Date: October 2025

Information Exchange Reference Architecture (IEF-RA)

Version 2.0 – beta 1

OMG Document Number: ptc/2025-10-01

Standard Document URL: https://www.omg.org/spec/IEF-RA/2.0/

This OMG document replaces the submission document (mars/2025-03-04). It is an OMG Adopted Beta Specification and is currently in the finalization phase. Comments on the content of this document are welcome and should be directed to issues@omg.org by July 2025.

You may view the pending issues for this specification from the OMG revision issues web page https://issues.omg.org/issues/lists.

The FTF Recommendation and Report for this specification will be published in April 2026. If you are reading this after that date, please download the available specification from the OMG Specifications Catalog.

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.

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 https://www.omg.org, under Specifications, Report a Bug/Issue.

Table of Contents

Preface viii

1 SCOPE 1
1.1 Organization of this Specification 2
1.2 Motivation 2
1.3 New Paradigm 4
1.4 IEF Approach 5
1.5 IEF Delivered Capabilities 5
1.6 IEF Objectives 6
1.7 IEF RA Assumptions 7
1.7.1 Data as a Strategic Asset 7
1.7.2 Interoperable vs Integrated Services 8
1.7.3 IEF Component Specifications 8
1.7.4 Error & Complex Conditions 8
1.7.5 File (Exchange Data Object) Metadata 8
1.8 IEF RA Design Principles 9
1.9 Adapting to change 9
1.9.1 Adapting During Operations 9
1.9.2 Adaptation during Design 10
1.10 Adapting to evolving ISS and DCS trends 10
1.10.1 IEF Alignment with Data-Centric Security 10
1.10.2 IEF Alignment with Zero-Trust Architecture 13
1.10.3 IEF Alignment with Cloud Security Practices 15
1.10.4 IEF Alignment to Data Centricity 16
1.10.5 IEF in Data Centricity, Digitization, and Digital Transformation 17
1.10.6 IEF Alignment with Secure Relationship Protocol 20
1.11 Alternate Configurations of IEF Services 21
1.11.1 Secure Data Service (SDS) 21
1.11.2 Direct PEP Access to Security Services 22
1.11.3 Streaming Data Configuration 23
1.11.4 External PAP 24
1.12 IEF Deployment Configurations 25
1.12.1 Micro-Service-Based PPS 25
1.12.2 Secure Data Service (SDS) 26
1.12.3 DCS Enable Data Lake Configuration 27

2 Conformance 28
2.1 Selecting a Compliance Point 28
2.2 Compliance Point Descriptions 28
2.2.1 Email Compliance Point 31
2.2.2 File Share Compliance Point 32
2.2.3 Instant Messaging or Chat Compliance Point 34
2.2.4 Data Messaging Compliance Point 35
2.3 Alternate PEP Configuration Structured Data Messaging Compliance Points 40
2.3.1 Multiple PEP for System-to-System Exchanges 41
2.3.2 Extended PEP Services 41
2.3.3 Secure Data Service 42

3 Normative References 44
3.1 Normative Specifications 44
3.2 Reference Materials (Informational) 44
3.3 Additional Specifications and Standards 45

4 Terms and definitions 47

5 Symbols 48

6 Additional Information 49
6.1 Intended Audience 49

6.2 Acknowledgements 49
6.3 Additional Materials 49
6.4 IEF RA Objective 49
6.5 Modelling Conventions 50
6.6 OMG Related Work 50

7 IEF Reference Architecture Specification 52
7.1 IEF-RA Scope 52
7.1.1 The IEF in Operations 52
7.1.2 IEF Component Details 53
7.2 IEF Component Characteristics 53
7.2.1 IEF Component Common Operations 53
7.2.2 IEF Component Specializations 57
7.2.3 IEF Internal Interface 62
7.3 Basic Deployments 64
7.3.1 IEF Unstructured Data Deployment 64
7.3.2 IEF Structured Data Deployment 65
7.4 Alternate Deployments 66
7.4.1 Direct PEP Integration 66
7.4.2 External Policy Administration Point 67
7.4.3 Secure Data Service (SDS) 68
7.5 IEF Component Core Functions 69
7.6 IEF Interfaces 71
7.6.1 SMB Interfaces and Component Interactions 72
7.6.2 PEP External Interfaces and Integration Points 83
7.6.3 Other External Interfaces and Integration Points 91

