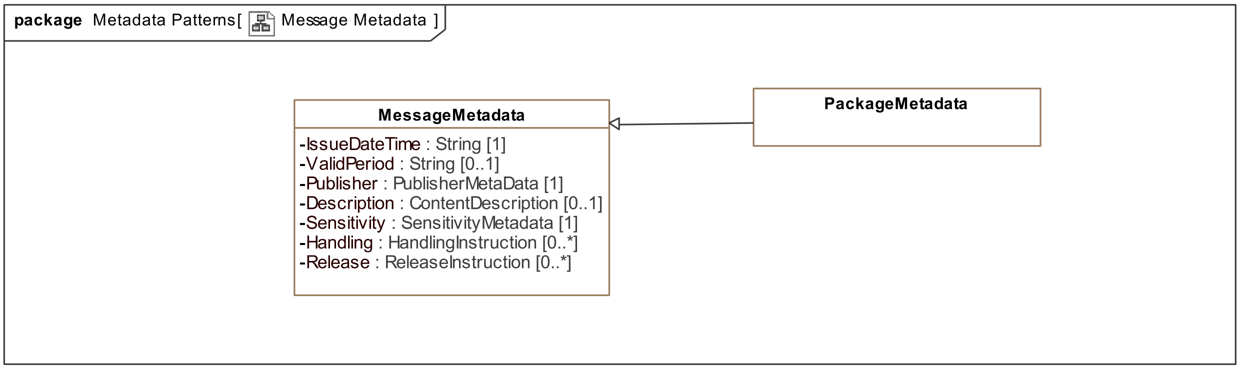

package Metadata Patterns[Message Metadata]
MessageMetadata
-IssueDateTime : String [1]
-ValidPeriod : String [0..1]
-Publisher : PublisherMetaData [1]
-Description : ContentDescription [0..1]
-Sensitivity : SensitivityMetadata [1]
-Handling : HandlingInstruction [0..*]
-Release : ReleaseInstruction [0..*]
PackageMetadata

Figure 47 -Message Metadata

The following table describes the message elements depicted in diagram “Message Metadata”.

Table 34 - Message Metadata Attributes

Element Name	Attributes
MessageMetadata	*Metadata* elements that may be included in the message envelope or header. These *DataElements* are typically sent in the clear (no encryption) in order for the sender's and recipients` infrastructure (e.g., **PEP**) to ascertain authorizations to access, process, store or share the message content. ***Owned Attributes:*** *Description* (type: ContentDescription) [0..1]: Brief description of the information element or message. *Handling* (type: HandlingInstruction) [0..*]: Handling instruction to the recipient for the information element or message. *IssueDateTime* (type: String) [1..1]: The TimeStamp identifying when the InformationElement was issued. *Publisher* (type: PublisherMetaData) [1..1]: The user releasing or sending the information or message. *Release* (type: ReleaseInstruction) [0..*]: *Sensitivity* (type: SensitivityMetadata) [1..1]: Sensitivity of the information element or message. *ValidPeriod* (type: String) [0..1]: The period or duration for which the data in the InformationElement is valid for use. The string is a composite of two DateTime strings separated by " / ".
PackageMetadata	Data (tags and markings) that describes the information in the InformationPackage. ***Owned Attributes:***

16.4.1.2 InformationElement Metadata

The following figure identifies the **Metadata** elements for an *InformationPayload* within a *Message* structure.

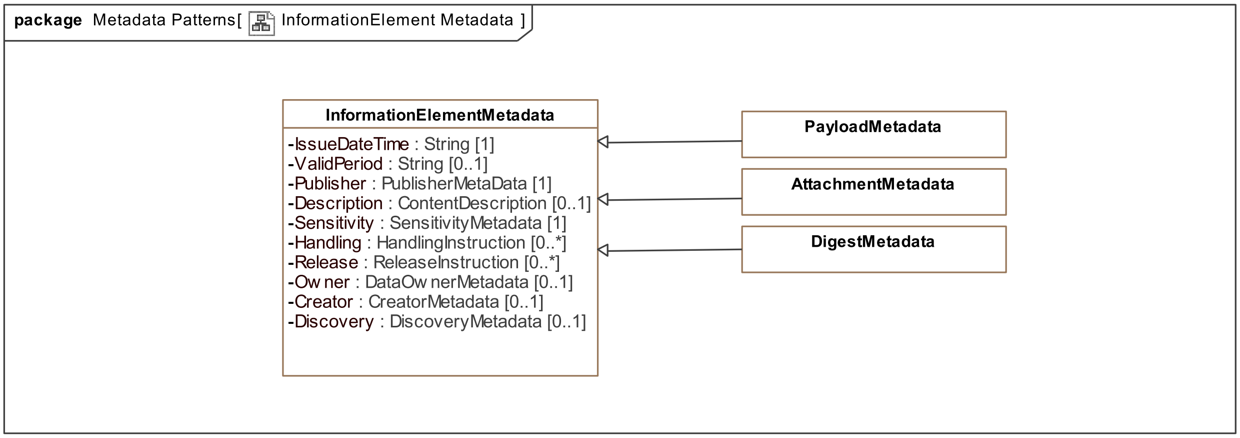

package Metadata Patterns[InformationElement Metadata]
InformationElementMetadata
-IssueDateTime : String [1]
-ValidPeriod : String [0..1]
-Publisher : PublisherMetaData [1]
-Description : ContentDescription [0..1]
-Sensitivity : SensitivityMetadata [1]
-Handling : HandlingInstruction [0..*]
-Release : ReleaseInstruction [0..*]
-Ow ner : DataOw nerMetadata [0..1]
-Creator : CreatorMetadata [0..1]
-Discovery : DiscoveryMetadata [0..1]
PayloadMetadata
AttachmentMetadata
DigestMetadata

Figure 48 -InformationElement Metadata

The following table describes the message elements depicted in diagram "InformationElement Metadata".

Table 35 - InformationElement Metadata Attributes

Element Name	Attributes
AttachmentMetadata	Set of metadata elements that may be attached to a message attachment. Specialization of InformationElement Metadata. ***Owned Attributes:***
DigestMetadata	Set of metadata elements that may be attached to a message digest. Specialization of InformationElement Metadata. ***Owned Attributes:***
InformationElementMetadata	Metadata describing an InformationElement. ***Owned Attributes:*** *Creator* (type: CreatorMetadata) [0..1]: Information regarding the creator of the information element or message.

	Description (type: ContentDescription) [0..1]: Brief description of the information element or message. *Discovery* (type: DiscoveryMetadata) [0..1]: Data elements provided to enable discovery of the infromation element or message. *Handling* (type: HandlingInstruction) [0..*]: Handling instruction to the recipient for the information element or message. *IssueDateTime* (type: String) [1..1]: The TimeStamp identifying when the InformationElement was issued. *Owner* (type: DataOwnerMetadata) [0..1]: Information regarding the owner or steward for the information element or message. *Publisher* (type: PublisherMetaData) [1..1]: The user releasing or sending the information or message. *Release* (type: ReleaseInstruction) [0..*]: Handling instruction to the PEP for the information element or message. *Sensitivity* (type: SensitivityMetadata) [1..1]: Sensitivity of the information element or message. *ValidPeriod* (type: String) [0..1]: The period or duration for which the data in the InformationElement is valid for use. The string is a composite of two DateTime strings separated by " / ".
PayloadMetadata	Set of metadata elements that may be attached to a message payload. Specialization of InformationElement Metadata. ***Owned Attributes:***

16.4.2 SecureAssetContainer

The following figure describes data and information structures used by the IEF to secure user information.

Table 36 - SecureAssetContainer Attributes	
Element Name	**Attributes**
SecureAssetContainer	An envelope that allows some unprotected information to exist outside of the protected (encrypted) payload. In addition to the encrypted payload, the envelope includes an envelope header containing metadata that enables the identification, and discovery of the information in the container, the securing of the InformationElement and the container itself. ***Owned Attributes:*** *EncrypedPayload* (type: OctetSeq) [1..1]: Original InformationElement (File). *Header* (type: EnvelopHeader) [1..1]:

16.5 Message Data Types

The following table described the data types used in the ISMB-Messages in the previous clauses.

Table 37 - Message Data Types Attributes	
Element Name	**Attributes**
AffectedInformationElement	Identified information element affected by a reported transaction. ***Owned Attributes:*** *InformationElementID* (type: OctetSeq) [1..1]: Unique identifier for the information element (Asset) being operated on. *InformationElementName* (type: String) [1..1]: The name of the target asset (e.g. component name, file name) affected by the request.
Attachment	A binary file (e.g., PDF file, image or video), and its metadata. ***Owned Attributes:*** *BinaryObject* (type: OctetSeq) [1..1]: A binary object containing an information element.

ChannelAttributes	Network attributes retrieved by the PEP from the users` security services. The attributes identify the classes (security level, and caveats) of information that the exchange services are authorized to carry. ***Owned Attributes:*** *ChannelAuthorizations* (type: SensitivityMetadata) [1..1]: The security level authorization(s) for the specified network or network services. *ChannelDescription* (type: String) [0..1]: *ChannelID* (type: OctetSeq) [1..1]: Unique identifier for the information exchange connection. *ChannelName* (type: String) [0..1]: Name of the information exchange connection. *MessagingProtocol.* (type: string) [1..1]: Messaging Protocol (/ Semantics / Canonical Model) used to structure and format data elements (e.g., NIEM). *MessagingServiceID* (type: OctetSeq) [1..1]: Unique identifier for the messaging service. *MessagingServiceName* (type: String) [1..1]: Identifies the messaging infrastructure (e.g., DDS, or AMQP) for the connection. *NetworkProtocol* (type: String) [0..1]:
Component-ConfigurationReport	ISMB-Message that contains the IEF Component configuration for the **PAP**. ***Owned Attributes:*** *ParameterSettings* (type: ParameterSettingReport) [1..*]: Array of Parameter settings.
ComponentParameter	The configuration parameter (feature) and the value (setting) to be applied). ***Owned Attributes:*** *FeatureName* (type: String) [1..1]: Name of the component feature being modified. *ParameterName* (type: String) [1..1]: Name of the parameter being modified. *ParameterSetting* (type: String) [1..1]: New value for the feature being modified.

ContentDescription	Metadata elements describing the content of the message. ***Owned Attributes:*** *Description* (type: String) [1..*]: Description of the message content.
CreatorMetadata	Metadata tags and markings that identify the creator of data or information elements. ***Owned Attributes:*** *CreationDate* (type: date) [1..1]: The date the information was created/generated. *CreationTime* (type:) [0..1]: The time the information was created/generated. *CreatorID* (type: String) [1..1]: Unique Identifier for the Data Creator. *CreatorLocation* (type: String) [1..*]: Physical location of the Data Creator. The location can be a physical location, network location, or other location agreed by the data sharing community. *CreatorName* (type: String) [0..1]: Unique Identifier for the Data Creator. *CreatorOrganization* (type: String) [1..1]: Name of the Organization the data creator represents.
DataOwnerMetadata	Tags and markings that identify the owner or steward of the data or information element. ***Owned Attributes:*** *OwnerID* (type: String) [1..1]: Unique Identifier for the Data. *OwnerLocation* (type: String) [0..*]: Physical location of the Data Owner. The location can be a physical location, network location, or other location agreed by the data sharing community. *OwnerName* (type: String) [0..1]: Name of the data creator. *OwnerOrganization* (type: String) [1..1]: Name of the Organization the data owner represents.

DiscoveryMetadata	Metadata elements that when combined with the Envelope Header metadata can be supplied to a data registry to enable the discovery of the InformationElement by an authorized user. Additional information element metadata may be added. ***Owned Attributes:*** *Author* (type: String) [1..*]: Author of the Information Elements (may include the Creator, Owner, or Publisher based on the user policy). *Description* (type: String) [1..1]: Brief Description of the InformationElement. *IssueDate* (type: date) [1..1]: The date the information element was issued. *KeyWords* (type: String) [0..*]: A word used to classify or organize digital content, or to facilitate an on-line discovery, search for the InformationElement(s) in the payload. *Location* (type: String) [1..1]: Authoritative Location (e.g., URL) for the InformationElement. *Version* (type: String) [1..1]: The version of the information element.
EncryptedPayload	Container element that holds the encrypted InformationElement. ***Owned Attributes:***
EnvelopHeader	Container element that holds the unencrypted data elements. ***Owned Attributes:*** *Caveat* (type: CaveatType) [0..*]: List field populated with a comma separated list of Caveat markings for the informationElement. *DiscoveryData* (type: DiscoveryMetadata) [0..1]: *HandlingInstruction* (type: String) [0..*]: String describing an instruction for the use or storage of the informationElement. *HashDigest* (type: OctetSeq) [1..1]:

	All Secure Asset Containers (SAC) include a digest that is calculated at creation time as a mechanism to detect tampering of the container. The digest is calculated using the SHA-xxx** hash algorithm. Digest calculation is dependent on the order in which the source material is submitted for the calculation. For SAC's, the digest is calculated as follows: • The contents of the encrypted InformationElement; • The Security Level for the InformationElement; • The caveats for the informationelement; • Privacy Indicators for the informationElement; • Special Handling instructions for the InformationElement; • The token for the encryption key; • The filename of the original file; and • The key used to encrypt the original file. **The selection of the hash algorithm is lefts to the user. *InformationElementName* (type: String) [1..1]: The filename of the original file. *keyToken* (type: String) [1..1]: When a request to decrypt the file is received, the container is opened and the keyToken is extracted. This is then used to retrieve the key from the KMS. *PrivacyIndicator* (type: String) [0..*]: List field populated with a comma separated list of privacy markings (e.g., PII) for the informationElement. *ReleaseInstruction* (type: String) [0..*]: String describing a restriction on the release of the informationElement. *SecurityLevel* (type: SecurityClassificationType) [1..1]: List field populated with a comma separated list of security marking for the informationElement. *** Selection of the hash algorithm is left to the user and their security needs.*
ExchangeServiceAttributes	Information exchange service attributes retrieved by the PEP from the users` security services. The attributes identify the classes (security level, and caveats) of information that the exchange services are authorized to carry. ***Owned Attributes:*** *CaveatAuthorizations* (type: CaveatType) [0..*]:

	The caveat authorization(s) for the specified information exchange services. *IES-ID* (type: OctetSeq) [1..1]: Unique ID for the Information Exchange Services (IES). *IES-Name* (type: String) [1..1]: Name of the Information Exchange Services. *SecurityAuthorization* (type: SecurityClassificationType) [0..*]: The security level authorization(s) for the specified Information Exchange Services.
FeatureStatusReport	Status of an individual feature. ***Owned Attributes:*** *FeatureName* (type: String) [1..1]: Name of the reporting feature. *FeatureStatus* (type: OperationStatusType) [1..1]: Current status of the reporting component feature.
HandlingInstruction	An instruction to the recipient of an information exchange specifying how this information must be handled. ***Owned Attributes:*** *Instructiontype* (type: HandlingInstructionType) [1..1]: Type of handling instruction. *InstructionValue* (type: String) [1..1]: Instruction to the recipient regarding the handing of the information in the element or message.
IES-Authorization	Authorization data for a specified information exchange Specification. ***Owned Attributes:*** *AuthorizationInstruction* (type: String) [0..*]: Handling or release instructions allowing the exchange services to disseminate the InformationElement. *AuthorizationResponse* (type: AuthorizationResponseType) [1..1]: The result of the PDP's adjudication on the exchange service's authorization to disseminate the InformationElement. *IES-ID* (type: OctetSeq) [1..1]: Unique ID for the Information Exchange Services (IES). *IES-Name* (type: String) [0..1]: Name of the Information Exchange Services.

	InformationElementID (type: OctetSeq) [1..1]: Unique identifier for the *InformationElement*. *InformationElementName* (type: String) [0..1]: The human readable name for the *InformationElement*.
ImpactedComponent	Components impacted by the execution of the operation. ***Owned Attributes:*** *ComponentID* (type:) [1..1]: Unique Identifier for the IEF component affected by the requested operation. *ComponentName* (type: String) [1..1]: Human readable name for the IEF component affected by the requested operation.
InformationElementAttributes	InformationElement attributes defining the sensitivity of the information, gathered from the message metadata issued by the PPS. ***Owned Attributes:*** *CaveatAuthorization* (type: CaveatType) [0..*]: The caveats assigned to the InformationElement. *HandlingInstruction* (type: HandlingInstructionType) [0..*]: Special user instruction governing the storage, access, release or movement of the information element. *InformationElementID* (type:) [1..1]: Unique identifier for the *InformationElement*. *InformationElementName* (type: String) [1..1]: The human readable name of the *InformationElement*. *SecurityAuthorization* (type: SecurityClassificationType) [0..*]: The security level assigned to the InformationElement.
InformationElementMetadata	Metadata describing an InformationElement. ***Owned Attributes:*** *Creator* (type: CreatorMetadata) [0..1]: Information regarding the creator of the information element or message. *Description* (type: ContentDescription) [0..1]: Brief description of the information element or message. *Discovery* (type: DiscoveryMetadata) [0..1]:

	Data elements provided to enable discovery of the information element or message. *Handling* (type: HandlingInstruction) [0..*]: Handling instruction to the recipient for the information element or message. *IssueDateTime* (type: String) [1..1]: The Time Stamp identifying when the InformationElement was issued. *Owner* (type: DataOwnerMetadata) [0..1]: Information regarding the owner or steward for the information element or message. *Publisher* (type: PublisherMetaData) [1..1]: The user releasing or sending the information or message. *Release* (type: ReleaseInstruction) [0..*]: Handling instruction to the PEP for the information element or message. *Sensitivity* (type: SensitivityMetadata) [1..1]: Sensitivity of the information element or message. *ValidPeriod* (type: String) [0..1]: The period or duration for which the data in the InformationElement is valid for use. The string is a composite of two DateTime strings separated by " / ".
MessageMetadata	*Metadata* elements that may be included in the message envelope or header. These *DataElements* are typically sent in the clear (no encryption) in order for the sender's and recipients` infrastructure (e.g., **PEP**) to ascertain authorizations to access, process, store or share the message content. ***Owned Attributes:*** *Description* (type: ContentDescription) [0..1]: Brief description of the information element or message. *Handling* (type: HandlingInstruction) [0..*]: Handling instruction to the recipient for the information element or message. *IssueDateTime* (type: String) [1..1]: The TimeStamp identifying when the InformationElement was issued. *Publisher* (type: PublisherMetaData) [1..1]: The user releasing or sending the information or message. *Release* (type: ReleaseInstruction) [0..*]: *Sensitivity* (type: SensitivityMetadata) [1..1]: Sensitivity of the information element or message. *ValidPeriod* (type: String) [0..1]:

	The period or duration for which the data in the InformationElement is valid for use. The string is a composite of two DateTime strings separated by " / ".
NetworkAuthorization	Authorization granted to the Network to carry the information Elements. ***Owned Attributes:*** *AuthorizationInstruction* (type: String) [0..*]: Handling or release instructions allowing the network to carry the InformationElement. *AuthorizationResponse* (type: AuthorizationResponseType) [1..1]: The result of the PDP's adjudication on the network's authorization to carry (/publish) the InformationElement. *InformationElementID* (type: OctetSeq) [1..1]: Unique identifier for the *InformationElement*. *InformationElementName* (type: String) [0..1]: The human readable name for the *InformationElement*. *NetworkID* (type:) [1..1]: Unique ID for the network and network services. *NetworkName* (type: String) [0..1]: Name of the network and network services.
Octet	A unit of digital information in computing and telecommunications that consists of eight bits. ***Owned Attributes:*** *Byte* (type: char) [1..1]: Eight bit data element.
OctetSeq	A Sequence of Octets. Represented as a string in XML. ***Owned Attributes:*** *String* (type: String) [1..1]: Finite sequence of characters (i.e., letters, numerals, symbols and punctuation marks).

OpAttribute	Attribute describing the operational context. ***Owned Attributes:*** *AttributeName* (type: OperationalContextAttributeType) [1..1]: The name of the Operational Attribute. *AttributeValue* (type: OctetSeq) [1..1]: Value of the operational attribute.
OperationalParameter	Parameter for a specified operation. ***Owned Attributes:*** *ParameterID* (type: OctetSeq) [0..1]: (Optional) Unique Identifier for the operation parameter. *ParameterName* (type: String) [1..1]: Human readable name for the parameter. *ParameterUnits* (type: String) [1..1]: Unit of measurement for the parameter. *ParameterValue* (type: OctetSeq) [1..1]: The assigned value for the parameter.
ParameterSettingReport	Report on a component parameter setting. ***Owned Attributes:*** *ParameterName* (type:) [1..1]: Name of the parameter. *ParameterValue* (type:) [1..1]: Value held by the parameter. *ParameterValueUnits* (type:) [0..1]: Standard quantity used in measurement.
Payload	InformationElement formatted in accordance with the specified exchange protocol (e.g., XSD). ***Owned Attributes:*** *TransmittedInformation* (type: OctetSeq) [1..1]: the actual information or message in transmitted data, as opposed to automatically generated metadata.

PPS-PublishMetadata	Metadata about the PPS publication and how the PEP must handle the publication. ***Owned Attributes:*** *HandlingInstructions* (type: String) [0..*]: Instructions to the recipient(s) of the InformationElements on how the information is to be handled (processed, stored, and shared). *IES-ID* (type: OctetSeq) [1..1]: Unique identifier of the information sharing agreement (or contract) specifying the release of the InformationElement(s). *IES-Name* (type: String) [0..1]: Name of the information sharing agreement (or contract) specifying the release of the InformationElement(s). *MessageID* (type: OctetSeq) [1..1]: Unique identifier for the message being published. *MessageName* (type: String) [0..1]: Name of the message being published. *MessageSensitivity* (type: SensitivityMetadata) [1..1]: Identifies the releaseability of the message content to recipients (sessions, domains, channels, topics, queues or individuals) authorized to access or transport information with the specified security level(s). *PayloadProtocol* (type: String) [1..1]: Identified the message protocol used to encode the InformationElement(s) in the message. *Recipients* (type: RecipientData) [0..1]: *ReleaseChannel* (type: ChannelAttributes) [1..1]: Identified the communication channel to be used to publish the information. *ReleaseInstruction* (type: String) [0..*]: Instructions to the PEP on how the information is to be handled (processed, and shared).
PPS-ReceiveMetadata	Metadata about the information element(s) being received. ***Owned Attributes:*** *HandlingInstructions* (type: HandlingInstruction) [0..*]: Instructions to the recipient(s) of the InformationElements on how the information is to be handled (processed, stored, and shared). *IES-ID* (type: OctetSeq) [1..1]: Unique identifier of the information sharing agreement (or contract) specifying the receipt of the InformationElement(s).

	IES-Name (type: String) [0..1]: Name of the information sharing agreement (or contract) specifying the receipt of the InformationElement(s). *MessageID* (type: OctetSeq) [1..1]: Unique identifier for the message being received. *MessageName* (type: String) [0..1]: Name of the message being received. *MessageSensitivity* (type: SensitivityMetadata) [1..1]: *PayloadProtocol* (type: String) [1..1]: Identifies the message protocol used to encode the InformationElement(s) in the message.
PublisherMetaData	Metadata Elements identifying the publisher (/sender) of the message. ***Owned Attributes:*** *PublisherID* (type: String) [1..1]: Unique Identifier for the Publisher. *PublisherLocation* (type: String) [1..*]: Location of the Data Publisher. The location can be a physical location, network location, or other location meaningful to the data sharing community. *PublisherName* (type: String) [0..1]: Name of the data publisher. *PublisherOrganization* (type: String) [1..1]: Name of the organization the Publisher represents.
RecipientAttributes	Recipient attributes retrieved by the PEP from the users security services. The attributes identify the classes (security level, and caveats) of information that the exchange services are authorized to carry. ***Owned Attributes:*** *CaveatAuthorizations* (type: CaveatType) [0..*]: The caveat authorization(s) for the recipient of the InformationElement(s). *NetworkLocation* (type: String) [0..1]: The network location for the recipient of the InformationElements. *PhysicalLocation* (type: String) [0..1]: The physical location for the recipient of the InformationElements. *RecipientID* (type:) [1..1]: Unique identifier for the recipient of the InformationElement(s).

	RecipientName (type: String) [1..1]: Human readable name for the recipient of the InformationElement(s). *Role* (type: String) [0..*]: The operational role of the recipient of the InformationElements. *SecurityAuthorization* (type: SecurityClassificationType) [0..*]: The security level authorization(s) for the recipient of the InformationElement(s).
RecipientAuthorization	Release or access authorization for a specified InformationElement. ***Owned Attributes:*** *AuthorizationInstruction* (type: String) [0..*]: Handling or release instructions allowing the release of the InformationElement to the recipient. *AuthorizationResponse* (type: AuthorizationResponseType) [1..1]: The result of the PDP's adjudication on the recipient's authorization to receive the InformationElement. *InformationElementID* (type: OctetSeq) [1..1]: Unique identifier for the *InformationElement*. *InformationElementName* (type: String) [0..1]: The human readable name for the *InformationElement*. *RecipientID* (type:) [1..1]: Unique identifier for the recipient of the InformationElement(s). *RecipientName* (type: String) [0..1]: Human readable name for the recipient of the InformationElement(s).
RecipientData	Specified recipients for the InformationElements. ***Owned Attributes:*** *RecipientAddress* (type: string) [0..1]: *RecipientID* (type:) [1..1]: Unique identifier for the specified recipient. *RecipientUserName* (type:) [1..1]: Human readable name for the specified recipient.