8 Policy Administration Point (PAP) 97
8.1 PAP Operations 98
8.2 Administer IEF Component Configurations 109
8.3 Administer Component Policy Operations 116
8.4 PAP Integration Configurations 120
8.4.1 Administer Component Policy Operations 120
8.4.2 Administer Component Policy Operations 120

9 Policy Decision Point (PDP) 122
9.1 PDP Operations 122
9.2 PDP Integration Configurations 127
9.2.1 PDP Integrated Into IEF 127
9.2.2 PDP Delivered as External Service 128

10 Policy Enforcement Point (PEP) 130
10.1 Core PEP Operations 130
10.2 Data PEPs 141
10.2.1 Email PEP 142
10.2.2 File PEP 154
10.2.3 Instant Messaging (Chat) PEP 165
10.2.4 Structured Messaging PEP 172
10.3 PEP General Configurations 184
10.3.1 SMB Integration to User Infrastructure 184
10.3.2 Direct Integration to User Infrastructure 187

11 Policy-based Packaging and Processing Services (PPS) 190
11.1 PPS Components 190
11.1.1 PPS Operations 191
11.1.2 ProcessReceivedMessage 193
11.1.3 ManageWrapperMemory 199
11.1.4 PackageReleasableData 205
11.1.5 InformationExchangeControl 209
11.2 PPS Configuration 226

12 Security Service Gateway (SSG) 228
12.1 SSG Component Operations 228
12.2 SSG Configurations 237

13 Cryptographic Transformation Service (CTS) 244
13.1 CTS Component Operations 244
13.2 CTS Configurations 249

14 Trusted Logging Service (TLS) 251
14.1 TLS Component Operations 251
14.2 TLS Configurations 256

15 Secure Messaging Bus (SMB) 259
15.1 Secure Messaging Bus (SMB) Configuration 259

16 SMB Data Structures (/Messages) 261
16.1 Secure Messaging Bus (SMB) 261
16.1.1 SMB Message 261
16.1.2 SMB Message Attributes 264
16.1.3 PAP Command Message 265
16.1.4 PAP Command Response Message 269
16.1.5 PAP Alert Warning Message 271
16.1.6 PDP Request Message 272
16.1.7 PDP Response Message 274
16.1.8 PPS Receive Message 276
16.1.9 PPS Request Message 278
16.1.10 PPS Publish Message 282
16.1.11 SSG Request Message 283
16.1.12 SSG Response Message 286
16.1.13 CTS Request Message 288
16.1.14 CTS Response Message 289
16.1.15 TLS Log Report Message 290
16.1.16 Ack Message 292
16.2 Metadata Patterns 292
16.2.1 Message Metadata 293
16.2.2 Information Element Metadata 294

Annex A: IEF-RA XSD and Data Types 296
Annex A.1 SMB Message XSDs (Informational) 296
Annex A.2 PPS Policy Memory Models (Informational) 296
Annex A.3 – PPS Policy XSD (Informational) 315
Annex A.4 Enumerations (Informational) 315

Annex B – Minimum Metadata (informational) 345

Annex C – Metadata Bindings (Informational) 347

Annex D – Glossary (Informational) 353

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 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 Meta-model), and industry-specific standards for dozens of vertical markets.

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

OMG Specifications

As noted, OMG specifications address middleware, modeling and vertical domain frameworks. All OMG Formal Specifications are available from this URL: *https://www.omg.org/spec*

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
9C Medway Road, PMB 274
Milford, MA 01757
USA

Tel: +1-781-444-0404
Fax: +1-781-444-0320

Email: *pubs@omg.org*

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

Issues

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

This page Intentionally left Blank

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 automating policy decisions and enforcement points to enable responsible information sharing across a wide range of operational scenarios. The IEF Reference Architecture (RA) guides the overall IEF effort, broadens the general understanding of ISS and Data-Centric Security (DCS) domain requirements, and guides the development of IEF implementation specifications. “This reference architecture does not provide sufficient detail to specify interoperable implementations. Other specifications in the IEF family will provide the requisite details, and these specifications align with the concepts and structures described in this framework and provide the requisite implementation details.”

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

1. Military (coalition and Civilian-Military) operations,
2. National Security,
3. Public Safety,
4. Crisis Management,
5. Border Security,
6. Emergency Management,
7. Peace Keeping and
8. Humanitarian Assistance.

Organizations conducting these missions and operations require the ability to share sensitive information (e.g., private, confidential, legally significant, and classified) securely with other agencies, other levels of government, and international and private sector partners. They also require the ability to:

1. Address planned and non-planned missions and operations,
2. Rapidly deploy and integrate a coalition/partner ISS capability,
3. Rapidly adapt ISS patterns to changing operational conditions and contexts:
    a. Commander’s intent (target outcomes),
    b. Coalition configuration, organizational structure, roles and responsibilities,
    c. Threats, risk, and severity,
    d. Operational stage,
    e. Plans, orders, and
    f. 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. Public Administration: government operations,
    b. Healthcare: Electronic Health Records (EHR),
    c. Finance: banking and insurance records,
    d. Justice: Criminal case files,
    e. Supply chain,
    f. Manufacturing and
    g. Customs and Immigration, and
3. The ability to log and audit the exchange of information holdings.

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

`

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 include:
    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.

1.1 Organization of this Specification

This specification includes sixteen (16) Clauses and four (4) Annexes:

- Clause 1: Provides an overview of the specification and its relationship to key IM/IT initiatives (e.g., Data-Centric Security, Zero-Trust, and Cloud).
- Clause 2: Defines the compliance points (e.g., email, chat, file share, and data messaging) for the IEF-RA.
- Clause 3: Identifies References for this specification.
- Clause 4: Identifies Terms and definitions used in various parts of the specification. This clause does not include concepts and properties comprising the IEF RA.
- Clause 5: Identifies any special Symbols/Acronyms used in developing this specification.
- Clause 6: Provides Additional Information about this specification.
- Clause 7: Provides an overview of the IEF Reference Architecture.
- Clause 8: Provides the core requirements for a Policy Administration Point (PAP).
- Clause 9: Provides the core requirements for Policy Enforcement Points (PEP).
- Clause 10: Provides the core requirements for a Policy Decision Point (PDP).
- Clause 11: Policy-based Packaging and Processing Services (PPS).
- Clause 13: Provides the core requirements for a Cryptographic Transformation Service (CTS).
- Clause 12: Provides the core requirements for a Security Services Gateway (SSG).
- Clause 15: Provides the core requirements for a Trusted (tamper-resistant) Logging Service (TLS).
- Clause 14: Provides the core requirements for the Secure Message Bus (ISMB).
- Clause 16: Provides the data syntax for ISMB Messages.
- Annex A: IEF-RA XSDs.
- Annex B: Identifies Minimum Metadata (Informational) used during IEF-RA testing.
- Annex C: Identify several Metadata Bindings (Informational) used for DCS and data exchange.
- Annex D: Provides a Glossary (Informational) for this specification.

1.2 Motivation

Numerous after-action and news reports on events such as SARS, the 2007 London subway 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, 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 immediately need information. These coalitions form with international cooperation, where each participating country offers resources to support the given crisis. These situations can occur suddenly, simultaneously, and without warning. Frequently, participants are coalition partners in one crisis and adversaries in another, raising difficult security issues concerning information sharing."[1]

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

1. Shared situational awareness,
2. Collaboration (e.g., operational planning and intelligence),
3. Coordination, and
4. Command-and-Control.

Operational users tend to emphasize the need to share and maximize the volume, variety, and quality of information that is discoverable and accessible by authorized users and partners. They recognize that information is vital to the formation, quality, and timeliness of decisions and 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 to ensure that only users with the appropriate credentials, authorizations, and needs can access designated information elements. The enforcement of need-to-know practices results in developing and deploying 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 and isolated regarding 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.

Security incidents like those listed below illustrate the limitations of conventional access control solutions and their inability to control and protect critical information assets sufficiently. They do not apply policies/rules to the content of individual information elements or provide Defense-in-depth, i.e., layering safeguards based on the value of critical data elements (e.g., security and privacy tags) within the information element. The following incidents illustrate current limitations:

1. 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 department information stores as hardened silos was 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 could use unrestricted and uncontrolled access to information to disclose significant amounts of sensitive data.[2]
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]

---

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

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

`

3. SLt. Delisle has admitted that selling secret information to the Russians over a 4 ½-year period jeopardizes Canada's ability to protect itself and 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]

Whether addressing insider threats (above) or the growing risks and costs associated with data breaches5,6 Decision-makers worldwide are seeking new and innovative ways to balance the requirement to share information and simultaneously protect sensitive data assets.