ReleaseInstruction	An instruction or set of instructions to the producer or publisher of the information specifying actions to be taken prior to the release of the information. (e.g., encryption requirements). ***Owned Attributes:*** *InstructionType* (type: ReleaseInstructionType) [1..1]: Type of release instruction. *InstructionValue* (type: String) [1..1]: Actual release instruction to be executed by the PEP.
RequestedOperation	A defined operation of function. ***Owned Attributes:*** *Component* (type: ImpactedComponent) [1..*]: Data used to authorize use of a component. *OperationalParam* (type: OperationalParameter) [0..*]: Configuration parameter for the operation. *OperationID* (type: OctetSeq) [1..1]: Unique identifier for the operation or function. *OperationName* (type: String) [1..1]: Human readable name of the operation or function.
SenderAttributes	Sender Authorizations and situational data retrieved by the PEP from the user's security services. ***Owned Attributes:*** *CaveatAuthorizations* (type: CaveatType) [0..*]: The user's caveat authorizations for the sender (/publisher) of the InformationElement(s). *NetworkLocation* (type: String) [0..1]: Network location for the sender (/publisher) of the InformationElement(s). *PhysicalLocation* (type: String) [0..1]: Physical location for the sender (/publisher) of the InformationElement(s). *Role* (type: String) [0..*]: The user's operational role(s) for the sender (/publisher) of the InformationElement(s). *SecurityAuthorization* (type: SecurityClassificationType) [0..*]:

	The user's security level authorizations for the sender (/publisher) of the InformationElement(s). *SenderID* (type: OctetSeq) [1..1]: Unique identifier for the sender (/publisher) of the InformationElement(s). *SenderName* (type: String) [1..1]: Human readable name for the sender (/publisher) of the InformationElement(s).
SenderAuthorization	Authorization granted to the sender to release the information elements. ***Owned Attributes:*** *AuthorizationInstruction* (type: String) [0..*]: Handling or release instructions allowing the release of the InformationElement by the sender. *AuthorizationResponse* (type: AuthorizationResponseType) [1..1]: The result of the PDP's adjudication on the sender's authorization to send (/publish) the InformationElement. *InformationElementID* (type: OctetSeq) [1..1]: Unique identifier for the *InformationElement*. *InformationElementName* (type: String) [0..1]: The human readable name for the *InformationElement*. *SenderID* (type: OctetSeq) [1..1]: Unique identifier for the sender (/publisher) of the InformationElement(s). *SenderName* (type: String) [0..1]: Human readable name for the sender (/publisher) of the InformationElement(s).
SensitivityMetadata	(Optional) Canadian government indicator of the sensitivity and a designated level of protection of an InformationElement. An indicator is applied by the Canadian Government. ***Owned Attributes:*** *KeyToken* (type: Octet) [0..*]: A data element that unique identifies a cryptographic key held in escrow. *ProtectionLevel* (type: ProtectionType) [0..1]: (Optional) Canadian government indicator of the sensitivity of an *InformationElement*. *ReleaseCaveats* (type: CaveatType) [1..*]:

	Warning orders governing the overall releaseability of the InformationElement to selected communities. Recipients must be authorized to receive or access information with the assigned warning orders. An InformationElement identifies each community to which it can be released. *ReleaseSecurityLevel* (type: SecurityClassificationType) [1..*]: A mark (tag or label) identifying the overall security level of the *InformationElement* content.
UID-Attribute	User Identity Attribute. ***Owned Attributes:*** *AttributeName* (type: UserAttributeType) [1..1]: The name of the identity attribute. *AttributeValue* (type: String) [1..1]: The value of the Identity Attribute.
UserAccessAttribute	User attributes and authorizations. ***Owned Attributes:*** *CaveatAuthorizations* (type: CaveatType) [0..*]: The user's caveat authorizations. *NetworkLocation* (type: String) [0..1]: Network location of the user. *PhysicalLocation* (type: String) [0..1]: Physical location of the user. *Role* (type: String) [0..*]: The user's operational role. *SecurityAuthorization* (type: SecurityClassificationType) [0..*]: The user's security level authorization. *UserID* (type:) [1..1]: Unique identifier for the user requesting authorization to perform the function or operation. *UserName* (type: String) [1..1]: Human readable name of the user requesting authorization to perform the function or operation.

UserAuthorizationAttribute	Authorization granted to the specified user(s). ***Owned Attributes:*** *Privilege* (type: Authorization) [1..*]: A data element identifying a privilege or authorization held by a user. *UserID* (type: OctetSeq) [1..1]: Unique identifier for a user.
UserIdentityData	Identifies the characteristic to be used to request additional information from the user's Identity, credential and access management services. ***Owned Attributes:*** *Identifier* (type: OctetSeq) [1..1]: The value of the user identifier. *UserIdentifier* (type: UserID-Type) [1..1]: Identifies the type of *UserID* being provided (e.g., UserName or email address)
UserPrivilege	Data elements that hold data about a user's privileges or authorization. ***Owned Attributes:*** *AttributeName* (type: UserPrivilegeType) [1..1]: The name of the Authorization Attribute. *AttributeValue* (type: String) [1..1]: The value of the Authorization Attribute.

Annex A: ISMB Message XSDs (Informational)

The annex provides a representative set of XML Schema definitions (XSD) for the messages exchange over IEF secure messaging bus. The XSDs will be formalized by the IEF component specifications as they are developed and published. XML one Messaging PSM for IEF inter-component communications.

An XSD for each of the following messages is included in:

- ptc/2019-02-17– IEF/-- ISMB Message XSDs.zip

Ack

An ISMB-Message from a message recipient to the message sender reporting that a specific message was received. The XSD for the "Ack" message can be found in the accompanying machine consumable file: **Ack.xsd**.

CTS-Request

An ISMB-message to the CTS that requests the transformation (encryption or decryption) of the specified InformationElement. The XSD for the "CTS-request" message can be found in the accompanying machine consumable file: **CTS-Request.xsd**.

CTS-Response

An ISMB-message from a CTS to an IEF component responding to a cryptographic transformation request. The XSD for the "CTS-Response" message can be found in the accompanying machine consumable file: **CTS-Response.xsd**.

Message_Data_Types

Collection of complex data types used during IEF communications to standardize inter-component communications. The XSD for ISMB Message Data Types can be found in the accompanying machine consumable file: **Message_Data_Types.xsd**.

Enumerations

Collection of enumeration used during IEF communications to standardize inter-component communications. The XSD for ISMB Enumerations can be found in the accompanying machine consumable file: **Ennumerations.xsd**.

InformationElementMetadata

Representational XSD describing the metadata elements that may be bound to information elements integrated into the PPS data messages. The XSD represents complex data type for metadata attached to PPS data message components. The XSD for Message Metadata can be found in the accompanying machine consumable file: **InformationElementMetadata.xsd.**

ISMB-Message

Messages passed between IEF elements across the Secure Messaging Bus (ISMB) to direct and report on IEF operations. The XSD for the core "ISMB" message can be found in the accompanying machine consumable file: **ISMB-message.xsd**.

ISSG-Request

An ISMB-message issued by an IEF component to the ISSG requesting data / information from a user specified and provisioned security service, e.g.:

- Identity Management;
- Privilege (/attribute / authorization) Management;
- Cryptographic Key Management; or
- TrustMark Registry.

The XSD for the "ISSG-Request" message can be found in the accompanying machine consumable file: ISSG-Request.xsd.

ISSG-Response

An ISMB-message from the ISSG to the IEF Component providing the requested data or Information elements. The XSD for the "ISSG-Response" message can be found in the accompanying machine consumable file: **ISSG-Reponse.xsd**.

Message Metadata

Definition of complex data type defining the metadata elements provided with PPS data messages. The XSD for Message Metadata can be found in the accompanying machine consumable file: **Message_Metadata.xsd**.

PAP-AlertWarning

An ISMB-Message from an IEF Component to the PAP (and the TLS) that informs the user (/administrator) of an unauthorized request to access or release information, unauthorized request to perform an operation, or other error conditions that are being generated through these requests. The XSD for the "PAP-AlertWarning" message can be found in the accompanying machine consumable file: PAP-**AlertWarning.xsd.**

PAP-Command

An ISMB-Message from the PAP to an IEF component directing the component to perform a specified operation or action. The XSD for the "PAP-Command" message can be found in the accompanying machine consumable file: **PAP-Command**.xsd.

PAP-CommandResponse

An ISMB-Message from an IEF component to the PAP providing the results to a PAP's command message. The XSD for the "PAP-CommandResponse" message can be found in the accompanying machine consumable file: **PAP-CommandResponse.xsd**.

PDP-Request

ISMB Message from the PEP to the PDP that conforms to a XACML request message (http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.html).

PDP-Response

ISMB Message from the PDP to the PEP that conforms to a XACML response message (http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.html).

PPS-Publish

An ISMB-message from a PPS to a Messaging-PEP directing the PEP to validate and verify that the PPS is authorized to issue the InformationElement(s) and that the specified recipient(s) are authorized to receive and access the enclosed content. The XSD for the "PPS-Publish" message can be found in the accompanying machine consumable file: **PPS-Publish.xsd**.

PPS-Receive

An ISMB-message from a Messaging-PEP to a PPS forwarding authorized InformationElements from the integrated messaging services or middleware. The following is the XSD for the "" message:

The XSD for the "PPS-Receive" message can be found in the accompanying machine consumable file: **PPS-Receive.xsd**.

PPS-Request

An ISMB-message from a Messaging-PEP to a PPS forwarding authorized information request. The XSD for the "PPS-Request" message can be found in the accompanying machine consumable file: **PPS-Request.xsd**.

Secure Asset Container

XSD for an XML version of the Secure Asset Container (SAC). The SAC is used to bind metadata to an unstructured data element (e.g., a file). The XSD for the "SAC" can be found in the accompanying machine consumable file: **SAC.xsd**.

TLS-LogMessage

An ISMB-Message issued by an IEF component to the Trusted Logging Service(s) describing:

- Operations on InformationElements protected by the IEF;
- Changes to the operating characteristics of an IEF Component; and
- Changes to the Data Policies or Access & Release Control policies.

These messages enable the tamper-resistant recording of IEF operations. The log(s) supports both Security Incident and Event Monitoring (SIEM) and forensic auditing of the environment. Each IEF transaction must be recorded in a manner that resists tampering and alteration of the log records.

The log is intended to maintain a chain-of-custody record for all information elements protected by an IEF implementation. Each log record is encrypted in motion and storage and assigned a chained digital signature. The XSD for the "TLS-LogMessage" message can be found in the accompanying machine consumable file: **TLS-LogMessage.xsd**.

TLS-DataRequest

An ISMB-message from a PAP or other component (e.g., monitoring or auditing) on the ISMB Messaging Bus to request log information from the TLS. The XSD for the "TLS-LogMessage" message can be found in the accompanying machine consumable file: **TLS-DataRequest.xsd**.

TLS-DataResponse

An ISMB-message from a TLS to a PAP or other component (e.g., monitoring or auditing) on the ISMB Messaging Bus that provisions requested log information. The XSD for the "TLS-DataResponse" message can be found in the accompanying machine consumable file: **TLS-DataResponse.xsd**.

Annex B: Policy XSDs (Informational)

The Policy-based Packaging and Processing Service (PPS) policy model is divided into two parts:

1. The Information Exchange Specification Model (ISE) that aligns the information specification with the communications used to transmit authorized information elements (/messages) to specified recipients; and
2. The Semantic Model that defined the rules, constraints and operation used to package and process information elements that are authorized for receipt and release.

These XSDs conform to the semantics specified in the IEPPV specification.

PPS Policies

A representative PPS policy model conforming to the IEPPV specification is provided as part of the machine consumable files provide with the IEF RA. It is provided as informational elements as it cannot be formalized until the PPS specification is issued. These XSDs were developed for the Shared Service Canada IEF technology demonstration project.

The exemplar XML PSM for these policy exchanges are provided in:

- ptc/2019-02-29 – IEF m/--PPS Policy Message XSDs.zip.

Information Exchange Specification XSD

A representative XSD for the IEPPV "Information Exchange Specification" can be found in the accompanying machine consumable file: **IEPPV-InformationExchangeSpecification.xsd**.

Semantic Model XSD

A representative XSD for the IEPPV "Semantic Model" can be found in the accompanying machine consumable file: **IEPPV-SemanticModel.xsd**.

Annex C: Enumerations (Informational)

The following clauses describe the data enumerations used in this reference architecture. They represent a foundational set of elements that may be extended by a user to support a specific operational or business domain. These enumerations are provided as examples – The IEF is agnostic to the specific enumerations adopted by the user (/community).

Command Type Enumerations

The following table identifies and describes the Command Type Enumerations used in the IEF RA.

Table 38 - Command Type Enumerations Attributes

Type Name	Values
ComponentStatusType	Identifies the allowable states for IEF components, component features and policies. *Active*: The specified element is present and in use. *ErrorCondition*: The operating state of the element is an error condition. *Inactive*: The specified element is present but not in use. *Indeterminate*: The operating state of the element is unknown.
DecisionType	Identifies the type of decision being requested. *ConfigurationChange*: Determine whether or not a user is authorized to change an IEF Component Configuration. *InformationRelease*: Determine whether or not an *InformationElement* is releasable to the specified recipients. *OperationAccess*: Determine whether or not a user is authorized to access the specified, platform, device, directory, file-share, file, application, function or other resource in the IEF environment. Primarily relates to PAP operation. *PAP-Access*: Determine whether or not a user is authorized to access the PAP.

	PolicyChange: Determine whether or not a user is authorized to change PDP or PPS policies.
IES-CommandType	Identifies allowable IEF command types. *Activate-IESelement*: Direct the PPS to activate the IES element. *Activate-Policy*: Direct the PPS to activate the IES. *Add-IESelement*: Direct the PPS to add an element to the IES. *Create-IESelement*: Direct the PPS to create an element for the IES using the included parameters, and add it to the IES. *Deactivate-IESelement*: Direct the PPS to deactivate the IES element. *Deactivate-Policy*: Direct the PPS to deactivate the IES. *Delete-IESelement*: Direct the PPS to delete an element to the IES. *Load-IESElements*: Direct the PPS to load the modified IES into the active policy set.
ISSG-RequestType	Identifies allowable types of ISSG data requests. *CryptographicKeys*: Request a new, or retrieve a cryptographic key from the user specified key management services. *IdentityInformation*: Request Identity information for specified users. *OperationalContext*: Request operational context information from the user situational awareness system/services. *Trustmarks*: Request TrustMarks for specified users from the user specified registry.

	UserAuthorizations: Request Attributes (/privileges / authorizations) for specified users.
ISSG-ResponseType	The type of response being provided by the ISSG to an IEF component. *CryptoGraphicKeys*: Message conveying cryptographic keys and tokens for an informationElement. *IdentityInformation*: Message conveying identity information about the sender or recipient(s). *OperationalContext*: Message conveying data pertaining to the operating/mission context in which the IEF is operating. *TrustMarks*: Message conveying TrustMark Information for an external recipient of informationElements. *UserAuthorizations*: Message conveying a sender's or Recipients' authorizations, attributes, access rights, or privileges.
PAP-CommandResponseType	The type of response being provided by an IEF component in response to a **PAP** Command. *Component-ConfigurationReport*: IEF Component configuration. *Component-StatusReport*: IEF Component status report. *ComponentParameterSet*: Persist the current component configuration to the IEF persistent store. *ComponentPolicyset*: Persist the current component policies to the IEF persistent store. Applicable to the PDP and PPS. *ComponentStatus*: Reporting the current overall status of the component and its features.

PAP-CommandType	Type of administrative commands that are issued by the **PAP** to an IEF Component. *Activate-ComponentFeature*: Direct an IEF Component to activate one or more features. *Activate-Policy*: Direct the IEF Component (i.e., **PDP** and **PPS**) to change the state of a specified set of policies in its environment from inactive to active. *Add-Policy*: Direct the IEF Component (i.e., **PDP** or **PPS**) to add the policies in the message to its policy environment. *Archive-Configuration*: Direct an IEF component to persist its current operating configuration to a specified location *Create-ExternalConnection*: Direct the IEF Component to create a communication channel (e.g., domain, topic, queue) to the user specified services. Applies to the **PEP**s and the **ISSG**. *Create-ISMBConnection*: Direct the ISMB to create a communication channel (e.g., domain, topic, queue) to enable communications between one or more IEF components. *Create-Session*: Direct the IEF Component (PEP or ISSG) to connect to a communication channel. *Deactivate-ComponentFeature*: Direct an IEF Component to deactivate one or more features. *Deactivate-Policy*: Direct the IEF Component (i.e., **PDP** and **PPS**) to change the state of a specified set of policies in its environment from active to inactive. *Modify-ComponentParameter*: Direct the IEF component to change the value of one or more of its configuration parameters (setting). *Request-ConfigurationReport*: Direct an IEF component to report on one or more of its operating characteristics (/parameters/features/resources). *Request-Policy*: Direct an IEF component (PDP or PPS) to report on one or more of its operating policies. *Request-StatusReport*:

	Direct an IEF component to report the current operating status of one or more of its features.
PAP-ModeType	Command to set the mode of operation for the PAP. *Advanced*: Provides the operator with full operational control governed by policy and their authorizations. This mode is available where administrators (operators) do not have security clearance equal to or higher than the highest sensitivity of information in the environment. *Debug*: Provide the operator with full operation of the PAP and IEF components. This mode is limited to test and development operations. In this mode the operator has the ability to override policy. *Default*: Provide the operator with a standard set of functions needed to manage and administer an IEF environment. What an administrator has access to is dependent on organizational policy, operating procedures and guidelines. *Disabled*: PAP operator interface is turned off. *Minimal*: Limiting operator controls to the minimum set needed to administer IEF components in operation. This mode is available where administrators (operators) do not have security clearance to the level of the highest sensitivity of information in the environment. This mode would not allow an operator to modify operating or policy configurations.
PDP-PolicyType	Identifies the classes of policy used by the PDP. *ConfigurationChangePolicy*: Category of policies applicable to the adjudication of requests to change IEF component or feature characteristics (configuration parameters). *PAP-AccessPolicy*: Category of policies applicable to the adjudication of requests to access PAP features. *PDP-PolicyChangePolicy*: Category of policies applicable to the adjudication of requests to change **PDP** Policies.

	PPS-PolicyChangePolicy: Category of policies applicable to the adjudication of requests to change **PPS** Policies. *ReleasabilityPolicies*: Category of policies applicable to the adjudication of information element releaseability.

Component Enumerations

The following table identifies and describes the Component Enumerations used in the IEF RA.

Table 39 - Component Enumerations Attributes

Type Name	Values
AlertWarningType	Identifies the types of messages passed to the PAP through the alerts and warnings channel. *Alert*: A notice to the user (IEF Administrator) of an unauthorized action or operation within an IEF Component, or persistent attempts to perform an unauthorized action. *ErrorCondition*: Occurrence of an Error in one of the IEF Components. *Information*: Informational message. *Warning*: A notice that informs the user (IEF Administrator) about issues with the operation of IEF Components.
DecisionPointType	Identifies the allowable IEF decision point types. *PDP*: Identifies the Decision Type as a Policy Decision Point that adjudicates access and release control decisions. *PPS*: Identifies the Decision Type as a Policy-based Packaging and Processing Service that: • Packages (aggregates, transforms, marks, redacts, structures and formats) an InformationElement tailored to the recipient's authorization.

	• Processes (parses, and transforms) DataElements so they can be marshaled to the specified data store.
IEFComponentType	Identifies allowable IEF Component types. *CTS*: Cryptographic Transformation Services (CTS) provide the ability to encrypt and decrypt information elements. *Email-PEP*: Policy Enforcement Point (PEP) provides a proxy that intercepts email messages between the email-client and email-server and determines: • If the specified recipients are authorized to receive the content of the message; and • If the sender of the email, on an IEF protected environment, is authorized to send the content embedded in the email. The PEP ensures that emails sent on an IEF protected environment are appropriately marked by the user. *File-PEP*: Policy Enforcement Point (PEP) provides a proxy that intercepts file-based operations and assures that files are handled in accordance with user specified policy. The PEP also ensured that the files are appropriately marked by the users and the content is always encrypted when at rest and in transit. *IM-PEP*: Policy Enforcement Point (PEP) provides a proxy that intercepts text messages between the Instant Messaging (IM) client and IM-server to determine: • If the specified recipients are authorized to receive the content of the message; and • If the sender on an IEF protected environment, is authorized to send the content of the message (based on user or chat-room included markings). *ISMB*: The IEF Secure Messaging Bus (ISMB) provides the communication pathways between IEF Components. *ISSG*:

	IEF Security Services Gateway (ISSG) provides the features needed to integrate the IEF environment with user specified security services and infrastructure. *Messaging-PEP*: Policy Enforcement Point (PEP) provides a proxy that intercepts messages between the messaging infrastructure and the data stores to determine: • If the specified recipients are authorized to receive the content of the message; and • If the sender of the message, on an IEF protected environment, is authorized to send the content embedded in the message. *PAP*: Policy Administration Point (PAP) provides a user interface and features that enable an authorized user to manage and administrate IEF components. *PDP*: Policy Decision Point (PDP) provides the business logic and processes to adjudicate access and release control decisions. *PPS*: Policy-based Packaging and Processing Service (PPS) provides the ability to: • Package (aggregates, transforms, marks, redacts, structures and formats) an InformationElement tailored to the recipient's authorization. • Process (parses, and transforms) DataElements so they can be marshaled to the specified data store.
IES-RoleType	Identifies the role of a PPS within a specific Information Exchange Specification (Agreement). *DataConsumer*: Indicates that the connection is a receiver (/consumer / recipient / reader) of information from the community. *DataConsumer-Producer*: Indicates that the connection is a receiver (/consumer / recipient / reader) of information from, provider (/publisher, producer / writer) into, the community. *DataProducer*: Indicates that the connection is a provider (/publisher, producer / writer) of information into the community.

ISMB-MessageType	Metadata tag that identifies the type of message being exchanged. *CTS-LogMessage*: **CTS** log message to the **TLS**. *CTS-Request*: PEP message to the CTS requesting that an InformationElement be encrypted or decrypted. *CTS-Response*: **CTS** message to the **PEP** with the transformed InformationElement following the encryption or decryption process. *ISSG-LogMessage*: **ISSG** log message to the **TLS**. *ISSG-SecurityServiceRequest*: IEF component request to the user's security infrastructure for information (e.g., User Privileges, Cryptographic Keys, and operational context). This request is made to the **ISSG** that provides a single integration point for user specified security services. *ISSG-SecurityServiceResponse*: **ISSG** response to an IEF component, with the information requested from the user's security infrastructure. *PAP-AlertWarning*: IEF Component message to the **PAP** (and the **TLS**) to inform the PAP (/ user /administrator) that there are unauthorized requests being made to the component, or error conditions are being generated through a request to the components. *PAP-Command*: **PAP** command message to an IEF Component. *PAP-CommandResponse*: IEF Component response to a **PAP** command. *PAP-LogMessage*: **PAP** log message to the **TLS**. *PDP-LogMessage*: **PDP** log message to the **TLS**. *PDP-Request*: IEF component message to the **PDP** that requests authorization to perform a specified operation. *PDP-Response*: PDP message to an IEF Component providing its determination whether or not the requested action is authorized. *PEP-LogMessage*:

	PEP log message to the **TLS**. *PPS-LogMessage*: **PPS** log message to the **TLS**. *PPS-ReceiveMessage*: **PEP** message to the **PPS** conveying information to be processes by the PEP. *PPS-Request*: PEP message to the PPS requesting information. *PPS-Response*: PEP message issued in response to a request for information. *PPS-SendMessage*: PPS message to the PEP conveying information to be released using the PEP's supported communication channels. *TLS-DataRequest*: Message to the TLS Requesting Data. *TLS-DataResponse*: Message from the TLS to a Monitoring or logging services containing log reports for a specified time period for a specified component. *TLS-LogMessage*: Message to the Trusted Logging Service.
OperationStatusType	Identifies the operating states for an IEF component or component feature. *DegradedOperation*: One or more specified features is not performing as expected. *NonStandardOperation*: An information or user operation has caused a non-standard, or unexpected event or result. *NormalOperation*: The IEF Component or feature is operating within expected parameters. *OperationalFailure*: An IEF component or feature is not responding.
PolicyAdjudicationType	*InformationRelease*: Request adjudication and authorization of a PEP request to release an InformationElement to the specified recipient. *InformationRequest*: Request adjudication and authorization of a PEP request to request information from a specified source.

	OperationRequest: Request adjudication and authorization of a PAP (/ operator / Administrator) to execute a specified operation (/ issue a command) to a specified IEF component.
PPS-RequestType	Identifies the type of request being made to a PPS. *IES-Modification*: A user is requesting a change to an Information Exchange contract (Specification). *IES-ParticipationRequest*: A user is requesting participation in an existing Information Exchange contract (Specification). *InformationElementPublication*: Request a single instance of an information element. *InformationElementsOfType*: Request a list of InformationElements of type being maintained by the PPS. The list is issued with the Name, Identifier and sensitivity of the information regarding the InformationElement.
PPS-ResponseType	Identifies the type of response being made by a PPS. *ErrorCondition*: Indicates that a request or action resulted in an error condition. *InformationElementInstance*: PPS is providing authorized information Elements.

Instruction Type Enumerations

The following table identifies and describes the Instruction Type Enumerations used in the IEF RA.

Table 40 - Instruction Type Enumerations Attributes

Type Name	Values
HandlingInstructionType	Valid *HandlingInstruction* types for the recipient of an *InformationElement*. *Acknowledge*: An instruction to the recipient of an information exchange directing the issuance of an acknowledgment to the receipt of the information to the provider of the information. *AttachmentFormatting*: Identifies the formatting of one or more of the message attachments. *BinaryDataRendering*: An instruction to the recipient of an information exchange defining the rules for rendering or displaying binary data. *DiscardAfter*: An instruction to the recipient of an information exchange specifying the rules for destruction or discarding of data included within an information package or message. This version directs the recipient to discard the information after a specified date and time. *DiscardBefore*: An instruction to the recipient of an information exchange specifying the rules for destruction or discarding of data included within an information package or message. This version directs the recipient to discard the information before a specified date and time. *DoNotForward*: An instruction to the recipient of an information exchange specifying that the information must not be forwarded to any other recipient or destination. *DoNotPersist*: An instruction to the recipient of an information exchange directing the recipient not to persist any of the information or data in a payload or message. *ForwardInstruction*: An instruction to the recipient of an information exchange to forward the information to authorized recipients in accordance with any provided list, or in accordance with specified information sharing agreements. *HandleInAccordanceWith*: An instruction to the recipient of an information exchange, directing the recipient to handle the information in the message in accordance with the instructions in a specified document. *MessageProtocol*:

	Instruction to the recipient of an information exchange that identifies the InformationElement formatting protocol (e.g., XSD). *StoreEncypted*: An instruction to the recipient of an information exchange directing the recipient to encrypt the information in the message before storing it.
ReleaseInstructionType	Release Instruction Types. *CryptographicKey*: Provides the token for the service to retrieve the cryptographic Key for the information element(s). *CryptographicServiceID*: Specifies the cryptographic service to be used by unique identifier. *CryptographicServiceName*: Specifies the cryptographic service to be used by name. *Encrypt*: An instruction or set of instructions to the producer of the information directing that the message or elements of the message need to be encrypted prior to release. *FormatInstruction*: Specifies a formatting instruction for the information element(s). *QualityOfService*: An instruction or set of instructions to the producer or publisher of the information specifying the quality of service requirements for the exchange of the information. *ReleaseChannel*: Specifies the communication channel for the release of the informationElement(s). *ReleaseProtocol*: Specifies the messaging protocol to use when sending the information elements.

Policy Enumerations

The following table identifies and describes the Policy Enumerations used in the IEF RA.

Table 41 - Policy Enumerations Attributes

Type Name	Values
AuthorizationResponseType	Identifies the type of response being provided by the PDP. This is the result of the policy evaluation process based on the submitted policy request by the user and the current security policy set. The decision is encoded into a decision element in the XACML response message. As per the XACML standard, this field can take one of three values: “Permit” (action is permitted), “Deny” (action is denied) or “Error”. When any error is encountered by the PDP (e.g. policy database is offline) an “Error” decision is returned and it is the responsibility of the PEP to handle this situation. *Deny*: Deny the requested action. *Error*: An error condition resulted from the evaluation. *Indeterminate*: No determination can be made. *NotApplicable*: Condition does not apply to the authorization request. *Permit*: Permit the requested action.
DecisionType	Identifies the type of decision being requested. *ConfigurationChange*: Determine whether or not a user is authorized to change an IEF Component Configuration. *InformationRelease*: Determine whether or not an *InformationElement* is releasable to the specified recipients. *OperationAccess*: Determine whether or not a user is authorized to access the specified, platform, device, directory, file-share, file, application, function or other resource in the IEF environment. Primarily relates to PAP operation. *PAP-Access*: Determine whether or not a user is authorized to access the PAP. *PolicyChange*: Determine whether or not a user is authorized to change PDP or PPS policies.

PDP-PolicyType	Identifies the classes of policy used by the PDP. *ConfigurationChangePolicy*: Category of policies applicable to the adjudication of requests to change IEF component or feature characteristics (configuration parameters). *PAP-AccessPolicy*: Category of policies applicable to the adjudication of requests to access PAP features. *PDP-PolicyChangePolicy*: Category of policies applicable to the adjudication of requests to change **PDP** Policies. *PPS-PolicyChangePolicy*: Category of policies applicable to the adjudication of requests to change **PPS** Policies. *ReleasabilityPolicies*: Category of policies applicable to the adjudication of information element releaseability.
PPS-PolicyType	Identifies the classes or policy used by the PPS. *ExchangeElementPolicy*: Category of policies defining the rules and constraints for assigning filters to SemanticElements and then to the exchange element. *ExchangePolicy*: Category of policies defining the rules and constraints for assigning exchange elements to exchange agreements. *SemanticPolicy*: Category of policies defining the rules and constraints for aggregating semantically complete data sets. Includes both the assembly of transactional and semantic elements described in the IEPPV Specification.
UserID-Type	Types of information used to access user attributes. *EmailAddress*: Email Address for the User. *UserID*: Unique Identifier for the user. *UserName*: User name for the User.

Security Enumerations

The following table identifies and describes the Security Enumerations used in the IEF RA.

Table 42 - Security Enumerations Attributes

Type Name	Values
CaveatType	Identifies types of Warning orders (Caveats) that are attached to protected or classified InformationElements to designate specific individuals, groups or communities that can access or receive the content of the InformationElement or Message. The identified caveats form a small subset of those used by Military, National and Public Security and the Legal community. This list is meant to be extended. *Commercial-in-Confidence*: The information contains elements that are proprietary (e.g., Intellectual Property, or trade secrets) to the data owner/creator and cannot be released without their approval. *Eyes-Only*: Information meant to be seen only by the person to whom it is directed. *FVEY*: Classified information restricted to the Five Eyes Community. Five Eyes, often abbreviated as FVEY, are an intelligence alliance comprising Australia, Canada, New Zealand, the United Kingdom and the United States (UKUS). These countries are bound by the multilateral UKUSA Agreement, a treaty for joint cooperation in signals intelligence. *Legally-Significant*: The information contains elements that are pertinent to a legal proceeding and cannot be released without special approvals. *NATO*: Classified information restricted to NATO countries and personnel. *Private*: The information contains elements that include Personal Identifying Information (PII) or other personal information. *Sensitive-But-Unclassified*: Unauthorized release of such material would cause "serious damage" to national security.

CTS-RequestType	Identification of allowable CTS operations. *DecryptElement*: Request that the CTS decrypt (decode) the specified InformationElement to its original form so authorized parties (with the decryption key) can read it. *EncryptElement*: Request that the CTS encrypt (encode) the specified InformationElement in such a way that only authorized parties (with the decryption key) can read it. *NewSAC*: Request that the CTS create a new SAC for the InformationElement using a new cryptographic Key.
ProtectionType	Identifies the protection levels for sensitive but unclassified (SBU) information. Protected Information is sensitive information which requires safeguarding but does not apply to the national interest. Its unauthorized release, destruction, removal, or modification could reasonably be expected to cause potential damage to a reputation, wrong an image, hurt or harm a person, corporation or government. Examples include personal information such as pay data and medical records, business information such as trade secrets. *Protected-A*: Applies to information that, if compromised, could reasonably be expected to cause injury or embarrassment outside the national interest, for example, disclosure of an exact salary figure, an individual's date of birth or even information pertaining to contracts and tenders. *Protected-B*: Applies to information that, if compromised, could reasonably be expected to cause serious injury outside the national interest, for example, loss of reputation or competitive advantage. Examples of this would include Medical, criminal or psychiatric records, trade secrets, and risk assessments *Protected-C*: Applies to the very limited amount of information that, if compromised, could reasonably be expected to cause extremely grave injury to an individual, the government or the national interest. *Public*: Applied to information that is expected to be publicly available.

SecurityClassificationType	Security level assigned to a government document, file, or record based on the sensitivity or secrecy of the information content. *Confidential*: Unauthorized or public release of such material would cause "damage" or be "prejudicial" to national security or national interests. *Restricted*: Unauthorized or public release of such material would cause "undesirable effects" to national interests. Some countries do not have such a classification. *Secret*: Unauthorized or public release of such material would cause "serious damage" to national security, public safety or national interests. *Top-Secret*: Unauthorized or public release of such material would cause "exceptionally grave damage" to national security, public safety or national interests. *Unclassified*: Used for government documents that do not have a classification listed above. Such documents can sometimes be viewed by those without security clearance.
ServiceStatusType	Identifies valid service response status types. *Error*: The request was unsuccessful - error message attached. *Successful*: The request was successful - data attached.

Security Service Enumerations

The following table identifies and describes the Security Service Enumerations used in the IEF RA.

Table 43 - Security Service Enumerations Attributes	
Type Name	**Values**
ISSG-ResponseType	The type of response being provided by the ISSG to an IEF component. *CryptoGraphicKeys*: Message conveying cryptographic keys and tokens for an informationElement. *IdentityInformation*: Message conveying identity information about the sender or recipient(s). *OperationalContext*: Message conveying data pertaining to the operating/mission context in which the IEF is operating. *TrustMarks*: Message conveying TrustMark Information for an external recipient of informationElements. *UserAuthorizations*: Message conveying a sender's or Recipients' authorizations, attributes, access rights, or privileges.
KMS-RequestType	The type of request being made to the users Key Management Service(s). *GenerateKey*: ISSG request to the user specified Key Management Services to create a cryptographic key for an InformationElement. *GenerateStore*: ISSG request to the user specified Key Management Services to create a store in the escrow service for a generated cryptographic key and return a token with which the key may be recovered. *RetrieveKey*: Retrieve an existing key from the key escrow system. *StoreKey*: ISSG request to the user specified Key Management Services to store a cryptographic key in the escrow system and return a token with which the key may be recovered.

KMS-ResponseType	The type of response being provided by the Key Management Services to the IEF Component. *KeyAndToken*: The returned key and token from the escrow system. *KeyOnly*: The returned key from the escrow system. *Token*: The returned token from the escrow system.
MetadataGroupType	Identifies grouping of metadata that may be required for various *InformationElements*. *AttachmentMetadata*: Set of metadata elements to be bound to a message attachment. *DiscoveryMetadata*: Set of metadata elements that aid in the discovery of an informationElement. *EmailMetadata*: Set of metadata elements to be bound to an Email message. *FileMetadata*: Set of metadata elements to be bound to a file. *MessageMetadata*: Set of metadata elements to be bound to a message. *PackageMetadata*: Set of metadata elements to be bound to a message package. *PayloadMetadata*: Set of metadata elements to be bound to a message payload.
MetadataType	Identifies the role of metadata in terms of the InformationElement it describes. *DataCreatorMetadata*: Data describing the creator of the *InformationElement*. *DataOwnerMetadata*: Data describing the owner of the *InformationElement*. *DataSensitivity*: Metadata describing the sensitivity (Privacy, confidentiality, classification or legal significance) of the *InformationElement*.

	Handling Instruction: Data describing the specific handling instructions for the *InformationElement*. *PublisherMetadata*: Data describing the publisher of the *InformationElement*. *ReleaseInstruction*: Data describing the specific release instructions for the *InformationElement*.
OperationalContextAttributeType	Data type providing information about the IEF's operating context that may affect policy decisions. *CoalitionType*: Current Coalition Type. *MissionType*: Current Mission Type. *OperationalPhase*: Current Operational phase. *OperationalResponsibility*: Current organizational responsibility. *OperationalRole*: Current operational roles. *OperationType*: Current Operation Type. *ThreatLevel*: Current operational threat level.
SecurityAttributeType	Identifies the types of attributes (/privileges / authorizations) being sought. *ApplicationAttribute*: Attributes pertaining to application access and constraints. *CommunicationAttribute*: Attributes pertaining to communications access and constraints. *DeviceAttribute*: Attributes pertaining to device access and constraints. *DirectoryAttribute*: Attributes pertaining to directory access and constraints. *InformationAttribute*: Attributes pertaining to InformationElement access.

	OperationalAttribute: Attributes pertaining to IEF operation/function access. Primarily pertaining to Administration functions.
UserAttributeType	List of standardized user identity attributes that may be used to authorize activity or information access. *COI-Affiliation*: The user's affiliation with a community-of-interest (COI) or community-of-practice (COP). *Firstname*: User's first or given name *GroupAffiliation*: User's affiliation or relationship with a collection of entities that share common functions, behaviors, rights, obligations, beliefs, and norms. *LastName*: User's Sur or last name. *LoginName*: Concatenated Name of the user (e.g., FirstName + LastName). *MailAddress*: User's email address for the electronic mailbox attribute following the syntax specified in RFC 822. *MiddleName*: User's middle name. *MissionAffiliation*: User's affiliation with a specified mission. *OperationalAffiliation*: User's affiliation with a specified operation. *OrganizationAffiliation*: User's affiliation or relationship to an organization or agency. *Role*: User's customary function (s) for the organization or community that connects the user to known behaviors, rights, obligations, beliefs, and norms. *UserID*:

	The network identification of the user for the purposes of inter and intra organizational authentication. A persistent, privacy-preserving identifier for a principal shared between a pair of coordinating entities.
UserPrivilegeType	User privilege and authorization types. *ApplicationAuthorization*: Authorization to access or operate the specified application. *CaveatAuthorization*: Authorization to access or receive information marked at that security level. *ChannelAuthorization*: Authorization to access the specified communication channel. *DeviceAuthorization*: Authorization to access the specified device. *DirectoryAuthorization*: Authorization to access the specified Directory or folder. *NetworkAuthorization*: Authorization to access the specified network. *PII-Authorization*: Authorization to access Personal Identifying Information (PII) information. *ProtectionLevelAuthorization*: Authorization to access or receive information marked at that protection level. *SecurityAutorization*: Authorization to access or receive information marked at that security level.

Annex D: Sequence Diagrams (Informational)

The following diagrams illustrate and describe several representative interaction sequences between the IEF components. The following clauses illustrate and describe a representative set of interactions between IEF components while authorizing the receipt or release of information elements (files, emails, text messages and/or structured messages). These diagrams are not intended to prescribe the operations of each component that is documented in a separate specification, E.g.:

- The PDP that can be implemented using the XACML specifications or equivalent; or
- The PPS defined in a separate IEF specification.

The sequence diagrams in this Annex illustrate a representative set of interactions between IEF components during their adjudication and enforcement user security policy for structure messaging, email exchange, file sharing and instant messaging. The diagrams are based on configurations from successful pilot implementations of IEF components and do not characterized all possible implementations. This reference architecture is intended to be extensible and provide for the integration of community and/or user defined additions and/or enhancements to the IEF defined data centric security services.

The choices in the selection of components and specifications guide the sequencing of activities within and between the components. These sequence diagrams are to illustrate how the IEF components can interoperate to assure that:

1. Each information element protected by the IEF is authorized for release to each specified recipient;
2. Each information element received by a PEP is authorized for inclusion (storage, integration, fusion) into the IEF Protected Information Store(s);
3. Each transaction is logged for real-time monitoring and/or forensic auditing; and
4. Each transaction that does not conform to data, access or release policies is reported to the user (administrator).

Given the possible variations in user policies, these diagrams are provided as informational to this reference architecture.

Embedded PPS Sequence

The following clauses provide representative sequence diagrams for IEF components enforcing information messaging protections.

PPS Receive Authorization

The following figure illustrates a representational exchange of information between IEF Components when validating that a user is authorized to receive information from an external source. The Messaging-PEP orchestrates the local environment (network, systems, devices and users) is authorized to access, store, and process received information elements.

*Note: there are multiple paths through the authorizations process depending on:

- The number of information elements, being published simultaneously;
- The complexity of the message structure (e.g., digest, packages, payloads and attachments);
- The number of recipients (e.g., Single topic or queue, or a list of individual recipients);
- The capabilities of each of the selected IEF components;

- The availability and fidelity of the user's (e.g., network, devices, systems, services, and users) authorizations, privileges and attributes; and
- The complexity and fidelity of the user's own policies.

Many of these considerations are addressed in the individual component specifications.

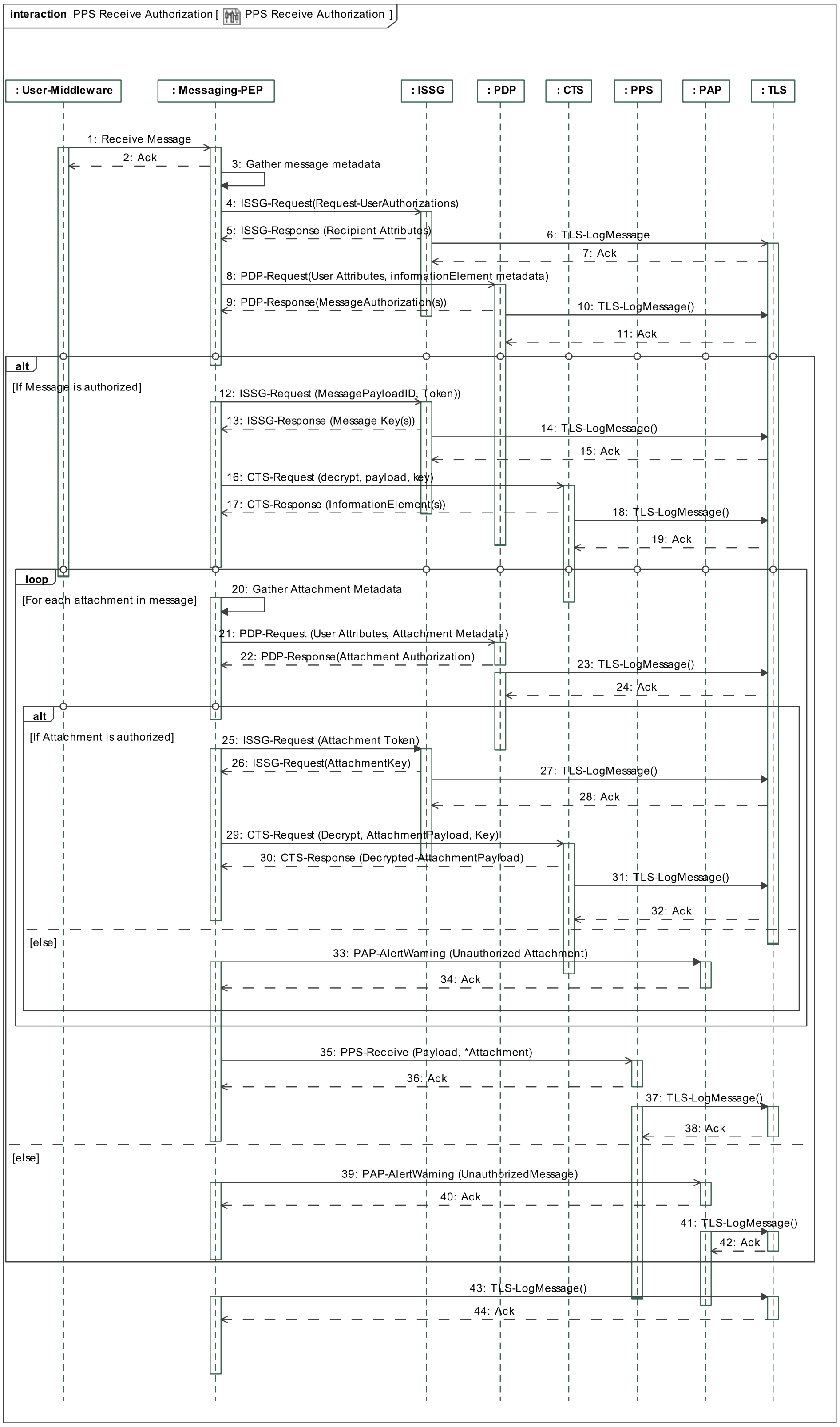

interaction PPS Receive Authorization [PPS Receive Authorization]
: User-Middleware
: Messaging-PEP
: ISSG
: PDP
: CTS
: PPS
: PAP
: TLS
1: Receive Message
2: Ack
3: Gather message metadata
4: ISSG-Request(Request-UserAuthorizations)
5: ISSG-Response (Recipient Attributes)
6: TLS-LogMessage
7: Ack
8: PDP-Request(User Attributes, informationElement metadata)
9: PDP-Response(MessageAuthorization(s))
10: TLS-LogMessage()
11: Ack
alt
[If Message is authorized]
12: ISSG-Request (MessagePayloadID, Token))
13: ISSG-Response (Message Key(s))
14: TLS-LogMessage()
15: Ack
16: CTS-Request (decrypt, payload, key)
17: CTS-Response (InformationElement(s))
18: TLS-LogMessage()
19: Ack
loop
[For each attachment in message]
20: Gather Attachment Metadata
21: PDP-Request (User Attributes, Attachment Metadata)
22: PDP-Response(Attachment Authorization)
23: TLS-LogMessage()
24: Ack
alt
[If Attachment is authorized]
25: ISSG-Request (Attachment Token)
26: ISSG-Request(AttachmentKey)
27: TLS-LogMessage()
28: Ack
29: CTS-Request (Decrypt, AttachmentPayload, Key)
30: CTS-Response (Decrypted-AttachmentPayload)
31: TLS-LogMessage()
32: Ack
[else]
33: PAP-AlertWarning (Unauthorized Attachment)
34: Ack
35: PPS-Receive (Payload, *Attachment)
36: Ack
37: TLS-LogMessage()
38: Ack
[else]
39: PAP-AlertWarning (UnauthorizedMessage)
40: Ack
41: TLS-LogMessage()
42: Ack
43: TLS-LogMessage()
44: Ack

Figure 49 - PPS Receive Authorization

The following table identifies and describes the interactions between IEF components as illustrated.

Table 44 - PPS Receive Authorization Messages

#	Messages			
1	Name	Receive Message		
	Description	The Messaging PEP intercepts all messages for local PPS instances, authenticates the message, and authorizes its release to the PPS for processing. The PEP assures the local environment is authorized to receive the content of the message.		
	Sender	User-Middleware	Receiver	Messaging-PEP
2	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) Acknowledge the receipt of the message to provide non-repudiation for the receipt of messages during a forensic audit. Acknowledgment of messages is optional, as the user may use the non-repudiation capabilities of an ISMB infrastructure (e.g., DDS) to assure delivery, quality of service and logging (real-time monitoring and forensic auditing).		
	Sender	Messaging-PEP	Receiver	User-Middleware
3	Name	Gather message metadata		
	Description	The Messaging-PEP parses the inbound message and extracts the metadata elements that specify the authorization required to receive and process the enclosed information element(s). The metadata required for receipt authorization may include: • [0..*] Security Level authorized to access the information; • [0..*] Caveats authorizations required to access the information; • [0..*] Protection Level authorizations required to access the information; • [0..*] Handling instructions; and • [0..1] Cryptographic Token. How many of these metadata elements are required, received and used depends on: • The availability and fidelity of the information from the user provisioned security services; • The needs and capabilities of the PDP; and • The fidelity of the user's or community's ISS policies.		
	Sender	Messaging-PEP	Receiver	Messaging-PEP
4	Name	ISSG-Request (Request-UserAuthorizations)		
	Description	The Messaging-PEP packages and issues an ISSG-Request to get the recipients (local IEF instance) security attributes (/privileges / authorizations) from the user's infrastructure. The PEP then issues the ISSG-Request to the ISSG to be actioned.		
	Sender	Messaging-PEP	Receiver	ISSG

Table 44 - PPS Receive Authorization Messages

#	Messages			
5	Name	ISSG-Response (Recipient Attributes)		
	Description	The ISSG transforms the ISSG-Request into the appropriate format for the specified service (e.g., ICAM service or TrustMark Registry), packages the request into a message and issues it to the specified services using the local messaging infrastructure. Upon receipt of the user's attributes, the ISSG transforms the information into the form required by the PEP, packages the information as an ISSG-Response message, and issues the message to the PEP using the ISMB.		
	Sender	ISSG	Receiver	Messaging-PEP
6	Name	TLS-LogMessage		
	Description	The ISSG records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	ISSG	Receiver	TLS
7	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to ISSG.		
	Sender	TLS	Receiver	ISSG
8	Name	PDP-Request (User Attributes, informationElement metadata)		
	Description	The Messaging-PEP gathers the user attributes and information element (message payload) metadata, packages the information as a PDP-Request and issues the request to the PDP for adjudication and determination.		
	Sender	Messaging-PEP	Receiver	PDP
9	Name	PDP-Response (MessageAuthorization(s))		
	Description	The PDP packages its determination(s) for each information element as a PDP-Response message and issues the message to the PEP.		
	Sender	PDP	Receiver	Messaging-PEP
10	Name	TLS-LogMessage ()		
	Description	The PDP records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	PDP	Receiver	TLS
11	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to PDP.		
	Sender	TLS	Receiver	PDP
12	Name	ISSG-Request (MessagePayloadID, Token))		
	Description	If the message is authorized for receipt, the PEP gathers the key-token from the message metadata, packages the token as an ISSG request to get the cryptographic key from the users' key management services (escrow). The ISSG transforms the PEP request into the appropriate format to the specified user service (e.g., KMS).		
	Sender	Messaging-PEP	Receiver	ISSG
13	Name	ISSG-Response (Message Key(s))		