As enterprises increasingly digitize their data and information assets and rehost workloads to as-a-service infrastructures, they must pay greater attention to providing enhanced protection for their data (including enterprise intellectual property and knowledge). In this new world, many organizations will be outsourcing critical areas of their cyber security control, including:

1. Physical Security,
2. Personnel Security,
3. Network Security,
4. Platform Security, and
5. Application Security (if SaaS is adopted).

The enterprise often seeks security management from external agencies with limited ability to monitor and audit data access and use.

1.3 New Paradigm

While the two priorities—sharing and safeguarding—are often viewed as mutually exclusive, they are mutually reinforcing. Information systems that strengthen protection and the fidelity of security 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:

1. Increase operational effectiveness,
2. Improve information-sharing capability,
3. Increase information safeguarding capability,
4. Increase the availability, deployment, and repurposing of common/shared services and infrastructure,
5. Reduce operational costs,
6. Reduce management costs and
7. Reduce acquisition costs.

The recent shift from "need-to-know" to "requirement-to-share" introduces increased risk to information and data management environments. An increasing number of users and partners must be authorized to access security enclaves. Once a participant is authorized, they can access a wide range of data and information elements. Conventional access control solutions do not provide the fidelity and flexibility needed to enforce policy /rules and constraints to the data level. A data-centric approach seeks to enforce defense-in-depth to the data layer based on the sensitivity of the data content (Figure 1).

4 http://www.theglobeandmail.com/news/national/convicted-spy-delisle-sold-csis-names-to-russians-court-told/article8030374/

5https://www.forbes.com/sites/chuckbrooks/2021/10/24/more-alarming-cybersecurity-stats-for-2021-/?sh=7271b7184a36

6 https://www.fortinet.com/resources/cyberglossary/cybersecurity-statistics

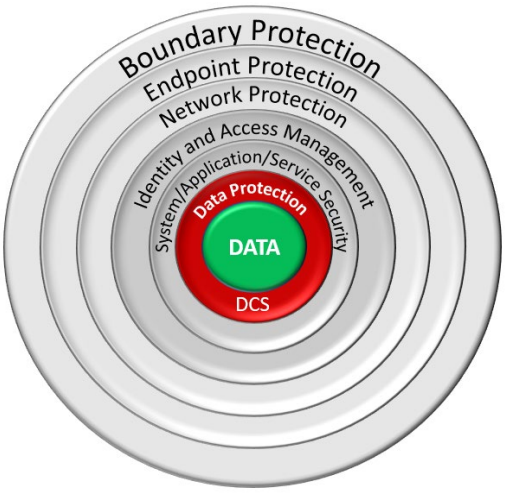


Figure 1 - Defense-in-depth for Data Centricity

As illustrated, Data-Centric Security (DCS) adds a layer of protection tailored to safeguarding data elements based on their sensitivity and the authorizations of individual users. The Information Exchange Framework Reference Architecture describes the service pattern and communications underpinning this data protection layer. DCS does not replace or eliminate any cyber security services and controls integrated into the environment.

1.4 IEF Approach

The IEF defines a framework for delivering Defense-in-depth solutions that address a wide range of information-sharing and safeguarding requirements. The IEF is an evolving set of specifications describing the requirements for:

1. A Reference Architecture (this document),
2. Formal Information Sharing and Safeguarding Policy Vocabularies,
3. Data-Centric Policy Decision Points (PDP),
4. Data-Centric Policy Enforcement Points (PEP),
5. Policy Administration Points (PAP),
6. Policy-based Packaging and Processing Services (PPS) and
7. Policy Development, Management, and Dissemination Tools (see PAP).

These specifications will enable users, vendors, and integrators to develop data and information management and protection measures that target responsible data and information sharing. IEF services will enforce ISS policies that govern what data and information content users can access, use, share, and modify. The IEF Reference Architecture defines an integration layer for open standards, off-the-shelf products, and services to enforce ISS policy at the data level rather than the networks, platforms, systems, and applications.