Table 44 - PPS Receive Authorization Messages

#	Messages			
	Description	Upon receipt of the key from the users KMS, the ISSG packages and issues an ISSG-Response message containing the message key.		
	Sender	ISSG	Receiver	Messaging-PEP
14	Name	TLS-LogMessage ()		
	Description	The ISSG records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	ISSG	Receiver	TLS
15	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to ISSG.		
	Sender	TLS	Receiver	ISSG
16	Name	CTS-Request (decrypt, payload, key)		
	Description	The Messaging-PEP gathers the message payload(s) and the cryptographic key(s), packages them as a CTS-Request and issues the message to the CTS to have the message decrypted.		
	Sender	Messaging-PEP	Receiver	CTS
17	Name	CTS-Response (InformationElement(s))		
	Description	After transforming (decrypting) the payload, the CTS packages the transformed information element as a CTS-Response and issues the message to the PEP.		
	Sender	CTS	Receiver	Messaging-PEP
18	Name	TLS-LogMessage ()		
	Description	The CTS records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	CTS	Receiver	TLS
19	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to CTS.		
	Sender	TLS	Receiver	CTS
20	Name	Gather Attachment Metadata		
	Description	For the purposes of this sequence diagram, each attachment is packaged within a Secure Access Container (SAC). The PEP unpacks the SAC metadata and extracts the sensitivity markings and handling instructions included for the attachment. The metadata required for receipt authorization may include: • [0..*] Security Level authorized to access the information; • [0..*] Caveats authorizations required to access the information; • [0..*] Protection Level authorizations required to access the information; • [0..*] Handling instructions; and • [0..1] Cryptographic Token. How many of these metadata elements are required, received and used depends on:		

Table 44 - PPS Receive Authorization Messages

#	Messages			
		• The availability and fidelity of the information from the user provisioned security services; • The needs and capabilities of the PDP; and • The fidelity of the user's or community's ISS policies.		
	Sender	Messaging-PEP	Receiver	Messaging-PEP
21	Name	PDP-Request (User Attributes, Attachment Metadata)		
	Description	The Messaging-PEP gathers the user attributes and information element (Attachment) metadata, packages the information as a PDP-Request and issues the request to the PDP for adjudication and determination.		
	Sender	Messaging-PEP	Receiver	PDP
22	Name	PDP-Response(Attachment Authorization)		
	Description	The PDP packages its determination(s) for the information element (attachment) as a PDP-Response message and issues the message to the PEP.		
	Sender	PDP	Receiver	Messaging-PEP
23	Name	TLS-LogMessage()		
	Description	The PDP records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	PDP	Receiver	TLS
24	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to PDP.		
	Sender	TLS	Receiver	PDP
25	Name	ISSG-Request (Attachment Token)		
	Description	If the attachment is authorized for receipt, the PEP gathers the key-token from the message metadata, packages the token as an ISSG-Request to the ISSG to get the cryptographic key from the users key management services (escrow).		
	Sender	Messaging-PEP	Receiver	ISSG
26	Name	ISSG-Request(AttachmentKey)		
	Description	The ISSG transforms the PEP request into the appropriate format to the specified user service (e.g., KMS). Upon receipt of the key from the users KMS, the ISSG packages and issues an ISSG-Response message containing the message key.		
	Sender	ISSG	Receiver	Messaging-PEP
27	Name	TLS-LogMessage()		
	Description	The ISSG records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	ISSG	Receiver	TLS
28	Name	Ack		

Table 44 - PPS Receive Authorization Messages

#	Messages			
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to ISSG.		
	Sender	TLS	Receiver	ISSG
29	Name	CTS-Request (Decrypt, AttachmentPayload, Key)		
	Description	If local Data Store is authorized to receive the attachment, the PEP gathers the Attachment payload and the cryptographic key, packages them as a CTS-Request and issues it to the CTS for processing (decryption).		
	Sender	Messaging-PEP	Receiver	CTS
30	Name	CTS-Response (Decrypted-AttachmentPayload)		
	Description	After transforming (decrypting) the payload, the CTS packages the transformed information element (as a CTS-Response and issues the message to the PEP.		
	Sender	CTS	Receiver	Messaging-PEP
31	Name	TLS-LogMessage ()		
	Description	The CTS records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	CTS	Receiver	TLS
32	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to CTS.		
	Sender	TLS	Receiver	CTS
33	Name	PAP-AlertWarning (Unauthorized Attachment)		
	Description	If the Messaging-PEP receives an attachment for which the local PPS is not authorized to receive, the PEP does not process (decrypt) the attachment, and prepares and issues a PAP-AlertWarning message to the PAP (/user).		
	Sender	Messaging-PEP	Receiver	PAP
34	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The PAP acknowledges the receipt of the message to PEP.		
	Sender	PAP	Receiver	Messaging-PEP
35	Name	PPS-Receive (Payload, *Attachment)		
	Description	The PEP packages all the authorized information elements, packages a PPS-Receive message and issues it to the PPS for processing.		
	Sender	Messaging-PEP	Receiver	PPS
36	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The PPS acknowledges the receipt of the message to PEP.		
	Sender	PPS	Receiver	Messaging-PEP

Table 44 - PPS Receive Authorization Messages

#	Messages			
37	Name	TLS-LogMessage ()		
	Description	The PPS records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	PPS	Receiver	TLS
38	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to PPS.		
	Sender	TLS	Receiver	PPS
39	Name	PAP-AlertWarning (UnauthorizedMessage)		
	Description	If the Messaging-PEP receives a message for which the local PPS is not authorized to receive, the PEP does not process (decrypt) any elements in the message, and prepares and issues a PAP-AlertWarning message to the PAP (/user).		
	Sender	Messaging-PEP	Receiver	PAP
40	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The PAP acknowledges the receipt of the message to PEP.		
	Sender	PAP	Receiver	Messaging-PEP
41	Name	TLS-LogMessage ()		
	Description	The PAP records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	PAP	Receiver	TLS
42	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to PAP.		
	Sender	TLS	Receiver	PAP
43	Name	TLS-LogMessage ()		
	Description	The PEP records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	Messaging-PEP	Receiver	TLS
44	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to PEP.		
	Sender	TLS	Receiver	Messaging-PEP

PPS Publish Authorization

The following figure illustrates a representational exchange of information between IEF Components when validating that a user is authorized to publish the specified set of information elements to a specified communication channel. The Messaging-PEP orchestrates the process for assuring that the published information is releasable to the recipient(s) over the specified communication channel.

*Note: there are multiple paths through the authorizations process depending on:

- The number of information elements, being published simultaneously;
- The complexity of the message structure (e.g., digest, packages, payloads and attachments);
- The number of recipients (e.g., Single topic or queue, or a list of individual recipients);
- The capabilities of each of the selected IEF components;
- The availability and fidelity of the user's (e.g., network, devices, systems, services, and users) authorizations, privileges and attributes; and
- The complexity and fidelity of the user's own policies.

Many of these considerations will be addressed in the individual component specifications.

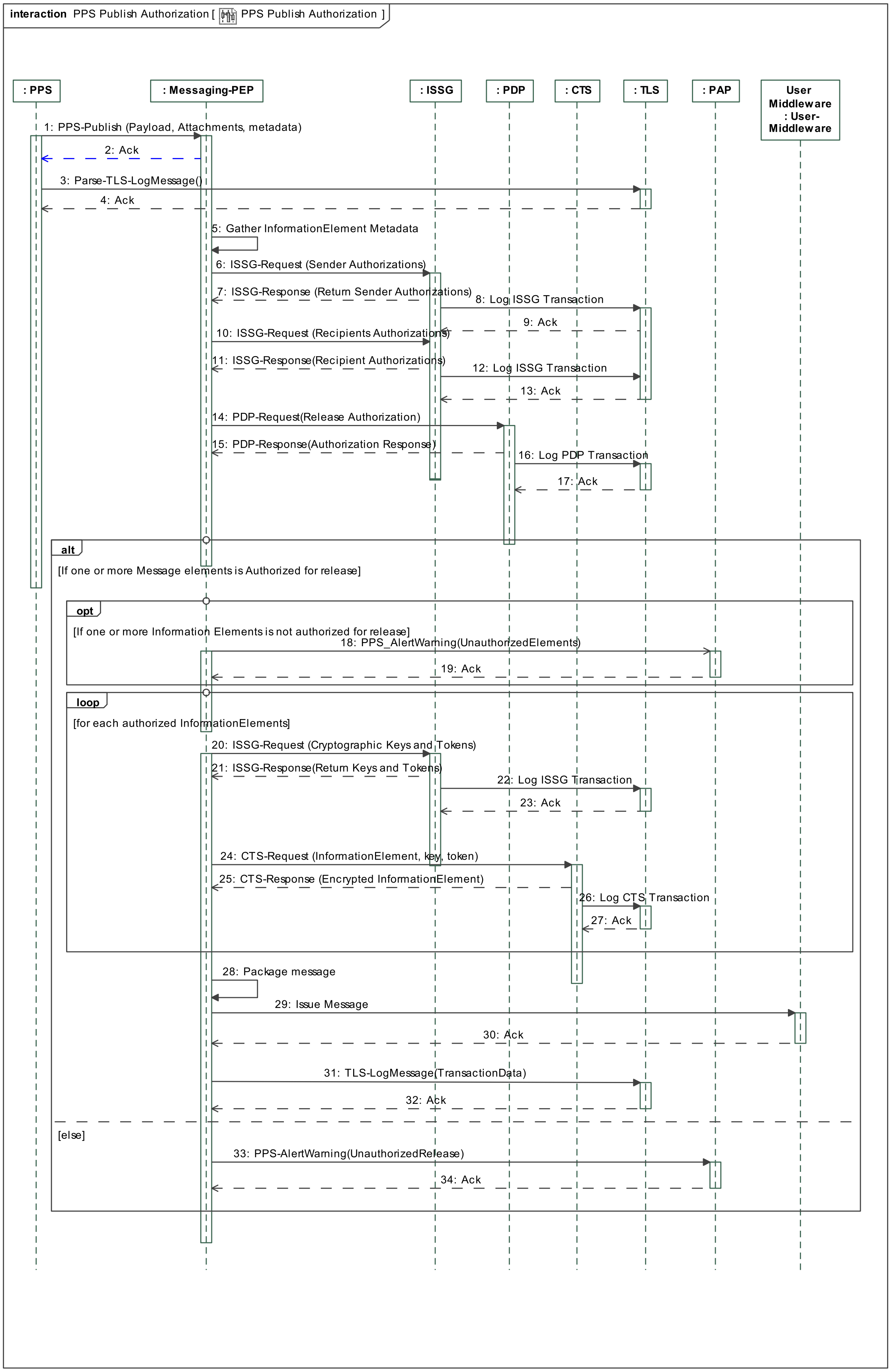

interaction PPS Publish Authorization [PPS Publish Authorization]
: PPS
: Messaging-PEP
: ISSG
: PDP
: CTS
: TLS
: PAP
User Middleware : User-Middleware
1: PPS-Publish (Payload, Attachments, metadata)
2: Ack
3: Parse-TLS-LogMessage()
4: Ack
5: Gather InformationElement Metadata
6: ISSG-Request (Sender Authorizations)
7: ISSG-Response (Return Sender Authorizations)
8: Log ISSG Transaction
9: Ack
10: ISSG-Request (Recipients Authorizations)
11: ISSG-Response(Recipient Authorizations)
12: Log ISSG Transaction
13: Ack
14: PDP-Request(Release Authorization)
15: PDP-Response(Authorization Response)
16: Log PDP Transaction
17: Ack
alt
[If one or more Message elements is Authorized for release]
opt
[If one or more Information Elements is not authorized for release]
18: PPS_AlertWarning(UnauthorizedElements)
19: Ack
loop
[for each authorized InformationElements]
20: ISSG-Request (Cryptographic Keys and Tokens)
21: ISSG-Response(Return Keys and Tokens)
22: Log ISSG Transaction
23: Ack
24: CTS-Request (InformationElement, key, token)
25: CTS-Response (Encrypted InformationElement)
26: Log CTS Transaction
27: Ack
28: Package message
29: Issue Message
30: Ack
31: TLS-LogMessage(TransactionData)
32: Ack
[else]
33: PPS-AlertWarning(UnauthorizedRelease)
34: Ack

Figure 50 - PPS Publish Authorization

The following table identifies and describes the interactions between IEF components as illustrated.

Table 45 - PPS Publish Authorization Messages

#	Messages			
1	Name	PPS-Publish (Payload, Attachments, metadata)		
	Description	The PPS packages (aggregates, transforms, marks, redacts, structures and formats) a releasable information element that conforms to an active Information Exchange Specification. The resulting Information Element (e.g., XML Document), its associated metadata and its identified attachments are packaged as a PPS-Publish Message, and issued to the Messaging-PEP for authorization and release to the users messaging infrastructure.		
	Sender	PPS	Receiver	Messaging-PEP
2	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) Acknowledge the receipt of the message to provide non-repudiation for the receipt of messages during a forensic audit. Acknowledgment of messages is optional, as the user may use the non-repudiation capabilities of an ISMB infrastructure (e.g., DDS) to assure delivery, quality of service and logging (real-time monitoring and forensic auditing).		
	Sender	Messaging-PEP	Receiver	PPS
3	Name	Log PPS Transaction		
	Description	The PPS records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	PPS	Receiver	TLS
4	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to PPS.		
	Sender	TLS	Receiver	PPS
5	Name	Gather InformationElement Metadata		
	Description	The PEP gathers the metadata and handling instructions for each of the informationElements (Payloads and Attachments) enclosed in the PPS publication.		
	Sender	Messaging-PEP	Receiver	Messaging-PEP
6	Name	ISSG-Request (Sender Authorizations)		
	Description	The PEP gathers the metadata about the sender from the PPS-Publish message metadata, and packages the data as an ISSG-Request (to get user attributes), and issues the request to the ISSG for processing.		
	Sender	Messaging-PEP	Receiver	ISSG
7	Name	ISSG-Response (Return Sender Authorizations)		

Table 45 - PPS Publish Authorization Messages

#	Messages			
	Description	The ISSG transforms the PEP request into the appropriate format for the specified user service (e.g., ICAM service or TrustMark Registry), and issues the request to that service. The response from the ICAM service is translated into the form to ISMB response requirements. The ISSG then packages and publishes an ISSG-Response message and issues it to the Messaging-PEP for processing.		
	Sender	ISSG	Receiver	Messaging-PEP
8	Name	Log ISSG Transaction		
	Description	The ISSG records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	ISSG	Receiver	TLS
9	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to ISSG.		
	Sender	TLS	Receiver	ISSG
10	Name	ISSG-Request (Recipients Authorizations)		
	Description	The PEP gathers the metadata about the recipient(s) specified in the PPS-Publish message metadata. The recipient(s) may be: 1. Individuals; 2. Organizations; 3. Community of Interest/practice; 4. Channel (e.g., DDS or AMQP) Topic; or 5. Channel Queue.		
	Sender	Messaging-PEP	Receiver	ISSG
11	Name	ISSG-Response (Recipient Authorizations)		
	Description	The ISSG transforms the PEP request into the appropriate format for the specified user service (e.g., ICAM service or TrustMark Registry), and issues the request to that service. The response from the ICAM service is translated into the form to ISMB response requirements. The ISSG then packages and publishes an ISSG-Response message and issues it to the Messaging-PEP for processing.		
	Sender	ISSG	Receiver	Messaging-PEP
12	Name	Log ISSG Transaction		
	Description	The ISSG records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	ISSG	Receiver	TLS
13	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to ISSG.		
	Sender	TLS	Receiver	ISSG

Table 45 - PPS Publish Authorization Messages

#	Messages			
14	Name	PDP-Request (Release Authorization)		
	Description	The PEP gathers the sensitivity metadata from each of the informationElements in the PPS-Publish message, as well as the sender and recipient authorizations. It packages this information as a PDP-request and issues the request to the PDP for adjudication and determination.		
	Sender	Messaging-PEP	Receiver	PDP
15	Name	PDP-Response (Authorization Response)		
	Description	The PDP adjudicates each of the requested authorizations and returns its decisions to the Messaging-PEP. The response is packaged as a PDP-AuthorizationResponse message and issued to the PEP for processing.		
	Sender	PDP	Receiver	Messaging-PEP
16	Name	Log PDP Transaction		
	Description	The PDP records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	PDP	Receiver	TLS
17	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to PDP.		
	Sender	TLS	Receiver	PDP
18	Name	PPS_AlertWarning (UnauthorizedElements)		
	Description	If one or more information elements in the message is not authorized for release, issue an alert to the PAP (system administrator).		
	Sender	Messaging-PEP	Receiver	PAP
19	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The PAP acknowledges the receipt of the AlertWarning to the PEP.		
	Sender	PAP	Receiver	Messaging-PEP
20	Name	ISSG-Request (Cryptographic Keys and Tokens)		
	Description	The Messaging-PEP packages an ISSG-Request for a cryptographic key and token for the information elements being released.		
	Sender	Messaging-PEP	Receiver	ISSG
21	Name	ISSG-Response (Return Keys and Tokens)		
	Description	The ISSG transforms the ISSG-Request into a form conforming to the user services interface requirement, packages the information and issues it in the appropriate format to the specified Key Management Services. This example assumes that the request initiates the generation of the key and storage of the key and token by the Key Escrow. Upon receipt of the Key(s) from the Key Escrow, the ISSG packages the key(s) and tokens and issues an ISSG_Response to the CTS.		
	Sender	ISSG	Receiver	Messaging-PEP

Table 45 - PPS Publish Authorization Messages

#	Messages			
22	Name	Log ISSG Transaction		
	Description	The ISSG records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	ISSG	Receiver	TLS
23	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to the PEP.		
	Sender	TLS	Receiver	ISSG
24	Name	CTS-Request (InformationElement, key, token)		
	Description	The Messaging-PEP packages the information element, key and token as a CTS-Request and issues the message to the CTS for processing (Encryption).		
	Sender	Messaging-PEP	Receiver	CTS
25	Name	CTS-Response (Encrypted InformationElement)		
	Description	The CTS packages the transformed InformationElement as CTS-Response and issues it to the Messaging-PEP.		
	Sender	CTS	Receiver	Messaging-PEP
26	Name	Log CTS Transaction		
	Description	The PDP records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	CTS	Receiver	TLS
27	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to CTS.		
	Sender	TLS	Receiver	CTS
28	Name	Package message		
	Description	The PEP gathers the authorized and encrypted information elements and packages them for release to the user's middleware using the user specified protocol.		
	Sender	Messaging-PEP	Receiver	Messaging-PEP
29	Name	Issue Message		
	Description	The Messaging-PEP issues the message to the specified middleware for dissemination to the recipients.		
	Sender	Messaging-PEP	Receiver	User-Middleware
30	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The Messaging Middleware interface acknowledges the receipt of the message to the PEP.		
	Sender	User-Middleware	Receiver	Messaging-PEP
31	Name	TLS-LogMessage (TransactionData)		

Table 45 - PPS Publish Authorization Messages

#	Messages			
	Description	The PEP records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	Messaging-PEP	Receiver	TLS
32	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to PEP.		
	Sender	TLS	Receiver	Messaging-PEP
33	Name	PPS-AlertWarning (UnauthorizedRelease)		
	Description	If the message is not authorized for release, the Messaging-PEP issues an Alert to the **PAP** (/Administrator) that the **PPS** is not authorized to release the i*nformation elements* to a specified Recipient or Communication Channel.		
	Sender	Messaging-PEP	Receiver	PAP
34	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to PEP.		
	Sender	PAP	Receiver	Messaging-PEP

PPS Request Authorization

The following figure illustrates a representational exchange of information between IEF Components when validating that a user is authorized to request the specified information from the PPS. The Messaging-PEP orchestrates the process for assuring that the published information is releasable to the recipient(s) over the specified communication channel. A user may make a number of requests:

- List of information types (i.e., Semantic Elements) supported by the PPS;
- List of active information elements (e.g., message, FilteredSemanticElements);
- List of active Information Exchange Specifications (/contracts);
- Inclusion into an existing Information Exchange Specification (/agreement /contract);
- The creation of a new Information Exchange Specification;
- A modification to an existing Information Exchange Specification;
- The availability and fidelity of the user's (e.g., network, devices, systems, services, and users) authorizations, privileges and attributes; and/or
- A one-time release of information from the PPS.

*Note: there are multiple paths through the authorizations process depending on:

- The number of information elements being published simultaneously;
- The complexity of the message structure (e.g., digest, packages, payloads and attachments);

- The number of recipients (e.g., Single topic or queue, or a list of individual recipients);
- The capabilities of each of the selected IEF components; and
- The complexity and fidelity of the user's own policies.

Many of these considerations will be addressed in the individual component specifications.

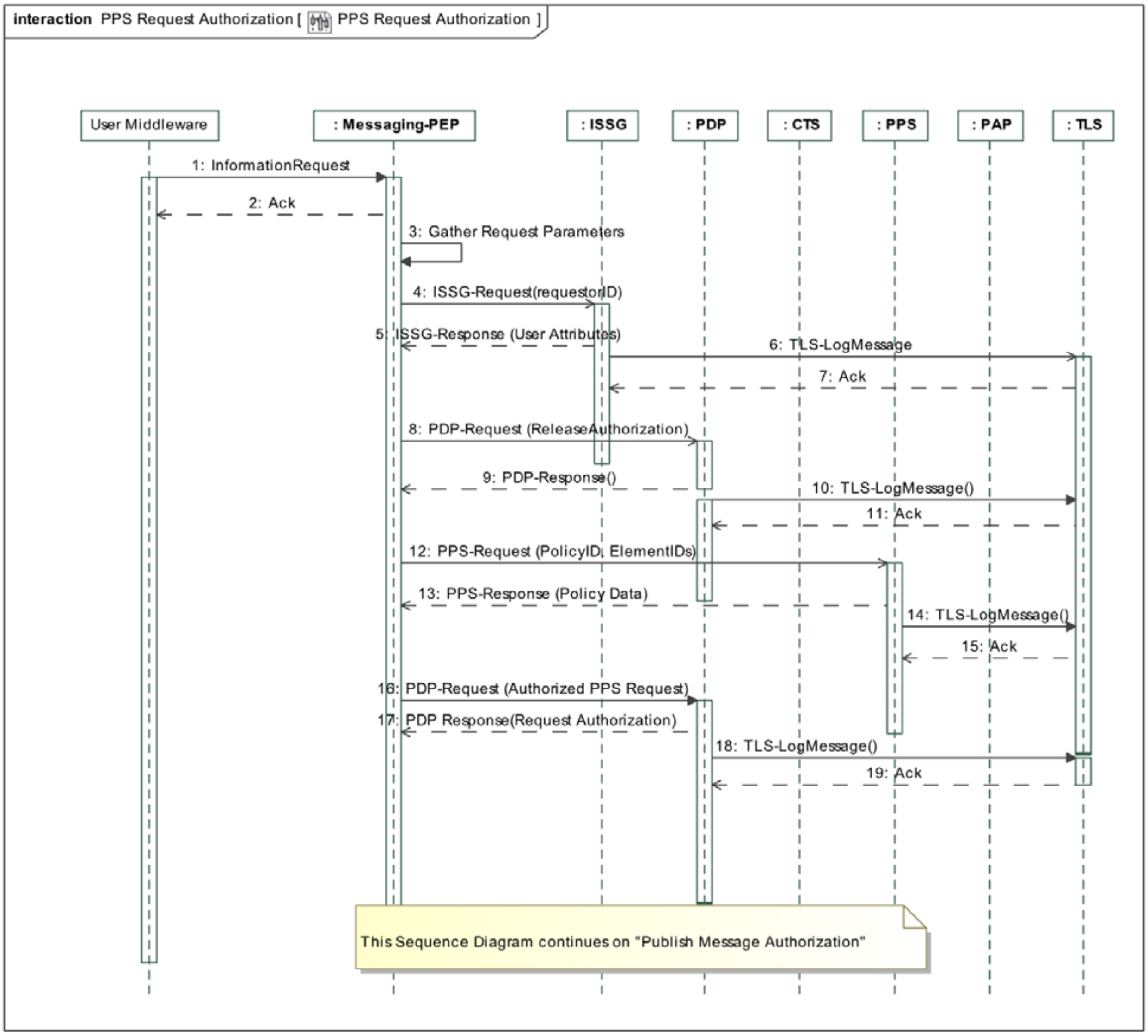

interaction PPS Request Authorization [PPS Request Authorization]
User Middleware
: Messaging-PEP
: ISSG
: PDP
: CTS
: PPS
: PAP
: TLS
1: InformationRequest
2: Ack
3: Gather Request Parameters
4: ISSG-Request(requestorID)
5: ISSG-Response (User Attributes)
6: TLS-LogMessage
7: Ack
8: PDP-Request (ReleaseAuthorization)
9: PDP-Response()
10: TLS-LogMessage()
11: Ack
12: PPS-Request (PolicyID, ElementIDs)
13: PPS-Response (Policy Data)
14: TLS-LogMessage()
15: Ack
16: PDP-Request (Authorized PPS Request)
17: PDP Response(Request Authorization)
18: TLS-LogMessage()
19: Ack
This Sequence Diagram continues on "Publish Message Authorization"

Figure 51 - PPS Request Authorization

The following table identifies and describes the interactions between IEF components as illustrated.

Table 46 - PPS Request Authorization Messages

#	Messages			
1	Name	InformationRequest		
	Description	The Messaging-PEP receives a message requesting information from a local user application. The PEP extracts the pertinent information out of the message to request authorization and package the PPS-Request Message.		
	Sender	User-Middleware	Receiver	Messaging-PEP
2	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The PEP acknowledges the receipt of the message to User Application.		
	Sender	Messaging-PEP	Receiver	User-Middleware
3	Name	Gather Request Parameters		
	Description	The PEP parses the request message, and extracts the request parameters needed to request release authorizations and package the PPS-Request.		
	Sender	Messaging-PEP	Receiver	Messaging-PEP
4	Name	ISSG-Request(requestorID)		
	Description	The PEP packages an ISSG-Request (to get user attributes and authorizations), and issues the request to the ISSG for processing.		
	Sender	Messaging-PEP	Receiver	ISSG
5	Name	ISSG-Response (User Attributes)		
	Description	The ISSG transforms the PEP request into the appropriate format to the specified user service (e.g., ICAM service or TrustMark Registry), and issues the request to that service. Based on the information from the user infrastructure, the ISSG translates into aa ISSG prepares an ISSG-Response, and issues it to the Messaging-PEP for processing.		
	Sender	ISSG	Receiver	Messaging-PEP
6	Name	TLS-LogMessage		
	Description	The ISSG records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	ISSG	Receiver	TLS
7	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to ISSG.		
	Sender	TLS	Receiver	ISSG
8	Name	PDP-Request (ReleaseAuthorization)		
	Description	The PEP gathers the sensitivity metadata for the requested MessageSpecifications or FilteredSemantics in the information request, as well as the sender and recipient authorizations. It packages this information as a PDP-request and issues the request to the PDP for adjudication and determination.		
	Sender	Messaging-PEP	Receiver	PDP

Table 46 - PPS Request Authorization Messages

#	Messages			
9	Name	PDP-Response ()		
	Description	The PDP returns is determination to the PEP to be enforced.		
	Sender	PDP	Receiver	Messaging-PEP
10	Name	TLS-LogMessage ()		
	Description	The PDP records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	PDP	Receiver	TLS
11	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to PDP.		
	Sender	TLS	Receiver	PDP
12	Name	PPS-Request (PolicyID, ElementIDs)		
	Description	The PEP issues a PPS-Request that includes the identifier for data policy being requested and the IUD of the elements to be included.		
	Sender	Messaging-PEP	Receiver	PPS
13	Name	PPS-Response (Policy Data)		
	Description	The information element defined by its policy ID (e.g., SemanticElement ID, FilteredSemanticElement ID, Message ID, or IES ID) holds the user`s (data owner) assessment of the sensitivity (e.g., security level, the specified [0..*] restrictions / caveats, and [0..*] handling instructions) and these elements are provided to the Messaging-PEP. The Messaging-PEP packages a PPS-Response.		
	Sender	PPS	Receiver	Messaging-PEP
14	Name	TLS-LogMessage ()		
	Description	The PPS records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	PPS	Receiver	TLS
15	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to PPS.		
	Sender	TLS	Receiver	PPS
16	Name	PDP-Request (Authorized PPS Request)		
	Description	The PEP gathers the sensitivity metadata from informationElements, and recipient authorizations, packages this information as a PDP-request, and issues the request to the PDP for adjudication and determination.		
	Sender	Messaging-PEP	Receiver	PDP
17	Name	PDP Response (Request Authorization)		
	Description	The PDP adjudicates each of the requested authorizations and returns its decisions to the Messaging-PEP. The response is packaged as a PDP-AuthorizationResponse message and issued to the PEP for processing.		

Table 46 - PPS Request Authorization Messages

#	Messages			
	Sender	PDP	Receiver	Messaging-PEP
18	Name	TLS-LogMessage ()		
	Description	The PDP records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	PDP	Receiver	TLS
19	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to PDP.		
	Sender	TLS	Receiver	PDP

Email Sequence Diagrams

The following clauses provide representative sequence diagrams for IEF components enforcing email protections.

Receive-Email

The following figure illustrates a representational exchange of information between IEF Components when validating that a user is authorized to receive an email with the provided content (body and attachments).

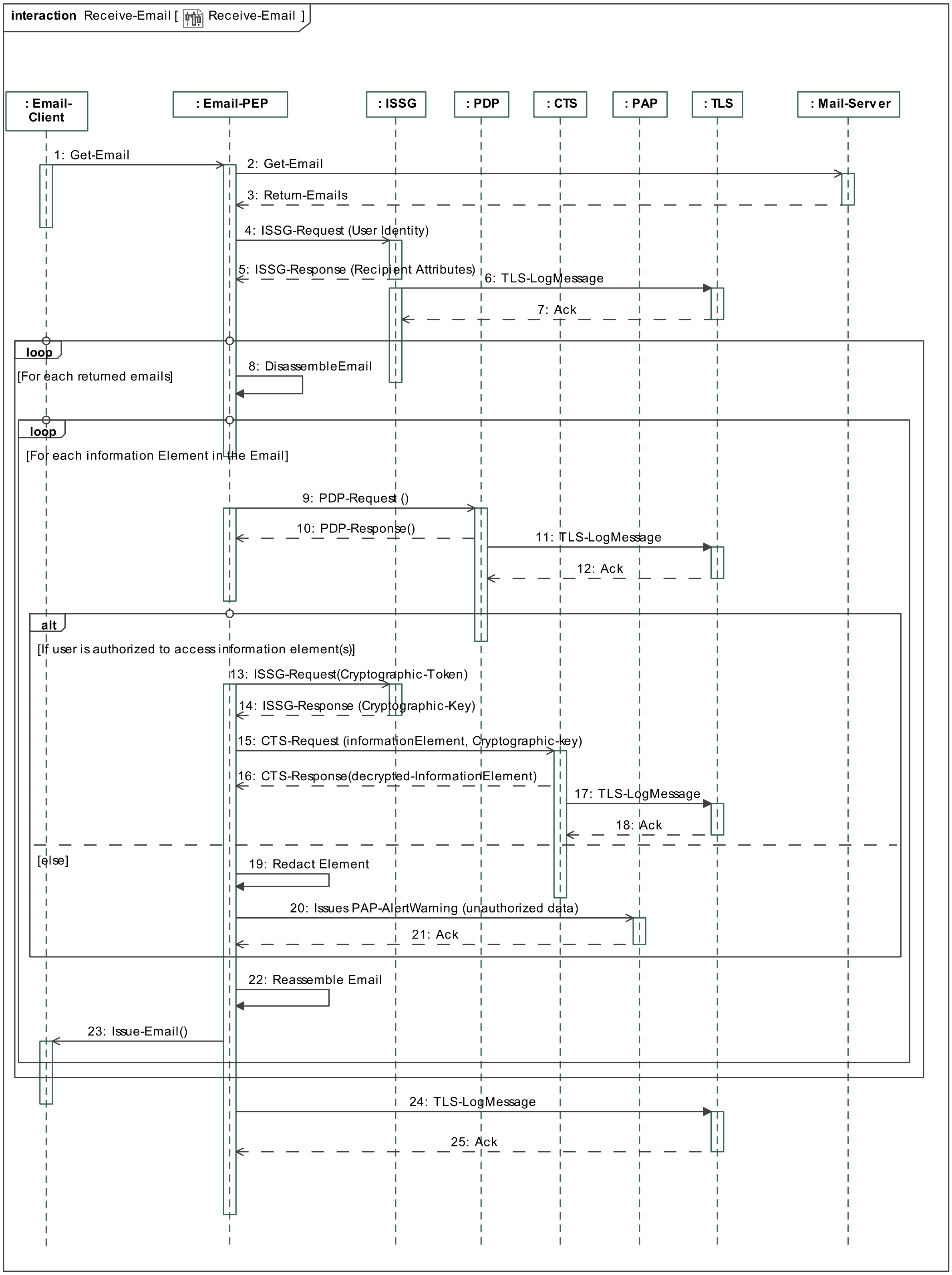

interaction Receive-Email [Receive-Email]
: Email-Client
: Email-PEP
: ISSG
: PDP
: CTS
: PAP
: TLS
: Mail-Server
1: Get-Email
2: Get-Email
3: Return-Emails
4: ISSG-Request (User Identity)
5: ISSG-Response (Recipient Attributes)
6: TLS-LogMessage
7: Ack
loop
[For each returned emails]
8: DisassembleEmail
loop
[For each information Element in the Email]
9: PDP-Request ()
10: PDP-Response()
11: TLS-LogMessage
12: Ack
alt
[If user is authorized to access information element(s)]
13: ISSG-Request(Cryptographic-Token)
14: ISSG-Response (Cryptographic-Key)
15: CTS-Request (informationElement, Cryptographic-key)
16: CTS-Response(decrypted-InformationElement)
17: TLS-LogMessage
18: Ack
[else]
19: Redact Element
20: Issues PAP-AlertWarning (unauthorized data)
21: Ack
22: Reassemble Email
23: Issue-Email()
24: TLS-LogMessage
25: Ack

Figure 52 - Receive-Email

The following table identifies and describes the interactions between IEF components as illustrated.

Table 47 - Receive-Email Messages

#	Messages			
1	Name	Get-Email		
	Description	The Email-PEP (Proxy) intercepts the email client to get email from the server.		
	Sender	Email-Client	Receiver	Email-PEP
2	Name	Get-Email		
	Description	The Email-PEP parses the request, extracts relevant data elements (e.g., user id), and then packages and issues the request to the Email Server to retrieve new messages for the specified user.		
	Sender	Email-PEP	Receiver	Mail-Server
3	Name	Return-Emails		
	Description	The Email-PEP intercepts the emails issued by the mail-server in response to the user request.		
	Sender	Mail-Server	Receiver	Email-PEP
4	Name	ISSG-Request (User Identity)		
	Description	The Email-PEP gathers the user identity information from the user's request, packages an ISSG-Request for User Authorizations, and issues it the ISSG.		
	Sender	Email-PEP	Receiver	ISSG
5	Name	ISSG-Response (Recipient Attributes)		
	Description	The ISSG retrieves the requested information from the user's ICAM, or privilege management system. The ISSG receives the response message from the ICAM services, parses the message, retrieves the users attributes, and then packages and issues an ISSG-Response to the Email-PEP.		
	Sender	ISSG	Receiver	Email-PEP
6	Name	TLS-LogMessage		
	Description	The ISSG records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	ISSG	Receiver	TLS
7	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to ISSG.		
	Sender	TLS	Receiver	ISSG
8	Name	DisassembleEmail		
	Description	The Email-PEP disassembles each email message to extract the information elements and supporting metadata.		
	Sender	Email-PEP	Receiver	Email-PEP
9	Name	PDP-Request ()		

Table 47 - Receive-Email Messages

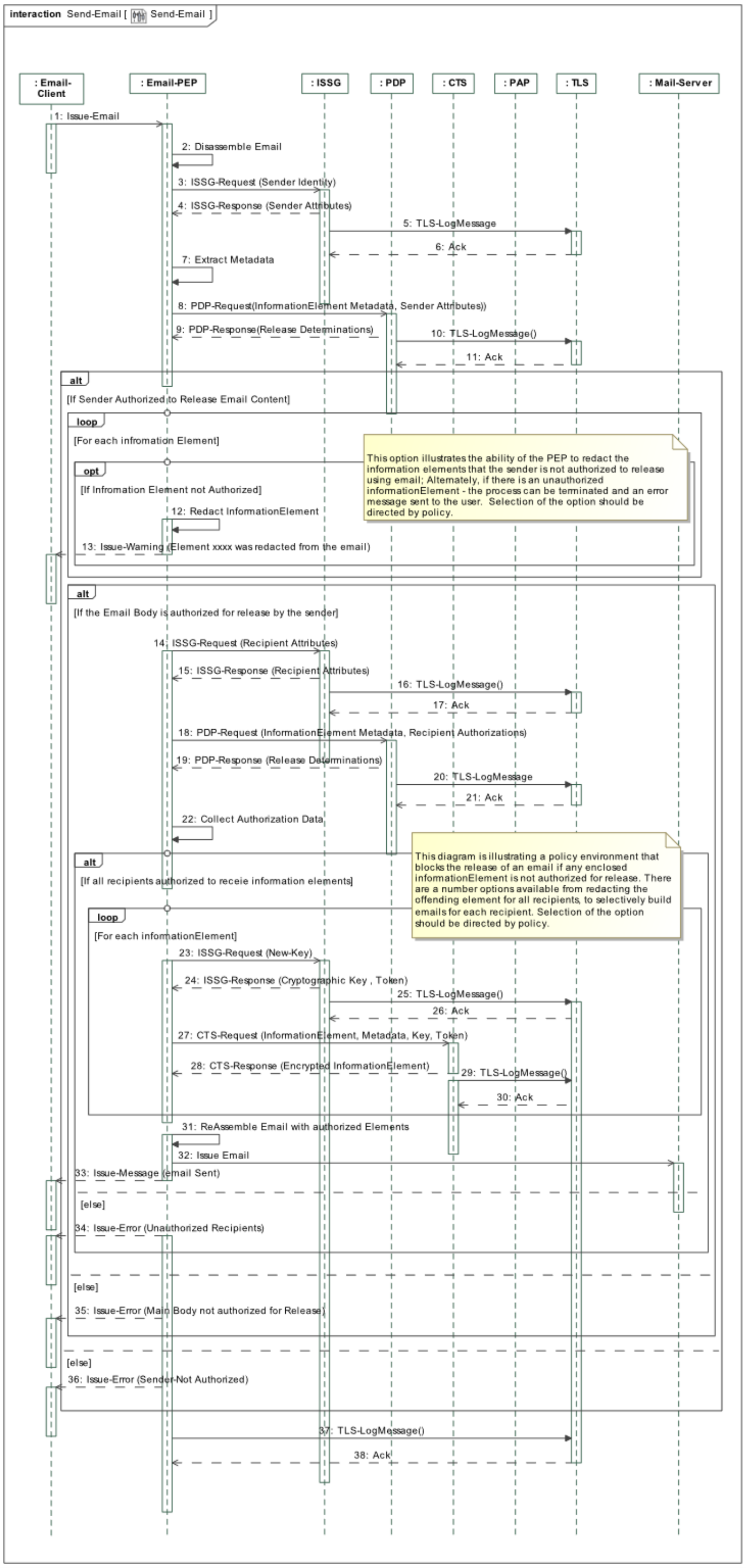
#	Messages			
	Description	The Email-PEP gathers the informationElement metadata and the user's (recipient's) attributes (authorizations), packages the data as a PDP-Request and issues it to the PDP for adjudication and determination.		
	Sender	Email-PEP	Receiver	PDP
10	Name	PDP-Response ()		
	Description	The PDP gathers the required policies from the policy data-store, adjudicates the user's rights to access the information elements, packages its determination as a PDP-Response and issues it to the Email-PEP for it to be enforced.		
	Sender	PDP	Receiver	Email-PEP
11	Name	TLS-LogMessage		
	Description	The PDP records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	PDP	Receiver	TLS
12	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to ISSG.		
	Sender	TLS	Receiver	PDP
13	Name	ISSG-Request (Cryptographic-Token)		
	Description	The Email-PEP gathers the cryptographic key token(s) from each informationElement's metadata, packages the tokens as an ISSG-Request to retrieve the cryptographic key(s) from the user's escrow service.		
	Sender	Email-PEP	Receiver	ISSG
14	Name	ISSG-Response (Cryptographic-Key)		
	Description	The ISSG parses the ISSG-Request, retrieves the informationElement key token(s), packages the token(s) as an escrow service request and issues the request. The ISSG receives the response from the escrow services, parses the message, retrieves the key-token pairs, then packages the data as an ISSG-response, and issues it to the Email-PEP.		
	Sender	ISSG	Receiver	Email-PEP
15	Name	CTS-Request (informationElement, Cryptographic-key)		
	Description	The Email-PEP packages the InformationElement and its cryptographic key as a CTS-Request and issues it to the CTS for processing.		
	Sender	Email-PEP	Receiver	CTS
16	Name	CTS-Response(decrypted-InformationElement)		
	Description	The CTS decrypts the informationElement, packages it as a CTS-Response message and issues the message to the Email-PEP.		
	Sender	CTS	Receiver	Email-PEP
17	Name	TLS-LogMessage		
	Description	The CTS records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		

Table 47 - Receive-Email Messages

#	Messages			
	Sender	CTS	Receiver	TLS
18	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to CTS.		
	Sender	TLS	Receiver	CTS
19	Name	Redact Element		
	Description	If the user is not authorized to access the informationElement, remove it from the email provided to the user. How the redaction is handled is based on user defined policy that may be encoded in the PEP or issued as an instruction from the PDP.		
	Sender	Email-PEP	Receiver	Email-PEP
20	Name	Issues PAP-AlertWarning (unauthorized data)		
	Description	(Optional) If an unauthorized information element is present in the email, the PEP may issue an alert/Warning to the PAP (administrator).		
	Sender	Email-PEP	Receiver	PAP
21	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The PAP acknowledges the receipt of the message to PEP.		
	Sender	PAP	Receiver	Email-PEP
22	Name	Reassemble Email		
	Description	After decrypting the authorized informationElements and removing unauthorized informationElements, the Email-PEP, repackages the email message.		
	Sender	Email-PEP	Receiver	Email-PEP
23	Name	Issue-Email ()		
	Description	The Email-PEP issues the reassembled email to the Email-client application.		
	Sender	Email-PEP	Receiver	Email-Client
24	Name	TLS-LogMessage		
	Description	The PEP records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	Email-PEP	Receiver	TLS
25	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to PEP.		
	Sender	TLS	Receiver	Email-PEP

Send-Email

The following figure illustrates a representational exchange of information between IEF Components when validating that a user is authorized send an email with the provided content (body and attachments).


interaction Send-Email [Send-Email]
: Email-Client
: Email-PEP
: ISSG
: PDP
: CTS
: PAP
: TLS
: Mail-Server
1: Issue-Email
2: Disassemble Email
3: ISSG-Request (Sender Identity)
4: ISSG-Response (Sender Attributes)
5: TLS-LogMessage
6: Ack
7: Extract Metadata
8: PDP-Request(InformationElement Metadata, Sender Attributes))
9: PDP-Response(Release Determinations)
10: TLS-LogMessage()
11: Ack
alt
[If Sender Authorized to Release Email Content]
loop
[For each infromation Element]
opt
[If Infromation Element not Authorized]
12: Redact InformationElement
13: Issue-Warning (Element xxxx was redacted from the email)
alt
[If the Email Body is authorized for release by the sender]
14: ISSG-Request (Recipient Attributes)
15: ISSG-Response (Recipient Attributes)
16: TLS-LogMessage()
17: Ack
18: PDP-Request (InformationElement Metadata, Recipient Authorizations)
19: PDP-Response (Release Determinations)
20: TLS-LogMessage
21: Ack
22: Collect Authorization Data
alt
[If all recipients authorized to receie information elements]
loop
[For each informationElement]
23: ISSG-Request (New-Key)
24: ISSG-Response (Cryptographic Key , Token)
25: TLS-LogMessage()
26: Ack
27: CTS-Request (InformationElement, Metadata, Key, Token)
28: CTS-Response (Encrypted InformationElement)
29: TLS-LogMessage()
30: Ack
31: ReAssemble Email with authorized Elements
32: Issue Email
33: Issue-Message (email Sent)
[else]
34: Issue-Error (Unauthorized Recipients)
[else]
35: Issue-Error (Main Body not authorized for Release)
[else]
36: Issue-Error (Sender Not Authorized)
37: TLS-LogMessage()
38: Ack

This option illustrates the ability of the PEP to redact the information elements that the sender is not authorized to release using email; Alternately, if there is an unauthorized informationElement - the process can be terminated and an error message sent to the user. Selection of the option should be directed by policy.

This diagram is illustrating a policy environment that blocks the release of an email if any enclosed informationElement is not authorized for release. There are a number options available from redacting the offending element for all recipients, to selectively build emails for each recipient. Selection of the option should be directed by policy.

Figure 53 - Send-Email

The following table identifies and describes the interactions between IEF components as illustrated.

Table 48 - Send-Email Messages

#	Messages			
1	Name	Issue-Email		
	Description	The User prepares and marks (adds required metadata) an email message and issues it to the Email server to be disseminated. The Email-PEP (Proxy) intercepts the email and assures that information is authorized for release and receipt by the specified users.		
	Sender	Email-Client	Receiver	Email-PEP
2	Name	Disassemble Email		
	Description	The Email-PEP disassembles (parses) the email message and extracts the information elements and associated metadata.		
	Sender	Email-PEP	Receiver	Email-PEP
3	Name	ISSG-Request (Sender Identity)		
	Description	The Email-PEP gathers the user (sender) identity information (Email Address) from the user's request, packages an ISSG-Request for User Authorizations, and issues it the ISSG.		
	Sender	Email-PEP	Receiver	ISSG
4	Name	ISSG-Response (Sender Attributes)		
	Description	The ISSG issues a request to the users ICAM services to retrieve the user's identity information and authorizations. The ISSG retrieves the users (Senders) identity information and attributes, packages them as an ISSG-Response and issues the response to the Email-PEP.		
	Sender	ISSG	Receiver	Email-PEP
5	Name	TLS-LogMessage		
	Description	The ISSG records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	ISSG	Receiver	TLS
6	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to ISSG.		
	Sender	TLS	Receiver	ISSG
7	Name	Extract Metadata		
	Description	The Email-PEP gathers the metadata from the informationElements (Email body or attachment). The metadata is gathered from: • The Email Header; and/or • The informationElement's SAC EnvelopeHeader.		
	Sender	Email-PEP	Receiver	Email-PEP

Table 48 - Send-Email Messages

#	Messages			
8	Name	PDP-Request (InformationElement Metadata, Sender Attributes))		
	Description	The Email-PEP gathers the metadata for each of the informationElements (email body and each attachment) and the user's (sender's) attributes (authorizations), packages the data as a PDP-Request message, and issues the message to the PDP for adjudication and determination.		
	Sender	Email-PEP	Receiver	PDP
9	Name	PDP-Response (Release Determinations)		
	Description	The PDP determines if the sender is authorized to release each information element based on current user policy. It then packages its determinations as a PDP-Response and issues it to the Email-PEP for it to be enforced.		
	Sender	PDP	Receiver	Email-PEP
10	Name	TLS-LogMessage ()		
	Description	The PDP records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	PDP	Receiver	TLS
11	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to PDP.		
	Sender	TLS	Receiver	PDP
12	Name	Redact InformationElement		
	Description	The Email-PEP removes the offending (not authorized for release) informationElement from the set of elements being processed for release.		
	Sender	Email-PEP	Receiver	Email-PEP
13	Name	Issue-Warning		
	Description	If informationElements are redacted because the Sender is not authorized to release them, PEP issues a warning to the user.		
	Sender	Email-PEP	Receiver	Email-Client
14	Name	ISSG-Request (Recipient Attributes)		
	Description	The PEP gathers the email-addresses for each of the recipients, packages them as a request to the user's ICAM services for the recipient’s identity and authorization attributes.		
	Sender	Email-PEP	Receiver	ISSG
15	Name	ISSG-Response (Recipient Attributes)		
	Description	The ISSG packages the email-address as a request to the user's ICAM services for the user's identity and authorization attributes. On receipt of the information from the ICAM services the ISSG packages them as an ISSG Response and issues it to the PEP.		
	Sender	ISSG	Receiver	Email-PEP
16	Name	TLS-LogMessage ()		
	Description	The ISSG records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		

Table 48 - Send-Email Messages

#	Messages			
	Sender	ISSG	Receiver	TLS
17	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to ISSG.		
	Sender	TLS	Receiver	ISSG
18	Name	PDP-Request (InformationElement Metadata, Recipient Authorizations)		
	Description	The Email-PEP gathers the metadata for each of the informationElements (email body and each attachment) and the user's (recipient's) attributes (authorizations), packages the data as a PDP-Request message, and issues the message to the PDP for adjudication and determination.		
	Sender	Email-PEP	Receiver	PDP
19	Name	PDP-Response (Release Determinations)		
	Description	The PDP determines if each recipient is authorized to receive each information element based on current user policy. It then packages its determinations as a PDP-Response and issues it to the Email-PEP for it to be enforced.		
	Sender	PDP	Receiver	Email-PEP
20	Name	TLS-LogMessage		
	Description	The PDP records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	PDP	Receiver	TLS
21	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to PDP.		
	Sender	TLS	Receiver	PDP
22	Name	Collect Authorization Data		
	Description	The Email-PEP gathers the PDP determinations from the PDP-Response.		
	Sender	Email-PEP	Receiver	Email-PEP
23	Name	ISSG-Request (New-Key)		
	Description	The Email-PEP packages and issues an ISSG-Request to request a new Cryptographic-Key and Token from the user's Key Management Services.		
	Sender	Email-PEP	Receiver	ISSG
24	Name	ISSG-Response (Cryptographic Key, Token)		
	Description	The ISSG retrieves the new cryptographic Key and Token, packages them as an ISSG-Response, and issues the response to the Email-PEP.		
	Sender	ISSG	Receiver	Email-PEP
25	Name	TLS-LogMessage ()		
	Description	The ISSG records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		

Table 48 - Send-Email Messages

#	Messages			
	Sender	ISSG	Receiver	TLS
26	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to ISSG.		
	Sender	TLS	Receiver	ISSG
27	Name	CTS-Request (InformationElement, Metadata, Key, Token)		
	Description	The Email-PEP packages a CTS-Request to encrypt the informationElement. The PEP gathers and packages: • The informationElement; • The informationElement's metadata; • The Cryptographic Key and Token.		
	Sender	Email-PEP	Receiver	CTS
28	Name	CTS-Response (Encrypted InformationElement)		
	Description	The CTS, encrypts the information element and: • if the email body, return it to the Email PEP; • if Attachment, package the Attachment as a SAC and return it to the Email-PEP. The CTS packages the encrypted informationElement or SAC as a CTS-Response and issues it to the Email-PEP for processing.		
	Sender	CTS	Receiver	Email-PEP
29	Name	TLS-LogMessage ()		
	Description	The CTS records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	CTS	Receiver	TLS
30	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to CTS.		
	Sender	TLS	Receiver	CTS
31	Name	ReAssemble Email with authorized Elements		
	Description	After encrypting* each of the authorized informationElement(s) the Email-PEP, repackages the email message with all the information elements encrypted.		
	Sender	Email-PEP	Receiver	Email-PEP
32	Name	Issue Email		
	Description	The PEP then issues the repackaged email to the Email Server for distribution.		
	Sender	Email-PEP	Receiver	Mail-Server
33	Name	Issue-Message (email Sent)		
	Description	Issue a message to the user that the email has been sent.		