1.5 IEF Delivered Capabilities

The IEF seeks to define 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 set of missions and operational requirements. The IEF will align and integrate existing methodology, tools, technologies, and protocol standards and specifications where practical. Most importantly, 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 decisions and enforcement points) used to enforce them. This separation will enable users to:

1. Evolve ISS policy/rules independent of services and infrastructure,
2. Re-host ISS policies to multiple operating environments,

`

3. Activate ISS policies that respond to changes in operational requirements,
4. 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,
5. Adapt to rapid changes in operational context:
    a. Changing mandates, roles, and responsibilities:
    b. Changing mission and operational context,
    c. Evolving threats and risks,
    d. Evolving institutional policy,
    e. Advancements in technology and
    f. Control development and lifecycle costs.

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

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

Integrating ISS policy development into architecture will promote the retention of institutional knowledge and an overall reduction in lifecycle costs.

1.6 IEF Objectives

The IEF will provide people, processes, and systems with the ability and capacity to work together efficiently and effectively to ensure quality information is available to the right user (/decision maker) at the right location and time. The following table outlines the objectives for the IEF-RA.

Table 1 - IEF Objectives

Objective	Description
Policy-driven	Provides traceability from information sharing and safeguarding policy to the rules and constraints executed during operations to auditable logs used to govern data usage.
Data-centric	Provides security controls governing access, use, release, and modification based on policies tailored to data and information sensitivity
Dynamic Interoperability	Provides data owners and custodians with the ability to tailor policies to address changing user needs and operational conditions
Time-aware	Provides real-time ISS policy enforcement to enable users and systems to respond to changes in operational context (e.g., threat, risk, roles and responsibilities, and access rights)
Flexibility, Adaptability, and Agility	Provides rapid reuse and repurposing of policy for planned and spontaneous operations and enables run-time administration and management of policy environments

Table 1 - IEF Objectives	
Objective	Description
Architecture Alignment	Provides architectural views and viewpoints that enable users to integrate DCS /IEF elements (e.g., policies and services) into their enterprise, system, and service architecture Provides techniques and tools that facilitate the translation of enterprise security and data policy into machine-readable and enforceable rules and constraints
Integration Overlay	Provides services that interoperate with existing systems and infrastructure
Defense-in-Depth	Provides an additional layer of security services and controls that enforce ISS policies at the data layer
Self-Defending	Provides internal components that safeguard DCS data (e.g., policies, configurations, instructions, and logs)
Vendor Agnostic	Provide specifications for users, vendors, and integrators to implement and integrate their products and services.
Reuse of Existing Standards and Specifications	Provides specifications and standards for the DCS/IEF components
Responsible Information Sharing	Provides data and integration patterns to enable users to implement solutions that balance user requirements to maximize the discovery and access to authorized users while protecting data assets from unauthorized access, release, appropriation, and manipulation
Information Advantage	Provide decision-makers with the highest quality of data and information available. The information advantage has two facets: 1. Providing high-quality information to decision-makers faster than an adversary or competitor, and 2. Denying one's information and knowledge to one's adversaries or competitors.
Data-Centric Security	Provide the ability to protect data at-rest, in-use, and in-transit and have the data elements (/objects) carry their own data controls.

1.7 IEF RA Assumptions

The following assumptions govern the development of this reference architecture.

1.7.1 Data as a Strategic Asset

Data is a valuable and versatile strategic asset that must carry its protection independent of its location within the environment.

`

1.7.2 Interoperable vs Integrated Services

The IEF specifies a set of interoperable services that communicate using standard messages over a secure messaging or data bus (SMB). The choice to implement an interoperable or service-based specification:

1. Enable users, implementors, and integrators with the ability to implement a best-of-breed solution supported by multiple vendors,
2. Enable the addition of multiple instances of IEF services to address specific uses or challenges in the environment, including:
    a. The deployment of multiple PEPs, each addressing the requirements of a data exchange infrastructure (e.g., DDS, web services, or enterprise service bus) or exchange protocol,
    b. The deployment of multiple services to vertically and horizontally scale to address significant data volumes (e.g., data lake and Internet of Things (IoT)) and
    c. The deployment of multiple PPSs to address complex data domains employing multiple messaging and storage standards and
3. Directly integrate IEF services into a user's system or application.

This focus on an open service-based architecture does not preclude implementations of integrated IEF systems or appliances.

1.7.3 IEF Component Specifications

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

1. PDP and Access Control Policy Language: XACML v3 (or Higher) specification,
2. Information Exchange Packaging Policy Vocabulary (IEPPV) defines an ontology and UML profile for defining ISS policies and
3. ISMB: DDS and XMPP specifications form part of existing IEF implementations.

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

1. PPS: Requires a service specification for a component that ingests and enforces data policies conforming to the IEPPV and
2. PAP: Requires a specification that defines how component configurations and policies can be managed and administered by the User.

1.7.4 Error & Complex Conditions

The specification will not define responses to individual IEF error conditions. Individual responses to error conditions (e.g., User not authorized to access specified InformationElement, requisite Metadata or Cryptographic Keys are unavailable) will be left to individual implementations and user-specific requirements. The response to these conditions depends on user ISS, data, and security policies beyond the scope of the reference architecture.

The reference architecture provides component descriptions and sequence diagrams describing the core interactions between these components.

1.7.5 File (Exchange Data Object) Metadata

The reference architecture does not specify the supported user file (/object/message) formats and whether these formats include the requisite metadata element. Although metadata (sensitivity markings) is critical to the approach, how a user addresses the need is beyond the scope of architecture. The architecture defines a Secure Asset Container (SAC) and Trusted Data Object as core features for securely handling information assets.

The metadata and binding profiles are addressed in other specifications, e.g.:

1. ADatP-4774, NATO, Confidentiality Metadata Label Syntax - ADatP-4774 Edition A, 2017-12-20,
2. ADatP-4778, NATO, Metadata Binding Mechanism - ADatP-4778 Edition A, 2018-10-26, and

3. IC-TDF (Intelligence Community Trusted Data Format), XML Data Encoding Specification for Trusted Data Format, 2014, https://www.dni.gov/index.php/who-we-are/organizations/ic-cio/ic-cio-related-menus/ic-cio-related-links/ic-technical-specifications/trusted-data-format.

1.8 IEF RA Design Principles

The fundamental principles of the IEF and its support environment include:

1. IEF-RA services, policies, and configurations focus
    a. Protecting data and information elements based on content sensitivity and
    b. Improving data and information quality through user-defined and enforced policy.
2. The Information Sharing and Safeguarding Policy used by IEF-RA Services conforms to open standard based s policy vocabularies, e.g.:
    a. XACML,
    b. SAML,
    c. RuleML and
    d. IEPPV.
3. Support modeling language profiles (e.g., UML) that enable integration into standard architecture and architecture frameworks (e.g., UAF).
4. Support multiple domain implementations to facilitate interoperability with existing user infrastructure.
5. Automating the enforcement of user-defined information sharing and safeguarding at the data level.
6. Implementing defense-in-depth to the data level.
7. Delivering operational flexibility, adaptability, and agility.
8. Integrating IEF-RA services with existing or user-specified infrastructure and supporting services.

General Best Practices:

9. Reuse existing open standards where and whenever possible.
10. Vendor-neutral specifications.
11. Enable independent software, policy, and configuration lifecycles (strategies, practices, standards, and tools).
12. Enable rapid and continuous development, integration, testing, certification, and deployment of software and policy.
13. Enable Model-Driven Architecture (MDA) approaches.
14. Enable Model-Based Systems Engineering (MBSE) approaches.
15. Align and integrate ISS policy development with Enterprise Architecture (EA) practices.
16. Separate the definition of policies (/rules/instructions) from the services employed to enforce them.
17. Provide users with the architectural artifacts that support and enable governance, auditing (real-time monitoring and forensic), modeling and simulation, retention of institutional knowledge, and lifecycle management.

1.9 Adapting to change

1.9.1 Adapting During Operations

Each IEF component is governed by policy and configuration that are managed and administered using a Policy Administration Point (PAP) using the PAP-Command Messages (See Annex A). The PAP provides an interface enabling authorized users to manage or schedule policy and configuration changes to specific IEF components during operations.