Table 48 - Send-Email Messages

#	Messages			
	Sender	Email-PEP	Receiver	Email-Client
34	Name	Issue-Error (Unauthorized Recipients)		
	Description	If a recipient sender is not authorized to release the content of the Email message, halt processing and issue an error to the User (Sender).		
	Sender	Email-PEP	Receiver	Email-Client
35	Name	Issue-Error (Main Body not authorized for Release)		
	Description	Issue an error message to the user that the email body was not authorized for release to one or more recipients.		
	Sender	Email-PEP	Receiver	Email-Client
36	Name	Issue-Error (Sender-Not Authorized)		
	Description	If the sender is not authorized to release the content of the Email message, halt processing and issue an error to the User (Sender).		
	Sender	Email-PEP	Receiver	Email-Client
37	Name	TLS-LogMessage ()		
	Description	The PEP records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	Email-PEP	Receiver	TLS
38	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to PEP.		
	Sender	TLS	Receiver	Email-PEP

File Share Sequence Diagrams

The following clauses provide representative sequence diagrams for IEF components enforcing file protections. The File-PEP authorizes that the recipient:

1. Is authorized to access the files share (e.g., derive and folder) and perform the requested activity; and
2. Is authorized to access the file in the share and perform the requested access.

When requesting information about a share, or the contents in a selected share, the recipient is only provided information about the files and folders they are authorized to access.

File Authorization

The following figure illustrates a representational exchange of information between IEF Components when validating that a user is authorized to access a requested file to perform the specified action. This includes requests to Create, Copy, Cut, Delete, Move, Open, Paste, and/or Save) save a file contained within the protected file share.

*Note: there are multiple paths through the authorizations process depending on:

- The number of files being requested simultaneously;
- The source and target for the requested informationElements (i.e., file);
- The capabilities of each of the selected IEF components;
- The availability and fidelity of the user's (e.g., network, devices, systems, services, and users) authorizations, privileges and attributes; and
- The complexity and fidelity of the user's own policies.

Many of these considerations will be addressed in the individual component specifications. This sequence outlines the process for accessing a single file located in the IEF protected file share.

For this example, we assume the request is acting on a file located in the IEF protected file share, meaning:

- The file is or will be encrypted using a symmetric key;
- The file is or will be appropriately marked; and
- The file will be maintained in a SAC.

For this example, we will also assume that the request is to access an existing file.

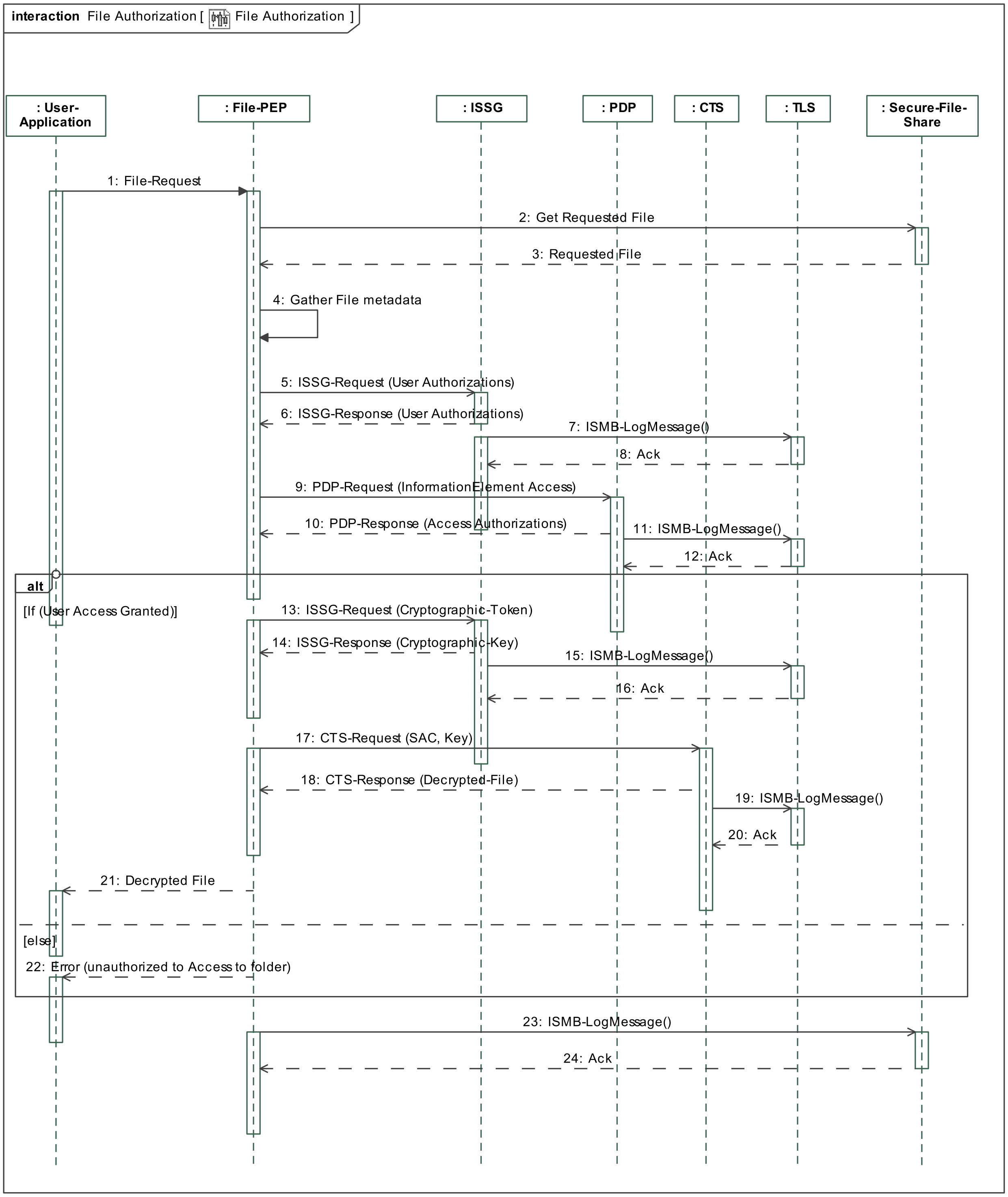

interaction File Authorization [File Authorization]
: User-Application
: File-PEP
: ISSG
: PDP
: CTS
: TLS
: Secure-File-Share
1: File-Request
2: Get Requested File
3: Requested File
4: Gather File metadata
5: ISSG-Request (User Authorizations)
6: ISSG-Response (User Authorizations)
7: ISMB-LogMessage()
8: Ack
9: PDP-Request (InformationElement Access)
10: PDP-Response (Access Authorizations)
11: ISMB-LogMessage()
12: Ack
alt
[If (User Access Granted)]
13: ISSG-Request (Cryptographic-Token)
14: ISSG-Response (Cryptographic-Key)
15: ISMB-LogMessage()
16: Ack
17: CTS-Request (SAC, Key)
18: CTS-Response (Decrypted-File)
19: ISMB-LogMessage()
20: Ack
21: Decrypted File
[else]
22: Error (unauthorized to Access to folder)
23: ISMB-LogMessage()
24: Ack

Figure 54 - File Authorization

The following table identifies and describes the interactions between IEF components as illustrated.

Table 49 - File Authorization Messages

#	Messages			
1	Name	File-Request		
	Description	The user application uses its standard interface to request an operation on a file, using a specified set of resources (source and/or target device/directory) within the IEF protected information store (e.g., file share).		
	Sender	User-Application	Receiver	File-PEP
2	Name	Get Requested File		
	Description	The PEP gets the file (SAC) from the IEF Protected Information Store (/File-share). The protected file-share is a designated location on the users' own infrastructure, where each file is encrypted using a symmetric key and enclosed in a Secure Asset Container (SAC).		
	Sender	File-PEP	Receiver	Secure-File-Share
3	Name	Requested File		
	Description	The file system provides the file's SAC for the requested file.		
	Sender	Secure-File-Share	Receiver	File-PEP
4	Name	Gather File metadata		
	Description	The PEP extracts the metadata from the Secure Asset Container.		
	Sender	File-PEP	Receiver	File-PEP
5	Name	ISSG-Request (User Authorizations)		
	Description	The PEP gathers the user identity information provided by the application and packages it as an ISSG-Request to gather the users' authorizations to access (write and delete) and use (read and modify) informationElements in the source location (device:/directory) and if needed, the target location.		
	Sender	File-PEP	Receiver	ISSG
6	Name	ISSG-Response (User Authorizations)		
	Description	The ISSG gathers the user's authorizations from the specified ICAM or Privilege Management System, and returns it to the PEP as an ISSG-Response message. The ISSG gathers authorizations to access the source location, target location (as needed) and file.		
	Sender	ISSG	Receiver	File-PEP
7	Name	ISMB-LogMessage ()		
	Description	The ISSG records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	ISSG	Receiver	TLS
8	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to ISSG.		

Table 49 - File Authorization Messages

#	Messages			
	Sender	TLS	Receiver	ISSG
9	Name	PDP-Request (InformationElement Access)		
	Description	The PEP packages the InformationElement metadata, requested operation and the users authorizations as a PDP-Request, and issues the message to the PDP for adjudication.		
	Sender	File-PEP	Receiver	PDP
10	Name	PDP-Response (Access Authorizations)		
	Description	The PDP adjudicates the user's authorization to access the file using the current user policies, then packages its decision and instructions as a PDP-Response message and issues it to the PEP for enforcement.		
	Sender	PDP	Receiver	File-PEP
11	Name	ISMB-LogMessage ()		
	Description	The PDP records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	PDP	Receiver	TLS
12	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to PDP.		
	Sender	TLS	Receiver	PDP
13	Name	ISSG-Request (Cryptographic-Token)		
	Description	The PEP packages the file's cryptographic-token as an ISSG-Request and issues it to the ISSG.		
	Sender	File-PEP	Receiver	ISSG
14	Name	ISSG-Response (Cryptographic-Key)		
	Description	The ISSG packages and issues a message to the user's key escrow services containing the key token and other required information. Upon receipt of the cryptographic key, the ISSG packages the key and token as an ISSG-Response and issues it to the PEP.		
	Sender	ISSG	Receiver	File-PEP
15	Name	ISMB-LogMessage ()		
	Description	The ISSG records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	ISSG	Receiver	TLS
16	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to CTS.		
	Sender	TLS	Receiver	ISSG
17	Name	CTS-Request (SAC, Key)		
	Description	The PEP packages the SAC and the retrieved Cryptographic-Key as a CTS-Request and issues it to the CTS for processing.		

Table 49 - File Authorization Messages

#	Messages			
	Sender	File-PEP	Receiver	CTS
18	Name	CTS-Response (Decrypted-File)		
	Description	The CTS extracts the file from the SAC and decrypts the file using the provided cryptographic-key. It then packages the decrypted file as a CTS-response and issues the message to the PEP.		
	Sender	CTS	Receiver	File-PEP
19	Name	ISMB-LogMessage ()		
	Description	The CTS records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	CTS	Receiver	TLS
20	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to CTS.		
	Sender	TLS	Receiver	CTS
21	Name	Decrypted File		
	Description	The PEP returns the decrypted file to the User Application.		
	Sender	File-PEP	Receiver	User-Application
22	Name	Error (unauthorized to Access to folder)		
	Description	If the user is denied access, an error message is issued to the user application.		
	Sender	File-PEP	Receiver	User-Application
23	Name	ISMB-LogMessage ()		
	Description	The PEP records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	File-PEP	Receiver	Secure-File-Share
24	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to PEP.		
	Sender	Secure-File-Share	Receiver	File-PEP

Folder Authorization

The following figure illustrates a representational exchange of information between IEF Components when validating that a user is authorized to access a requested folder in the protected file share.

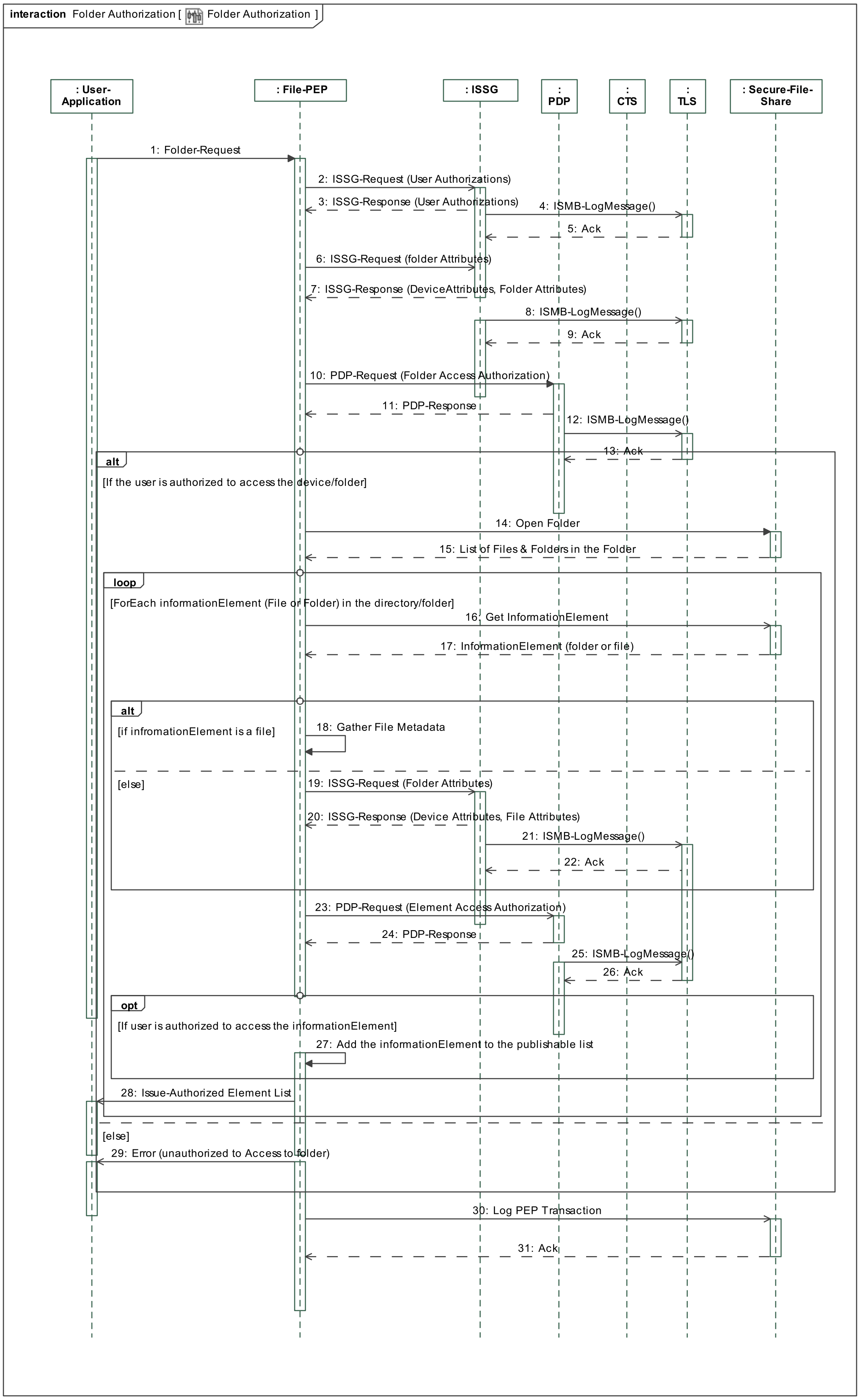

interaction Folder Authorization [Folder Authorization]
: User-Application
: File-PEP
: ISSG
: PDP
: CTS
: TLS
: Secure-File-Share
1: Folder-Request
2: ISSG-Request (User Authorizations)
3: ISSG-Response (User Authorizations)
4: ISMB-LogMessage()
5: Ack
6: ISSG-Request (folder Attributes)
7: ISSG-Response (DeviceAttributes, Folder Attributes)
8: ISMB-LogMessage()
9: Ack
10: PDP-Request (Folder Access Authorization)
11: PDP-Response
12: ISMB-LogMessage()
13: Ack
alt
[If the user is authorized to access the device/folder]
14: Open Folder
15: List of Files & Folders in the Folder
loop
[ForEach informationElement (File or Folder) in the directory/folder]
16: Get InformationElement
17: InformationElement (folder or file)
alt
[if infromationElement is a file]
18: Gather File Metadata
[else]
19: ISSG-Request (Folder Attributes)
20: ISSG-Response (Device Attributes, File Attributes)
21: ISMB-LogMessage()
22: Ack
23: PDP-Request (Element Access Authorization)
24: PDP-Response
25: ISMB-LogMessage()
26: Ack
opt
[If user is authorized to access the informationElement]
27: Add the informationElement to the publishable list
28: Issue-Authorized Element List
[else]
29: Error (unauthorized to Access to folder)
30: Log PEP Transaction
31: Ack

Figure 55 - Folder Authorization

The following table identifies and describes the interactions between IEF components as illustrated.

Table 50 - Folder Authorization Messages

#	Messages			
1	Name	Folder-Request		
	Description	The user application uses its standard interface to request an operation on a folder: • Change Folder; • Open Folder; or • List folder contents.		
	Sender	User-Application	Receiver	File-PEP
2	Name	ISSG-Request (User Authorizations)		
	Description	The File-PEP gathers identity information provided by the user application request, packages it as an ISSG-Request in order to gather the users' authorizations to access (write and delete) and use (read and modify) the requested informationElements, as well as the user's authorizations to access location (device:/directory) of the information and if needed the target location. It then issues the message to the ISSG to request the information from the user's own security services.		
	Sender	File-PEP	Receiver	ISSG
3	Name	ISSG-Response (User Authorizations)		
	Description	The ISSG gathers the users' authorizations from the specified ICAM or Privilege Management Service, and returns it to the PEP as an ISSG-Response message.		
	Sender	ISSG	Receiver	File-PEP
4	Name	ISMB-LogMessage ()		
	Description	The ISSG records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	ISSG	Receiver	TLS
5	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to ISSG.		
	Sender	TLS	Receiver	ISSG
6	Name	ISSG-Request (folder Attributes)		
	Description	The File-PEP gathers the device and directory information and packages it as ISSG-Request to gather the device and folder authorization requirements.		
	Sender	File-PEP	Receiver	ISSG
7	Name	ISSG-Response (DeviceAttributes, Folder Attributes)		

Table 50 - Folder Authorization Messages

#	Messages			
	Description	The ISSG gathers the device and folder authorization requirements from the specified ICAM or Privilege Management System, and returns it to the File-PEP as an ISSG-Response Message.		
	Sender	ISSG	Receiver	File-PEP
8	Name	ISMB-LogMessage ()		
	Description	The ISSG records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	ISSG	Receiver	TLS
9	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to ISSG.		
	Sender	TLS	Receiver	ISSG
10	Name	PDP-Request (Folder Access Authorization)		
	Description	The File-PEP gathers the folder authorization requirements and the user's authorizations, packages the information as a PDP-Request, and issues the message for adjudication and determination.		
	Sender	File-PEP	Receiver	PDP
11	Name	PDP-Response		
	Description	The PDP adjudicates the request against current user policy, packages its determinations and instructions as a PDP-Response message and issues it to the file-PEP for enforcement.		
	Sender	PDP	Receiver	File-PEP
12	Name	ISMB-LogMessage ()		
	Description	The PDP records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	PDP	Receiver	TLS
13	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to PDP.		
	Sender	TLS	Receiver	PDP
14	Name	Open Folder		
	Description	The PEP opens the folder and requests the list of files and folders in the folder.		
	Sender	File-PEP	Receiver	Secure-File-Share
15	Name	List of Files & Folders in the Folder		
	Description	The files system returns the contents of the folder for the processing and access authorization.		
	Sender	Secure-File-Share	Receiver	File-PEP
16	Name	Get InformationElement		
	Description	The PEP requests the informationElement (File or folder) from the protected file store.		

Table 50 - Folder Authorization Messages

#	Messages			
	Sender	File-PEP	Receiver	Secure-File-Share
17	Name	InformationElement (folder or file)		
	Description	The file share returns either the File SAC or Folder metadata to the PEP.		
	Sender	Secure-File-Share	Receiver	File-PEP
18	Name	Gather File Metadata		
	Description	If the element being evaluated is a file, gathers the metadata for the file from the SAC EnvelopeHeader.		
	Sender	File-PEP	Receiver	File-PEP
19	Name	ISSG-Request (Folder Attributes)		
	Description	The File-PEP gathers the device and directory information and packages it as an ISSG-Request to gather the device and folder authorization requirements.		
	Sender	File-PEP	Receiver	ISSG
20	Name	ISSG-Response (Device Attributes, File Attributes)		
	Description	The ISSG gathers the device and folder authorization requirements from specified ICAM or Privilege Management System, and returns it to the File-PEP as an ISSG-Response Message.		
	Sender	ISSG	Receiver	File-PEP
21	Name	ISMB-LogMessage ()		
	Description	The ISSG records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	ISSG	Receiver	TLS
22	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to ISSG.		
	Sender	TLS	Receiver	ISSG
23	Name	PDP-Request (Element Access Authorization)		
	Description	The PEP gathers the folder-element authorization requirements and the user's authorizations, packages the information as a PDP-Request, and issues the message for adjudication and access determination.		
	Sender	File-PEP	Receiver	PDP
24	Name	PDP-Response		
	Description	The PDP packages its authorization decisions and instructions as a PDP-Response message and issues it to the Files-PEP for processing.		
	Sender	PDP	Receiver	File-PEP
25	Name	ISMB-LogMessage ()		
	Description	The PDP records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	PDP	Receiver	TLS

Table 50 - Folder Authorization Messages

#	Messages			
26	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to the PDP.		
	Sender	TLS	Receiver	PDP
27	Name	Add the informationElement to the publishable list		
	Description	The PEP only publishes the name and attributes of the informationElements (files and folders) the user is authorized to access. As the PEP processes the informationElements it maintains a list publishable (accessible), redacting any informationElement the user is not authorized to access. In this way users are not aware of any informationElements they cannot access and information with differing sensitivities and restrictions can be stored in a common file system.		
	Sender	File-PEP	Receiver	File-PEP
28	Name	Issue-Authorized Element List		
	Description	The PEP packages the list of informationElement(s) the user is authorized to access and issues it to the user's application.		
	Sender	File-PEP	Receiver	User-Application
29	Name	Error (unauthorized to Access to folder)		
	Description	If the user is not authorized to access the requested folder, the PEP issues an error message to the user.		
	Sender	File-PEP	Receiver	User-Application
30	Name	Log PEP Transaction		
	Description	The PEP records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	File-PEP	Receiver	Secure-File-Share
31	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to PEP.		
	Sender	Secure-File-Share	Receiver	File-PEP

Instant Messaging Sequence Diagrams

The following clauses provide representative sequence diagrams for IEF components enforcing Instant Messaging protections.

IM Receive

The following figure illustrates a representational exchange of information between IEF Components when validating that a user is authorized to receive a massage from a chat room or individual sender.

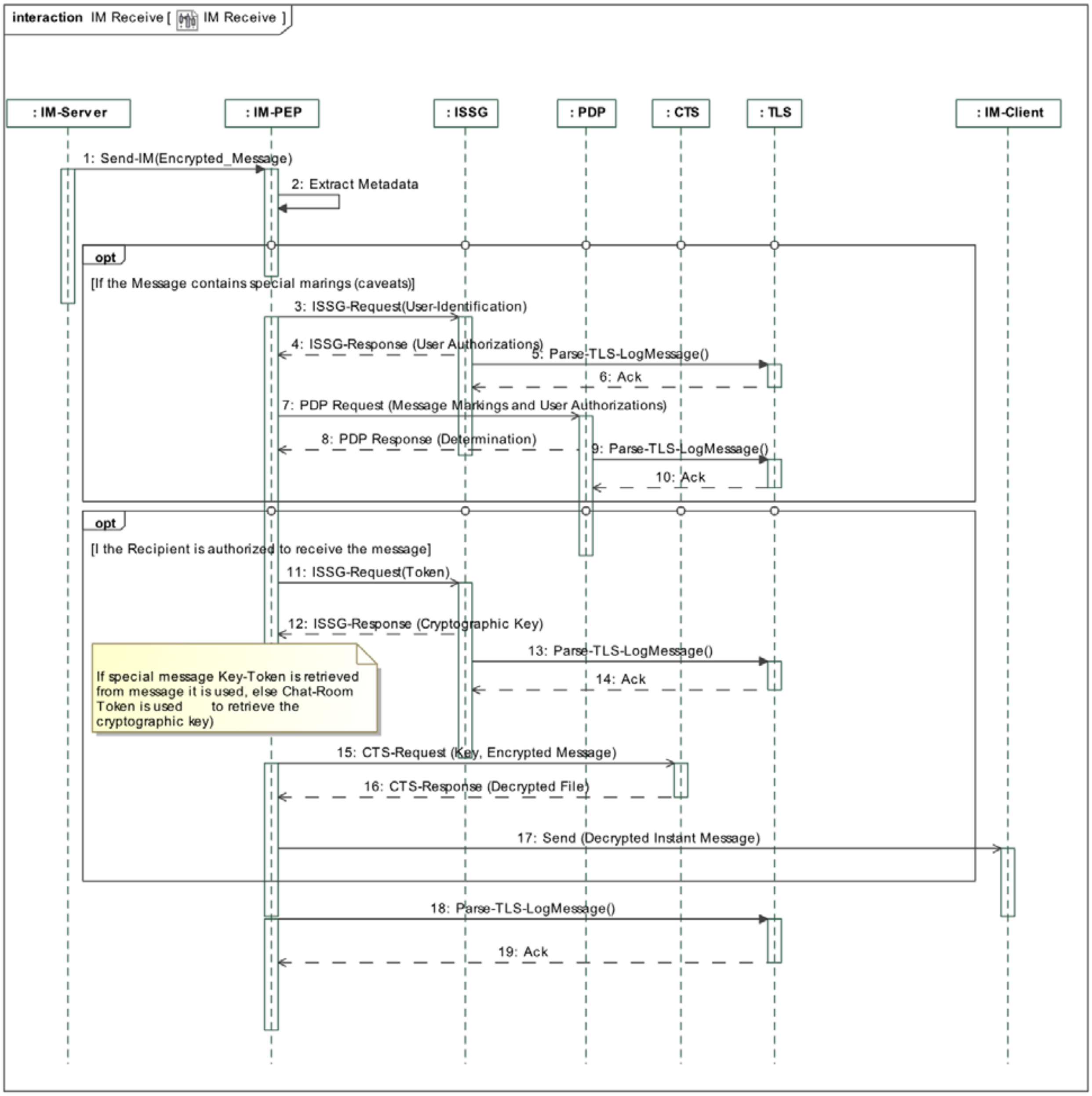

interaction IM Receive [IM Receive]
: IM-Server
: IM-PEP
: ISSG
: PDP
: CTS
: TLS
: IM-Client
1: Send-IM(Encrypted_Message)
2: Extract Metadata
opt
[If the Message contains special marings (caveats)]
3: ISSG-Request(User-Identification)
4: ISSG-Response (User Authorizations)
5: Parse-TLS-LogMessage()
6: Ack
7: PDP Request (Message Markings and User Authorizations)
8: PDP Response (Determination)
9: Parse-TLS-LogMessage()
10: Ack
opt
[I the Recipient is authorized to receive the message]
11: ISSG-Request(Token)
12: ISSG-Response (Cryptographic Key)
13: Parse-TLS-LogMessage()
14: Ack
If special message Key-Token is retrieved from message it is used, else Chat-Room Token is used to retrieve the cryptographic key)
15: CTS-Request (Key, Encrypted Message)
16: CTS-Response (Decrypted File)
17: Send (Decrypted Instant Message)
18: Parse-TLS-LogMessage()
19: Ack

Figure 56 - IM Receive

The following table identifies and describes the interactions between IEF components as illustrated.

Table 51 - IM Receive Messages

#	Messages			
1	Name			
	Description	The IM-Server sends the instant message to the users IM-Client, which is intercepted by the IM-PEP (Proxy).		
	Sender	IM-Server	Receiver	IM-PEP
2	Name	Extract Metadata		
	Description	The PEP extracts the metadata from the message header.		
	Sender	IM-PEP	Receiver	IM-PEP
3	Name	ISSG-Request (User-Identification)		
	Description	If the message metadata contains a special message indicator (special caveat), the PEP packages an ISSG-Request to gather the recipient`s authorizations. The PEP authorizes the User to access the chat room when they log into the room.		
	Sender	IM-PEP	Receiver	ISSG
4	Name	ISSG-Response (User Authorizations)		
	Description	The ISSG gathers the user identity information, packages and issues a message to the User`s ICAM services. Upon receipt of the user's authorizations from the ICAM services, the ISSG packages them as an ISSG-Response and issues them to the PEP.		
	Sender	ISSG	Receiver	IM-PEP
5	Name	TLS-LogMessage ()		
	Description	The ISSG records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	ISSG	Receiver	TLS
6	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to the ISSG.		
	Sender	TLS	Receiver	ISSG
7	Name	PDP Request (Message Markings and User Authorizations)		
	Description	The PEP gathers the sensitivity tags, as well as the recipient`s authorizations. It packages this information as a PDP-request and issues the request to the PDP for adjudication and determination.		
	Sender	IM-PEP	Receiver	PDP
8	Name	PDP Response (Determination)		

Table 51 - IM Receive Messages

#	Messages			
	Description	The PDP adjudicates the request using current user policies, packages the response as a PDP-AuthorizationResponse message and issues to the PEP to be enforced.		
	Sender	PDP	Receiver	IM-PEP
9	Name	TLS-LogMessage ()		
	Description	The PDP records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	PDP	Receiver	TLS
10	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to the PDP.		
	Sender	TLS	Receiver	PDP
11	Name	ISSG-Request (Token)		
	Description	The PEP packages an ISSG-Request (with chat-room or special message token) and sends the request to the ISSG to be actioned. If the message has a special caveat, the token is taken from the message metadata, else the chat-room token is used. Each chat-room uses a single Key.		
	Sender	IM-PEP	Receiver	ISSG
12	Name	ISSG-Response (Cryptographic Key)		
	Description	The ISSG transforms the key request into a message to the user's key management (escrow) service and issues the message to the User specified service. Upon receipt of the Key, the ISSG packages an ISSG-Response and issues it to the PEP.		
	Sender	ISSG	Receiver	IM-PEP
13	Name	TLS-LogMessage ()		
	Description	The ISSG records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	ISSG	Receiver	TLS
14	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to the ISSG.		
	Sender	TLS	Receiver	ISSG
15	Name	CTS-Request (Key, Encrypted Message)		
	Description	The PEP packages the cryptographic key and the message as a CTS-Request to have the CTS decrypt the message.		
	Sender	IM-PEP	Receiver	CTS
16	Name	CTS-Response (Decrypted File)		
	Description	The CTS decrypts the message and returns it to the PEP.		
	Sender	CTS	Receiver	IM-PEP

Table 51 - IM Receive Messages

#	Messages			
17	Name	Send (Decrypted Instant Message)		
	Description	The PEP repackages the Instant Message and issues the message to the recipients IM-Client application.		
	Sender	IM-PEP	Receiver	IM-Client
18	Name	TLS-LogMessage ()		
	Description	The PEP records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	IM-PEP	Receiver	TLS
19	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to the PEP.		
	Sender	TLS	Receiver	IM-PEP

IM Send

The following figure illustrates a representational exchange of information between IEF Components when validating that a user is authorized to send a massage to a chat room or individual recipients.

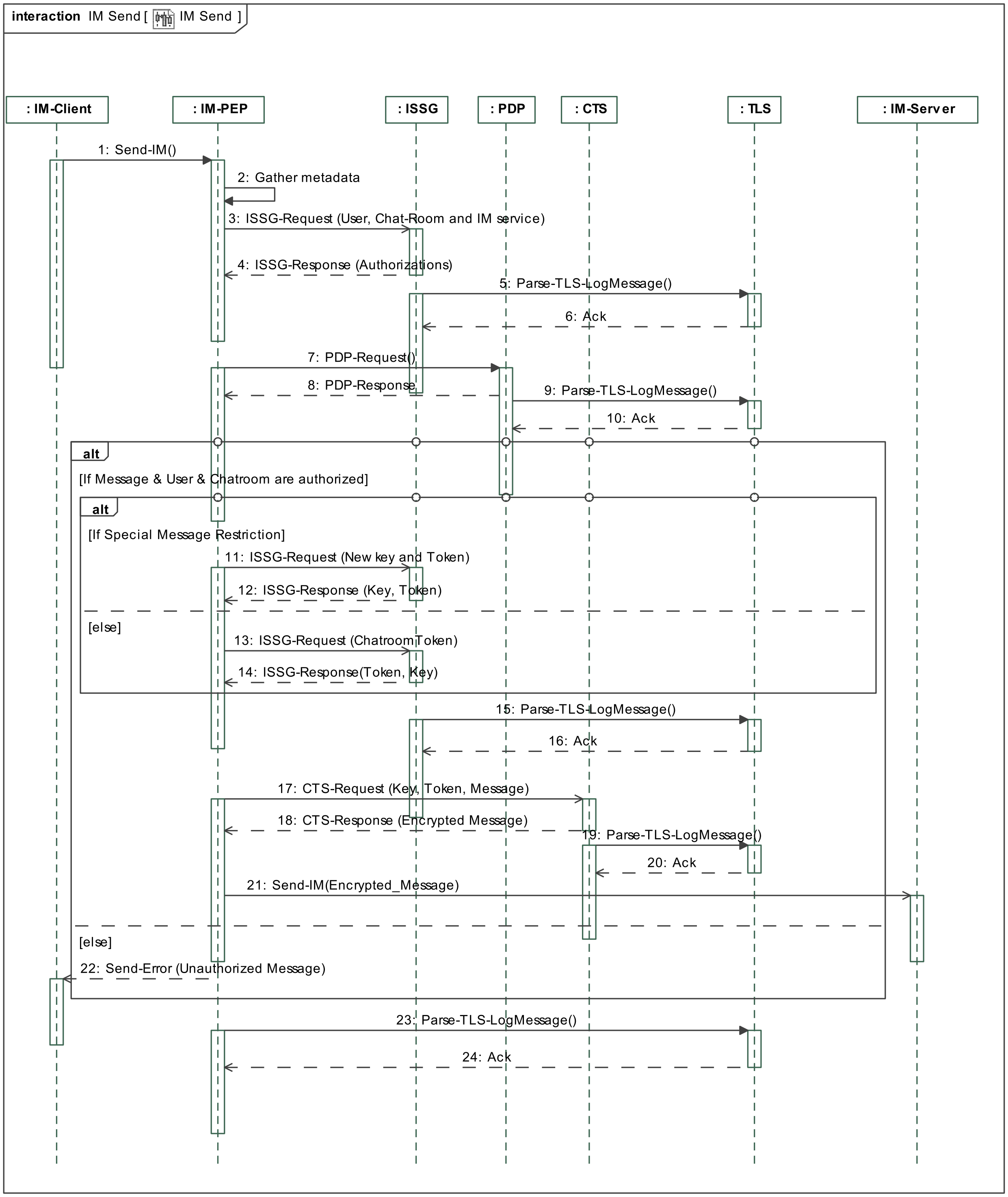

interaction IM Send [IM Send]
: IM-Client
: IM-PEP
: ISSG
: PDP
: CTS
: TLS
: IM-Server
1: Send-IM()
2: Gather metadata
3: ISSG-Request (User, Chat-Room and IM service)
4: ISSG-Response (Authorizations)
5: Parse-TLS-LogMessage()
6: Ack
7: PDP-Request()
8: PDP-Response
9: Parse-TLS-LogMessage()
10: Ack
alt
[If Message & User & Chatroom are authorized]
alt
[If Special Message Restriction]
11: ISSG-Request (New key and Token)
12: ISSG-Response (Key, Token)
[else]
13: ISSG-Request (Chatroom Token)
14: ISSG-Response(Token, Key)
15: Parse-TLS-LogMessage()
16: Ack
17: CTS-Request (Key, Token, Message)
18: CTS-Response (Encrypted Message)
19: Parse-TLS-LogMessage()
20: Ack
21: Send-IM(Encrypted_Message)
[else]
22: Send-Error (Unauthorized Message)
23: Parse-TLS-LogMessage()
24: Ack

Figure 57 - IM Send

The following table identifies and describes the interactions between IEF components as illustrated.

Table 52 - IM Send Messages

#	Messages			
1	Name	Send-IM()		
	Description	The user prepares and marks the text message and sends it to a designated chat-room.		
	Sender	IM-Client	Receiver	IM-PEP
2	Name	Gather metadata		
	Description	The PEP gathers the metadata from the message header: • User information; • Markings (e.g., special classifications or restrictions); • Designated chat-room.		
	Sender	IM-PEP	Receiver	IM-PEP
3	Name	ISSG-Request (User, Chat-Room and IM service)		
	Description	The PEP gathers the information needed to prepare a request for the user's identity, and chat-room information. The PEP then packages and issues an ISSG-Request to the users security infrastructure to gather the authorizations for the user, chat room and IM services.		
	Sender	IM-PEP	Receiver	ISSG
4	Name	ISSG-Response (Authorizations)		
	Description	The ISSG transforms the PEP request into the appropriate format for the specified user service (e.g., ICAM service or TrustMark Registry), and issues the request to that service. The response from the ICAM services is translated into the form to ISMB response requirements. The ISSG then packages and publishes an ISSG-Response message and issues it to the Messaging-PEP for processing.		
	Sender	ISSG	Receiver	IM-PEP
5	Name	TLS-LogMessage ()		
	Description	The ISSG records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	ISSG	Receiver	TLS
6	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to the ISSG.		
	Sender	TLS	Receiver	ISSG
7	Name	PDP-Request ()		

Table 52 - IM Send Messages

#	Messages			
	Description	The PEP gathers the sensitivity metadata, the user's authorization and the chat-room authorization from the message. It packages this information as a PDP-request and issues the request to the PDP for adjudication and determination.		
	Sender	IM-PEP	Receiver	PDP
8	Name	PDP-Response		
	Description	The PDP adjudicates the requested authorization using the current user policies. Upon completion, it packages the decision and release instructions as a PDP-AuthorizationResponse and issues to the PEP for processing.		
	Sender	PDP	Receiver	IM-PEP
9	Name	TLS-LogMessage ()		
	Description	The PDP records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	PDP	Receiver	TLS
10	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to PDP.		
	Sender	TLS	Receiver	PDP
11	Name	ISSG-Request (New key and Token)		
	Description	If the user identifies that the message has special restrictions (limited access within the chat-room participants) - the PEP requests a unique cryptographic key-token pair for the message. Typically, there is a single key-token pair for the message. The PEP packages an ISSG-Request for a new Cryptographic key and token from the users Key Management Services.		
	Sender	IM-PEP	Receiver	ISSG
12	Name	ISSG-Response (Key, Token)		
	Description	The ISSG transforms the new key request into a message to the user's key management services and issues the message. Upon receipt of the Key and Token, the ISSG packages an ISSG-Response and issues it to the PEP.		
	Sender	ISSG	Receiver	IM-PEP
13	Name	ISSG-Request (Chat roomToken)		
	Description	If this is a standard message to the chat-room, the PEP packages an ISSG-Request (with chat-room token) sends the request to the ISSG.		
	Sender	IM-PEP	Receiver	ISSG
14	Name	ISSG-Response (Token, Key)		
	Description	The ISSG transforms the key request into a message to the user's key management (escrow) service and issues the message to the User specified service. Upon receipt of the Key, the ISSG packages an ISSG-Response and issues it to the PEP.		
	Sender	ISSG	Receiver	IM-PEP
15	Name	TLS-LogMessage ()		

Table 52 - IM Send Messages

#	Messages			
	Description	The ISSG records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	ISSG	Receiver	TLS
16	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to the ISSG.		
	Sender	TLS	Receiver	ISSG
17	Name	CTS-Request (Key, Token, Message)		
	Description	The PEP packages the key, and message as a CTS-Request, and issues it to the CTS for encryption.		
	Sender	IM-PEP	Receiver	CTS
18	Name	CTS-Response (Encrypted Message)		
	Description	The CTS encrypts the message, packages it as a CTS-Response and issues it to the PEP.		
	Sender	CTS	Receiver	IM-PEP
19	Name	TLS-LogMessage ()		
	Description	The CTS records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	CTS	Receiver	TLS
20	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to the CTS.		
	Sender	TLS	Receiver	CTS
21	Name	Send-IM(Encrypted_Message)		
	Description	The PEP repackages the instant message (with the encrypted payload), and issues it to the IM-Server for dissemination to the chat-room participants.		
	Sender	IM-PEP	Receiver	IM-Server
22	Name	Send-Error (Unauthorized Message)		
	Description	If the User, Chat-Room or IM-Service is not authorized to handle the content in the message, the message is rejected and an Error Message is issued to the Users IM-Client application.		
	Sender	IM-PEP	Receiver	IM-Client
23	Name	TLS-LogMessage ()		
	Description	The PDP records the transaction to the TLS by preparing and issuing a TLS-LogMessage.		
	Sender	IM-PEP	Receiver	TLS
24	Name	Ack		
	Description	(Optional, if required by the users messaging protocol) The TLS acknowledges the receipt of the message to the PEP.		

Table 52 - IM Send Messages

#	Messages			
	Sender	TLS	Receiver	IM-PEP

Annex E – Glossary (Informational)

Term	Definition
Accurate	Free from error or defect; consistent with a standard, rule, or model; precise; exact.
Acknowledge	An instruction to the recipient of an information exchange directing the issuance of an acknowledgment to the receipt of the information to the provider of the information.
Action Instruction	An instruction directing the producer or receiver of a message to take a specific action, (1) message specific rules governing the release of the information, or (2) message specific actions to be taken upon receipt of the message.
Active Policy	The set of policies (/rules) that are instantiated and set “active” for one or more of the environment policy enforcement and decision points.
Adaptive Information Sharing	The ability to selectively share information content based on operational or business context (e.g., roles, relationship, risks, threats, severity, scale, and trust). This includes the ability of users (manually) or systems (automatically) to adjust active ISS Policies to accommodate changes in business and operational context.
Aggregation	Defines the process through which data elements are combined to referentially and semantically complete data sets.
Asymmetric Information Sharing	The ability to share content with different communities, agencies or individuals conforming to legislative, regulatory, policy, contractual or service level requirements – while leveraging standard or shared protocols, interfaces and infrastructure.
Attachment	A computer or electronic file (typically unstructured) sent with or included in a message, email or instant message.
Attachment Element	A binary file (e.g., PDF file, image or video) or document, and information about the binary or document, such as the size and type and description. Source: Logical Entity Exchange Specification (LEXS).
Attachment Semantic	A Semantic that specifies the rules for assembling the attachments to a message. It also provides the rules for generating an attachment summary and linkages.
Attachment Specification	A specification of the rules governing attachment of binary information elements to an information exchange or message.
Attachment Summary	A summary or list of attachments for a specific data package.
Attribute	A defined property of an entity, object, triple, schema, etc. Source: A Dictionary of Computing. Oxford University Press, 2008. Oxford Reference Online. Oxford University Press.
C2	Command and Control.
C4I	Consultation, Command, Control, Communications and Intelligence.
Caveat	Markings (e.g., meta-data, tags, or labels) specifying a restriction or warning order pertaining to a specific data / information element, or operating environment.

Term	Definition
Caveat Separation	The process for selective exchange of information based on security policy and security profiles of the information and consumer of the information. Caveat separation may apply to data elements with the information or the aggregation of information.
Challenged Networks or Communication	Under operational conditions most frontline communications are provided by radio (HF, VHF, or HCDR). These forms of communications are inherently less robust than the Wi-Fi and wired networks realized by most organizations. Challenged refers to the reality that these networks: • Have limited bandwidth capability (as low as 1Kb/Sec); • Are prone to outages (e.g., range limitations, jamming, and voice override); • Large node count; and • Packet loss.
Classified Information	Sensitive information to which access is restricted by law or regulation to particular classes of persons. A formal security clearance is required to handle classified documents or access classified data.
Common Operating Picture	A collaborative set of technologies that provide the user(s) with a shared understanding of the operational environment including: Threats; Opportunities; Resources; Situational Awareness and other relevant information. The technologies combine to integrate perspectives; deliver actionable knowledge and structure information to the specific User(s) needs.
Common Representational Operating Picture	Is equivalent to the COP but limits access to that information required to exercise the role or feature of the user.
Communication Channel	A means of communication or access. For the purposes of this specification communication channels will be limited to the middleware used to move information between suppliers (/publishers) and consumers (/subscribers).
Community	A community of interest or community of practice.
Community of Interest	A collaborative group of users that must exchange information in pursuit of its shared goals, interests, missions, or business processes and therefore must have shared vocabulary for the information exchanges. DoD 8320.2, December 2, 2004. Or, a group of people interested in sharing information and knowledge in a particular topic or domain of discourse.
Community of Practice	Informal, self-organized, network of peers with diverse skills and experience in an area of practice or profession. Such groups are held together by the members' desire to help others (by sharing information) and the need to advance their own knowledge. Or, a group of people or organizations that are active practitioners in a domain of activity (e.g., Emergency Management, Law Enforcement, Military Operation) and where it is not appropriate for non-practitioners to participate.
Conceptual Interoperability	The assumptions and constraints of the meaningful abstraction of reality – are aligned, the highest level of interoperability is reached. This requires that conceptual models are documented based on engineering methods enabling their interpretation and evaluation by other engineers.
Confidential Information	Privileged communication shared with only a few people for furthering certain purposes, such as with an attorney for a legal matter, or with a doctor for

Term	Definition
	treatment of a disease. Receiver of confidential information is generally prohibited from using it to take advantage of the supplier of that information.
Content Centric	Refers to both information-centric and data-centric.
Contract	(Source: SOPES and UPDM) A contract represents a grouping of Semantic construction rules and information flow controls which specify a formal information sharing agreement between two or more operational nodes or participants in a domain or community. Equivalent terms in this specification are Information Exchange Agreement, Information Exchange Specification and Information Exchange Contract.
COP	Common Operational Picture.
CP	Compliance Point.
Crisis Management	Coordinated actions taken to diffuse crises, prevent their escalation into armed conflict and/or contain resulting hostilities. The crisis management machinery provides decision-makers with the necessary information and arrangements to use appropriate instruments (political, diplomatic, economic, and military) in a timely and coordinated manner. (MC 400/1).
CRO	Crisis Response Operation.
CROP	Common Representative Operational Picture.
CTS	The Cryptographic Transformation Service (CTS) is the IEF component that encrypts and decrypts InformationElements as authorized by policy.
Data	Facts used usually to calculate, analyze, or plan.
Data Centric	Enforce policies/rules against individual data assets; often referring to metadata or tags included within an information asset.
Data Composite	A data set resulting from the aggregation of data elements.
Data Integration	The process of combining two or more data elements from separate sources into a single semantically and referentially complete piece of information (or business object).
Data Integrity	Compliance to the allowable types, ranges or domain values for each data element (or attribute).
Data ownership	Identification that the data or information is controlled by the entity in such a way that only that entity is allowed to modify the data or information elements.
Data Packaging	See Information Packaging.
Data Pattern	A plan, diagram, or model to aggregate data elements.
Data Stewardship	Accountable for integrity and quality of data.
Data Creator Metadata	Metadata tags and markings that identify the creator of data or information elements.
DataElement	Representation of information (data) in a formalized manner suitable for communication, interpretation, or processing by humans or by automated means. In the context of IEPPV, data elements are atomic facts. Derived from UPDM.
Data Owner Metadata	Tags and markings that identify the owner or steward of the data or information elements.

Term	Definition
Data Sensitivity	Metadata describing the sensitivity (Privacy, confidentiality, classification or legal significance) of the InformationElement.
Deadline	A QoS attribute describing the latest acceptable time for the occurrence of certain events.
Decision Advantage	Enable commanders and/or decision makers, based upon information advantage and situational understanding, to make effective and informed decisions more rapidly than their adversary, thereby allowing one to dramatically increase the pace, coherence, and effectiveness of operations.
Decrypt	To convert encrypted text into its equivalent plain text by means of a crypto-system and its key or password.
Defense-in-Depth	(1) The coordinated application of multiple security services (countermeasures) to protect the integrity of the assets and resources of an enterprise. The strategy is based on the principle that it is more difficult for an adversary to defeat a complex and multi-layered defense than to penetrate a single barrier.
	(2) A layering of information safeguards to protect a specific information asset based on the reported value or key of that asset (e.g., security and privacy tags) bound to the instance of the information or data element.
	(3) Layer of security services that directly apply security policy to data and information elements based on the sensitivity of individual data and information elements and the authorizations of the publisher and each recipient.
Definition	A representation of a concept by a descriptive statement which serves to differentiate it from related concepts.
DEM	Data Exchange Mechanism.
DHS	Department of Homeland Security.
Digest	An information structure, format and syntax common to all communities. It provides the ability for systems to handle heterogeneous data without having to understand the specific context and or semantics of the source. As long as the entities relevant to the packaged data items are represented in the Digest, users will be able to discover, link, map, etc. to the information within. The Digest provides the common level of understanding, it does not mean that all sources have to populate all elements, or that all consumers have to use all elements; merely that at a schema level all applications understand the Digest. Implementers only need to build one module in order to produce or consume a basic set of data understandable by many. It also means that implementers do not have to develop large applications for each exchange, but rather build one that handles the basics and then additional smaller modules in order to produce or consume more complex exchanges. The objective of the Digest is to present the most common characteristics of real-world objects that can be supported by any data source or data consumer. Digest-level data objects may be further augmented or described with additional details in included packages or narrative text integrated into the message. The information in the digest must be semantically complete for both the data source or data consumer; the information package contents may rely on the digest to complete its semantics. The enforcement of a "Digest Semantic" by a software service will result in the generation of the digest for the instance of the Information Package. In other

Term	Definition
	applications, where the digest is not used, the “Payload” comprises the entire data portion of the message content.
Distribution Specification	A specification of the rules governing the assignment of InformationElements to a specific information dissemination service (e.g., User Application, Service Interface, and Middleware).
DND	Department of National Defence.
DNDAF	Department of National Defence Architecture Framework.
DOD	Department of Defense.
DODAF	Department of Defense Architecture Framework.
Domain	A sphere of knowledge or information identified by a name.
DTF	Domain Task Force.
Dynamic Coalition	A dynamic coalition may be formed spontaneously, members may join and leave without warning, and the nature of the relationships between participants may vary dramatically across the coalition as well as throughout its lifetime.
Dynamic Interoperability	As a system operates on data over time, the state of that system will change, and this includes the assumptions and constraints that affect its data interchange. The systems are able to identify the state changes in the assumptions and constraints and they can adjust or be adjusted to address changes in context or situation. The effect of the information exchange within the participating systems is unambiguously defined.
EDXL	Emergency Data Exchange Language Distribution Element: • EDXL Common Alerting Protocol (EDXL-CAP); • EDXL Distribution Element (EDXL-DE); • EDXL Hospital AVailability Exchange (EDXL-HAVE); • EDXL Resource Messaging (EDXL-RM); • EDXL Reference Information Model (EDXL-RIM); • EDXL Situation Reporting (EDXL-SitRep); and • EDXL Tracking Emergency Patients (EDXL-TEP).
Encrypt	To alter (encode) data using a mathematical algorithm so as to make the data unintelligible to unauthorized users while allowing a user with a key or password to convert the altered data back to its original state.
Encrypted Payload	An encrypted information asset conveyed as part of a Message.
Features	Within the scope of this specification, this term refers to software functions, services, or methods used to deliver the specified capability.
File	A collection of information, referred to by file name; for example, a user-created document, program data, or the program itself. With a program, the information is held on backing store (i.e., usually on magnetic disk) in order (a) to enable it to persist beyond the time of execution of a single job and/or (b) to overcome space limitations in main memory. Files with a very brief existence (i.e., in case (b) above, or where they simply carry information between one job and the next in sequence) are called work files. See also master file, data file.