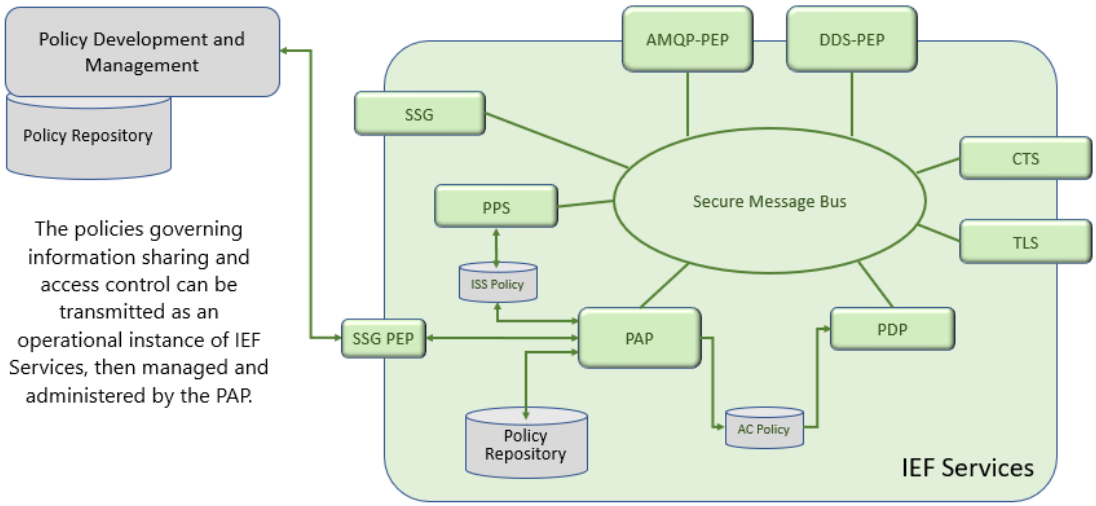


Figure 2 - Adapting to Change

The PAP can request policy sets and component configurations from the users' policy development or policy management environments. The PAP can request pre-developed policies or configurations from an authorized policy repository (/library) or enterprise organization authorized to develop and deploy policies (/rules and constraints) to operations. In addition, the PAP can allow an operational user to make selected changes to the policy environment of a node or set of nodes. Using these features, an authorized user(s) can adapt IEF operations to changes in the operational environment (e.g., threats, partners, and command intent).

IEF services can be rapidly deployed and configured for enterprise infrastructure, cloud (private, public, hybrid, and multi), or deployed directly to mission networks. The services operate as an application suite, containerized microservices, or virtualized to the cloud. Their policies and configurations can be tailored to specific User or mission needs.

1.9.2 Adaptation during Design

The IEF-RA is based on a (micro-) service-based architecture that enables each service to evolve independently, governed by a standard message-based Application Program Interface (API). This independence will enable implementors to extend, enhance, or replace services to meet changing operational needs. Service-based architecture is the foundation of agile and continuous development programs, enabling IEF-RA implementations to adapt the designs, implementations, and configurations to changing operational needs.

1.10 Adapting to evolving ISS and DCS trends

Several data and information security trends have occurred since the release of IEF-RA v1.0 in 2019. The following sub-clauses discuss the alignment or misalignment of the IEF-RA with these trends.

This version of the IEF-RA specification integrates features that enable the IEF to adapt to the lessons learned through various testing efforts and incorporate comments and issues raised by stakeholders and interested parties.

1.10.1 IEF Alignment with Data-Centric Security

Data-centric security encompasses the practices, policies, approaches, standards, tools, and technologies that focus on protecting data assets (content) throughout that data's lifecycle versus solely focusing on network and cyber security, which focus primarily on infrastructure (e.g., communications, networks, platforms, devices and information systems) risks.

The IEF initiative and the IEF-RA derive from and continue to track the efforts of the Western allies (e.g., NATO) and their evolving efforts to employ DCS to enhance interoperability and data protection to deliver information and decision advantage. To this end, the IEF-RA components have been implemented, tested, and evaluated at the Coalition Warrior Interoperability Exercise (CWIX), NATO's largest interoperability event for the exploration, experimentation, examination, and exercise of IM/IT solutions and standards.

The following table outlines IEF-RA's alignment with DCS's objectives.