Term	Definition
	Source: A Dictionary of Computing. Ed John Daintith and Edmund Wright. Oxford University Press, 2008. Oxford Reference Online.
Filter	A profile or script containing the rules to restrict the assembly of data or information elements.
Gateway	An IEF component that connects the ISMB and the users' network and infrastructure. The gateway provides a single integration point for security services hosted on other parts of the user environment and provides the ability to pass message traffic, security redaction and filtering, proxies or protocol translations at various network layers.
Hash Digest	All Secure Asset Containers (SAC) include a digest that is calculated at creation time as a mechanism to detect tampering of the container. The digest is calculated using the SHA-512 hash algorithm. Digest calculation is dependent on the order in which the source material is submitted for the calculation. For SAC's, the digest is calculated as follows: • The contents of the encrypted file; • The caveat for the original file; • The token for the key; • The filename of the original file; and • The key used to encrypt the original file.
HCDR	High Capacity Digital Radio.
HF	High Frequency.
ICAM	Identity, Credentials and Access Management.
IE	Information Exchange.
IEA	Information Exchange Agreement.
IEDM	Information Exchange Data Model.
IEF	Information Exchange Framework.
IEM	Information Exchange Mechanism.
IEPAS	Information Exchange Policy-based Authorization Service(s).
IEPPS	Information Exchange Policy-based Packaging & Processing Service.
IEPPV	Information Exchange Packaging & Processing Policy Vocabulary.
IES	Information Exchange Specification.
Information	(1) Data in Context; or (2) Composite of data elements used to inform a decision.
Information Advantage	Enable the provision of information needed to develop a degree of control in the information domain that permits the conduct of operations without effective opposition.
Information Artifact	A composite of data elements that satisfy the Semantic construction rules for an agreement to exchange information between a supplier and a consumer.

Term	Definition
Information Centric	Enforce policies/rules against individual information assets (assemblies of data elements that satisfy information-sharing requirements).
Information Consumer	Any User, System Application, Channel or Node using information managed by the IEF.
Information Exchange Policy Set	A general term identifying a group of Information Exchange Policies exchanged between the IEF components that include: • Rules and constraints governing the packaging and processing of data and information elements; and • Rules and constraints governing the release and distribution of information (Semantic) elements. Information Exchange Policy represent a serialization of the Policy Models defined using the Information Exchange Packaging Policy Vocabulary (IEPPV).
Information Exchange Specification	An agreement between an information supplier and information consumer to exchange selected information, based on a specified format, protocol and communication link. Core Element of the Information Exchange Packaging Policy Vocabulary.
Information Package	A standard representation of structured, semi-structured, and binary information applicable to an information sharing agreement. Packages may contain metadata, a Digest, a Structured Payload, Rendering Instructions, and optional linkages depending on the established agreements.
Information Payload	A formatted dataset without protocols and metadata required for an information exchange. Derived from: Body (payload) The part of a cell or packet in a network that holds the information supplied by the end-user for transmission from the sender to the receiver. A Dictionary of Computing. Ed John Daintith and Edmund Wright. Oxford University Press, 2008. Oxford Reference Online. Data Payload: Refers to the “actual data” in a packet or file minus all headers attached for transport and minus all descriptive meta-data. In a network packet, headers are appended to the payload for transport and then discarded at their destination. In a key-length-value structure, the key and length are descriptive data about the value (the payload).
Information Packaging	The process of assembling (aggregating, transforming, tagging/marking and redacting/filtering) data and information elements and formatting them to service a specific information exchange requirement.
Information Processing	The parsing, transformation and marshaling of information and data elements to information or data store(s).
Information Producer	This includes any user, application or system producing information for distribution or dissemination.
Information Quality	Describes the ability of organizations, systems and persons to provide information that is:

Term	Definition
	• Trustworthy: Information quality and content can be trusted by stakeholders, decision makers and users; • Relevant: Information content tailored to specific needs of the decision maker; • Timely. Information provided when and where it is needed to support the decision-making process; • Usable: Information is presented in a common functional format, easily understood by the decision makers and their supporting applications; • Complete: Information that provides all necessary and relevant data (where available) to facilitate a decision; • Concise: Information is provided in a form that is brief and succinct, yet including all important information; • Trusted: Information that is accepted as authoritative by stakeholders, decision makers and users; and • Secure: Information is protected from inadvertent or Malicious Release to unauthorized persons, systems or organizations.
Information Recipient	Any User, System Application, Channel or Node using information managed by the IEF.
Information Supplier	This includes any user, application or system supplying information for distribution or dissemination.
Information Element	An item of information that flows between operational activities and nodes. For IEPPV, an information element refers to a grouping of data elements (including other information elements) providing meaning within the context of an operation or situation. Derived from: MODAF: A formalized representation of information subject to an operational process. DoDAF: Information that is passed from one operational node to another. Associated with an information element are such performance attributes as timeliness, quality, and quantity values. (DoDAF) Information Exchange: The collection of information elements and their performance attributes such as timeliness, quality, and quantity values. (DoDAF). Note: Within the architectural context of the UPDM, SOPES, and IEPPV, the Information element provides a description of, or specification for, the data or information processed or exchanged.
Instruction	The description of an operation that is to be performed by a computer or human operator. Derived from: “The description of an operation that is to be performed by a computer. It consists of a statement of an operation to be performed and some method of specifying the operands (or their locations) and the disposition of the result of the operation.” A Dictionary of Computing. Ed John Daintith and Edmund Wright. Oxford University Press, 2008.
Instrument	See Policy Instrument.
ISE	Information Sharing Environment.
ISMB	IEF Secure Messaging Bus.

Term	Definition
ISS	Information Sharing and Safeguarding.
ISSG	The IEF Security Services Gateway (ISSG) provides a single point of integration between IEF components and the users' security services and infrastructure.
KMS	Key Management Services.
Legally Significance	Having importance to a legal proceeding or action.
Legally Significant Information	Information that must be captured, maintained and protected in order to inform a legal proceeding or action.
LEISP	Law Enforcement Information Sharing Program.
Levels of Interoperability	The level to which practices and services deliver the ability and capacity to ensure the right information is available to the right people or system at the right time. • Level 0: Stand-alone systems have No Interoperability or Integration. • Level 1: On the level of Technical Interoperability, a communication protocol exists for exchanging data between participating systems. • Level 2: The Syntactic Interoperability level introduces a common structure to exchange information; i.e., a common data format is applied. • Level 3: If a common information exchange reference model is used, the level of Semantic Interoperability is reached. On this level, the meaning of the data is shared; the content of the information exchange requests are unambiguously defined. • Level 4: Pragmatic Interoperability is reached when the interoperating systems are aware of the methods and procedures that each system is employing. • Level 5: As a system operates on data over time, the state of that system will change, and this includes the assumptions and constraints that affect its data interchange. If systems have attained Dynamic Interoperability, they are able to comprehend the state changes that occur in the assumptions and constraints that each is making over time, and they are able to take advantage of those changes. • Level 6: Finally, if the conceptual model – i.e. the assumptions and constraints of the meaningful abstraction of reality – are aligned, the highest level of interoperability is reached: Conceptual Interoperability.
LEXS	Logical Entity eXchange Specification.
LEXS(2)	LEISP Exchange Specification.
Local Policy Store	A storage location local to (or part of) the PDP or IEPPS for the current set of policies. This store may comprise a memory-based, file-based or databased configuration based on the implementors' design considerations.
Local Services	Services defined and used by the user community to perform a specific set of functions.
Marshaling	Defines the process through which data sets are divided and put into the data elements described by the underlying data store(s).

Term	Definition
MDA	Model Driven Architecture.
MEM	Message Exchange Mechanism.
Memorandum of Understanding	A bilateral or multilateral agreement between parties.
Message	A formatted InformationElement transferred by a message switching system (or Network). Messages may be of any length, from a few bits to a complete file, and no part of a message is released to its final recipient until all of the message has been received at the network node adjacent to the destination. Source: A Dictionary of Computing. Ed John Daintith and Edmund Wright. Oxford University Press, 2008. Oxford Reference Online.
Message Element	An identifiable part of a message structure containing contextually relevant data or information elements. Message elements are integrated and formatted in accordance with contract or information exchange specification rules and instructions prior to release.
Messaging Protocol	The rules, formats and functions for exchanging messages between the components of a messaging system.
Metadata	Data (tags and markings) that describes other data.
Middleware	Software that serves as an intermediary between systems software and an application.
MILS	Multi-Independent Levels of Security.
MIP	Multilateral Interoperability Programme.
MLS	Multi-level Security.
MOD	Ministry of Defence.
MODAF	Ministry of Defence Architecture Framework.
MOF	Meta-Object Facility.
MOU	Memorandum of Understanding.
Multi-Independent Levels of Security	Multiple Independent Levels of Security/Safety (MILS) is a high-assurance security architecture based on the concepts of separation and controlled information flow; implemented by separation mechanisms that support both untrusted and trustworthy components; ensuring that the total security solution is non-bypassable, evaluable, always invoked and tamperproof. A MILS system employs one or more separation mechanisms (e.g., Separation kernel, Partitioning Communication System, physical separation) to maintain assured data and process separation. A MILS system supports enforcement of one or more application/system specific security policies by authorizing information flow only between components in the same security domain or through trustworthy security monitors (e.g., access control guards, downgraders, crypto devices, etc.).
Multilevel Security	Multilevel Security (MLS): Information systems and networks that provide the ability to process information with incompatible classifications (i.e., at different security levels and caveats). These systems and networks permit access to

Term	Definition
	information elements by authorized users (users holding the appropriate security clearances and needs-to-know), and prevent access to users that lack authorization.
NAF	NATO Architecture Framework.
NATO	North Atlantic Treaty Organization.
NGO	Non-Government Organization.
OCL	Object Constraint Language.
Octet	An octet is a unit of digital information in computing and telecommunications that consists of eight bits. The term is often used when the term byte might be ambiguous, since historically there was no standard definition for the size of the byte.
OctetSeq	A variable-length sequence of octets, as in Abstract Syntax Notation One (ASN.1), is referred to as an octet string.
OctetString	A variable-length sequence of octets, as in Abstract Syntax Notation One (ASN.1), is referred to as an octet string.
ODM	Ontology Definition Meta-model.
Ontology	In the context of knowledge sharing, the term ontology means a specification of a conceptualization. Ontology is a description (like a formal specification of a program) of the concepts and relationships that can exist for an agent or a community of agents. This definition is consistent with the usage of ontology as set-of-concept-definitions, but more general. And it is certainly a different sense of the word than its use in philosophy
OODB	Object Oriented Database.
OODBMS	Object Oriented Database Management System.
Operating Concept	It describes the operating characteristics for a system, typically from the viewpoint of the individuals who will use that system. It also describes how the set of systems capabilities (e.g., services, decision & enforcement points and interfaces) may be employed to achieve a desired objective or end state. Ideally it offers a clear methodology to realize the goals and objectives for the system, while not intending to be an implementation or transition plan itself. The following elements may be included: • Statement of the goals and objectives; • Operational conditions/contexts affecting the system; • Organizations, activities, processes and interactions among participants using the system; • Specific operational concept and processes for fielding the system; and • Processes for initiating, developing, maintaining and adapting the system.
Operation	For the purpose of this specification the term operation is restricted to events and activities describing a Crisis Response Action including Military.

Term	Definition
Operational Context	A set of network, node, system, application or user characteristics that define the current state of dynamically evolving operational conditions. Operational context data may include: • Role and Responsibility; • Phase of the operation; • Operational Threat; • Operational Risk; • Command Intent; • Physical location; • Available communications links; and • Access device.
Operational Domain	The sphere of knowledge, influence, or activity for a specific mission or operation.
ORDBMS	Object-Relational Database Management System.
OWL	Web Ontology Language.
Packages	For the purposes of this specification, "Package" refers to the process of aggregating, transforming, marking, redacting, structuring, and formatting data for access or release.
PAP	The Policy Administration Point (PAP) provides an authorized user (administrator) with an interface to access services to manage and administer the configuration and policy environments for IEF Components in its designated operating environment.
Participant	A list of entities to produce or receive the information or message.
Pattern	A plan, diagram, or model to be followed in making things (in this instance – dataset conforming to information sharing and safeguarding agreement).
PDP	The Policy Decision Point adjudicates access to, or the release of InformationElements to specified users.
PDU	Protocol Data Unit.
PEP	A Policy Enforcement Point (PEP) intercepts each InformationElement transiting between a user client application and the server (Email, Instant Messaging and File Share, and Data) to ensure the requesting user is authorized to perform the requested action on the specified information element(s).
Personal Identifiable Information	Information that can be used on its own or with other information to identify, contact, or locate a single person, or to identify an individual in context.
PII	Personal Identifiable Information.
PIM	Platform Independent Model.
Planned Incident	An incident for which there exists standard operating procedures or safeguards to mitigate or recover from the impact of the incident.

Term	Definition
Planned Threat	A threat for which there exists standard operating procedures or safeguards to prevent or mitigate the impact of the threat.
Policy	A definite course or method of action selected from among alternatives and in light of given conditions to guide and determine present and future decisions. Within this document it refers to: "a defined course or method of action in response to a request for or change in information or data. Within the context of this specification, "specification of a method of action for aggregating, transforming and filtering data and information elements to conform to stipulated Semantic construction rules for an information sharing agreement or Community of Interest". Or, a definite course or method of action selected from among alternatives and in light of given conditions to guide and determine present and future decisions (Webster Merriam Dictionary).
Policy automation	The use of software services to automate the selection of a course of action (decision) and the execution of that selected course of action (execution). For the purpose of this specification, this refers to the actions to be taken by IEF services (including decision and enforcement points) to share and safeguard information assets.
Policy Driven	A process through which user defined policy instruments are translated into machine readable rules (/instructions) and enforced by software services and systems. This process results in full traceability from policy instrument to implementation (policy decisions and enforcement points).
Policy Instrument	A formal business document used by an organization to direct and describe methods or processes to be used or applied. These instruments may include: Legislation, regulation, agency policy, memorandum of understanding (MoU), and service level agreements (SLA). Or, formal documents describing a plan of action by an individual agency or community to handle information sharing and safeguarding (e.g., legislation, regulation, memorandum of understanding and service level agreements).
Policy Management Environment	Standards, tools, techniques and technology used to develop and test ISS policy sets for one or more missions or operations.
Policy Model	An architectural model aligning information sharing and safeguarding instruments with a specific data domain.
Policy Transformation	The single or multi-stage transformation of policy instruments into machine readable and enforceable rules.
Policy-Right	A permission granted through the application of a policy.
PPS	The Policy-based Packaging and Processing Service (PPS) transitions structured InformationElements (e.g., NIEM, EDXL, and HL7) between data stores and information exchange services in accordance with local policy conforming to the Information Exchange Packaging Policy Vocabulary (IEPPV).
Pragmatic Interoperability	The systems are aware of the methods and procedures that each system is using. The use of the data – or the context of its application – is understood by the participating systems; the context in which the information is exchanged is unambiguously defined. This layer puts the (word) meaning into context.
Privacy Metadata	Tags and/or markings that support the enforcement of privacy policy.

Term	Definition
Private Information	Information about behavior that occurs in a context in which an individual can reasonably expect that no observation or recording is taking place, and information which has been provided for specific purposes by an individual, which the individual can reasonably expect will not be made public.
Proprietary Information	Privately owned knowledge or data, such as that protected by a registered patent, copyright, or trademark.
Protocol Data Unit	Binary variable length messaging protocol used by the MIP Data Exchange Mechanism.
PSA	Public Safety Agency.
PSM	Platform Specific Model.
Publisher Metadata	Tags and markings that support the publishing of sharable information to a data registry, repository, or publication-subscription middleware infrastructure. This metadata provides the structures required to represent the data as well as that associated with publishing and storage of data. The data registry, repository, or middleware receives and records the published metadata in a manner for users and systems to discover the associated information elements.
PVO	Private Volunteer Organization.
QOS	Quality of Service.
QoS History	A record of past information generated by the system that is kept around for the benefit of applications that are late joining the network.
Quality Information	See Information Quality.
Quality of Service	A set of attributes that can be used to define the middleware's capabilities to meet the requirements of the application for the purpose of data-delivery or management such as reliability, ownership policy, history size, time-to-keep, etc.
RDF	Resource Description Framework.
Real-time	Refers to the event-triggered (e.g. data change) global update of information across all nodes, systems and applications requiring access to the information.
Redact	To obscure or remove (text or data) from a document prior to publication or release. This feature is typically performed by data filters.
Reference Architecture	Defines abstract architectural elements within the domain in terms that are independent of specific technologies, protocols, and tools.
Reference Model	Illustrates the interaction between IEF components when processing a message received for a PPS operating within the IEF's environment.
Releasable Dataset	A collection of data elements that can be provided to the recipient(s) as defined by policy.
Releasable Message	A message where the content can be provided to the recipient(s) as defined by policy.
Reliability	A QoS attribute describing the guarantees and feedback provided to the application regarding the delivery of the information supplied to the middleware.
Responsible Information Sharing	Compliant with law, regulation and policy; consistent with community and agency strategy and direction, to include protection of information, sources and methods, and civil liberties and privacy; and accountable through governance

Term	Definition
	and oversight - maximize the quantity and quality of information that is discoverable and accessible to authorized users and partners. Compliant with legislation, regulation and policy; consistent with agency strategy, policy and direction; and accountable through governance and oversight: • Maximize the volume, variety and quality of information that is discoverable and accessible by authorized users; • Protect sensitive (classified, private, confidential and legally significant) information from unauthorized access/release and tampering; • Protect information sources and processing-methods; • Protect civil rights/liberties; and • Ensure that information is assured in its content, safe in transmission and use, and safeguarded from the threat of malicious acts, unauthorized use, clandestine exfiltration or compromise by remote intrusion.
SA	Situational Awareness.
SAC	Secure Asset Container.
Safeguard	Policies, rules, services, and technologies that serve to guard or protect data and information elements from malicious or inadvertent release of sensitive or protected information.
Secure Asset Container	An envelope that allows some unprotected information to exist outside of the protected (encrypted) payload. In addition to the encrypted payload, the envelope includes an envelop header containing metadata that enables the identification, and discovery of the information in the container, the securing of the InformationElement and the container itself.
Security Filter	A specialization of a filter that provides the rules that restrict the assembly of data and information elements based on the values of a security tag or label.
Security Metadata	Tags and markings that assist in the enforcement of security policy and malicious or inadvertent release of classified information to unauthorized recipients.
Semantic Integrity	Compliance to the structure, format and content (mandatory or optional) for information sets (or business objects).
Semantic Interoperability	Semantics concerns the study of meanings. Semantic interoperability refers to the ability of information systems to exchange information/data with unambiguous, shared meaning. It is a requirement to enable information integration, machine analytics, inferencing, knowledge discovery, and data federation. Semantic interoperability is not only concerned with the packaging of data (structure and syntax), but the simultaneous provision of intent and meaning (semantics).
Semantic Pattern	A plan, diagram, or model to aggregate Transactional patterns that conform to an information sharing and safeguarding agreement; or

Term	Definition
	Data patterns that describe a set of data that conforms to an information sharing specification. It comprises a SemanticElement enclosing a specified set of TransactionalElements and WrapperElements described in PPS policy conforming to the IEPPV.
Sensitive Information	Information elements identified as classified, private, confidential or legally significant.
Sensitivity Markings	A general reference to an InformationElement's Security Level [1], caveats [0..*], Privacy Markings [0..*], and Legal Significance Marking [0..*].
Service Level Agreement	An agreement between two or more parties where the level of service is formally defined.
Session	The software connection to the information dissemination services to be used for the exchange of information under the InformationExchangeSpecification. Derived from the Seven Layer Reference Model: 1. Session Layer - Identifies the service of binding two presentation service entities together logically and controls the dialogue between them as far as message synchronization is concerned. 2. Presentation Layer - Provides a set of services that may be selected by the application to enable it to interpret the meaning of the data exchanges. Such services include management of the entity exchange, display, and control of the structured data. The presentation layer is the heart of the seven-layer proposal, enabling disparate terminal and computer equipment to intercommunicate. A Dictionary of Computing. Ed John Daintith and Edmund Wright. Oxford University Press, 2008. Oxford Reference Online. Oxford University Press.
SIEM	Security Information and Event Management.
SLA	Service Level Agreement.
SOPES	Shared Operational Picture Exchange Services.
Source Data	Raw data (sometimes called source data or atomic data) is data that has not been processed for use. A distinction is sometimes made between data and information to the effect that information is the end product of data processing.
Specialized Data Set	A collection of data that is specifically tailored to a specific context and recipient.
Specialized Message	A message for which the content is specifically tailored to a specific context and recipient.
Specification	Specification: A detailed precise presentation of something. Within the context of the IEPPV, a detailed and precise presentation of rules governing the assembly or processing of information elements.
SPI	Sensitive Personal Information; See PII.
Stage	To gather and prepare information for release to a community in accordance with established policy, memorandum of understanding or service level agreements.
Stakeholder	A person with an interest or concern in the effective application of ISS Policy.

Term	Definition
Static Filter	A filter created at design-time that cannot be modified at run-time.
Syntactic Interoperability	A common structure to exchange information; i.e., a common data format is applied. On this level, a common protocol to structure the data is used; the format of the information exchange is unambiguously defined.
Tear line	A physical line on a message or document separating categories of information that have been approved for disclosure and release.
Technical Interoperability	Technical Interoperability: An agreed communication protocol exists for exchanging data between participating systems. The protocol operates over an agreed and established communication infrastructure allowing systems to exchange bits and bytes, and the underlying networks and protocols are unambiguously defined.
TLS	The Trusted Logging Service (TLS) is a service that securely records IEF component activity as a transactional history of the policy decisions and access control enforcement.
Trust	Within the scope of this Specification – Trust refers to the level of confidence an information supplier has relating to the release of selected information to a specific consumer of that information.
UML	Unified Modeling Language.
Unplanned Incidents	An occurrence of an action or situation that is not addressed by plans or operating procedures.
Unplanned Threat	An expression of intention to inflict evil, injury, or damage that is not accounted for in the threat risk assessment or mitigation plans.
UPDM	Unified Profile for DODAF and MODAF.
User	A user is a participant that requests access to a resource (e.g., IEF Component, InformationElement, or DataElement), that has a specified set of privileges (e.g., policy rights, authorizations and attributes), that permit or deny access of that participant to services and/or resources (e.g., data elements, information elements, system devices, applications, and/or services). An authorized user may be a provider or a recipient of resources. Authorized users may include: • Individual (/person); • Organization; • Role; • Community; • Topic; • Queue; • Platform; • System; • Application; • Service (e.g., IEF Component); • Communication Channel;

Term	Definition
	- Session; and - Network. Each member of the identified categories must have credentials, attributes, authorizations, and/or policy rights that specify their rights to access the resource.
Validate	To give official sanction, confirmation, or approval to a specified element.
Verify	To ascertain the truth or correctness of, as by examination, research, or comparison of a specified element.
Vocabulary	A representation of a set of concepts by formal, descriptive statements which serves to differentiate those concepts from related concepts within a given domain or area of expertise. Terminological dictionary (3.7.1) which contains designations (3.4.1) and definitions (3.3.1) from one or more specific subject fields (3.1.2). NOTE: The vocabulary may be monolingual, bilingual or Multilingual. ISO 1087-1:2000.
WatchPoint	A trigger mechanism used by an application to commence the assembly of a TransactionalElement. A data model assigns this tagged value to a WrapperElement aggregation arc in the Transactional pattern. Additions to the underlying data store for this WrapperElement triggers the application to start building the composite. Derived from SOPES IEDM V1: Wrapper.
WrapperElement	A logical construct that wraps or encapsulates the definition of a data set, table entity, triple, file, etc. A Wrapper directly maps to a data instance (e.g., row of data in a database application) in the logical data model and the physical data model. Derived from SOPES IEDM V1: Wrapper.
XMI	XML Metadata Interchange.
XML	Extensible Markup Language.