Table 2 - IEF-RA – DCS Alignment		
#	Collection of Common DCS Objectives	IEF-RA Alignment
1	Standards-based binding and verification of trusted binding of metadata to data objects	All IEF PPS and PEP implementations have been tested using AdatP-4774 Confidentiality labels and AdatP-4778 binding for REST and SOAP. The IEF-RA is agnostic to the metadata and binding standards to be applied.
2	Standards-based guarding capability for data and information exchange based on the bound confidentiality labels	The PEP and PPS each provide guarding functions. The IEF PEP guard is based on the metadata values bound to the object or message. The PPS guard is based on the values in metadata and data objects. The user can author IEPPV policies to direct the PPS to generate metadata bound to the message or data object.
3	Standards-based application and management of metadata within the Enterprise	The IEF-RA services apply and use metadata following user-specified policy provisioned at runtime.
4	Availability of security accredited standards-compliant systems	Beyond the scope of the IEF-RA. Security Assessment and Authorization (SA&A) is a process the user performs to accredit an information system or service to operate. The IEF-RA services would have to go through this process for each user or community.
5	Metadata use within the enterprise to support the enterprise data and information management policy	The IEF-RA and its services implement data and information management policies by enforcing user-specified access control and ISS (or IEPPV) policies.
6	Distributed content inspection and control across the Communication Information System (CIS) domain	The PPS provides content inspection for structured and semi-structured data objects by enforcing ISS policy within the PPS. The other IEF-RA services (e.g., PEP/PDP) only inspect the metadata bound to an exchange data object.
7	Enforcement of object-level protection, including distributed enforcement of security and information management policies	The IEF-RA defines a policy-driven information-sharing environment derived from architecture and used to trace runtime policies (rules and constraints) to enterprise security and IM policies.

`

Table 2 - IEF-RA – DCS Alignment		
#	Collection of Common DCS Objectives	IEF-RA Alignment
		PPS policies are defined using the IEPPV UML profile, and ISS policies are aligned with enterprise architecture (e.g., UAF) views and viewpoints. The resulting architecture provides: • Traceability between interfaces, systems, processes, missions, and policy, • Retention of institutional memory and knowledge, and • Data that enables governance and security assessments and approvals (SA&A).
8	Enterprise and federated identity and access management mechanisms for use in support of object-level protection	An IEF-RA implementation can interface with Identity, Credential, Access Management (ICAM), and access control (e.g., ABAC and RBAC) services through the Security Services Gateway or PEP. Note: the access control services in the user environment can replace the PDP defined herein
9	User and terminal identity management within the Enterprise	The IEF-RA enables identity management by integrating with the users' infrastructure (e.g., Identity, Credential, and Access Management (ICAM)). This integration is performed through the PEPs and SSG.
10	Enforcement of object-level protection, including distributed enforcement of security and information management policies	The IEF-RA defines a set of services that move policy enforcement to the data access point. Pulling the controls from the perimeter back to the data provides protection distributed across the enterprise.
11	Metadata and security enforcement mechanisms across the enterprise	The IEF-RA defines a set of services that move policy enforcement to the data source. This means that every access and release decision is adjudicated at the source – enforcing object security at each IEF-protected data source.
12	Centralized and federated management of policy representations	Policy Administration Points manage operational ISS policies. PAPs can be operated centrally or federated across the environment.
13	Labeling of all data types, e.g.: • Email messages, • Chat messages, • Data objects, • Data messages, and	The IEF-RA services are responsible for the following: • Email messages → Email PEP, • Chat messages → Chat PEP, • Data files → File PEP,

Table 2 - IEF-RA – DCS Alignment		
#	Collection of Common DCS Objectives	IEF-RA Alignment
	• Data streams (e.g., videos, videoconferencing, and voice conversations).	• Data objects → Data PEP, • Data messages →Data PEP, and • Data streams (e.g., videos, videoconferencing, and voice conversations) →Data PEP. The PEP provides a core set of services that can be tailored to the specific data type and protocols.
14	Standard mechanism (e.g., PKI) to enable enterprise Confidentiality, Integrity, and Accessibility (CIA), e.g.: • Confidentiality (e.g., SSL/TLS encryption and S/MIME email encryption), • Data Integrity (e.g., Document/Code signing and S/MIME email signing), and • Authentication (e.g., Web page authentication, Machine and user authentication, and Two-factor authentication).	PEPs can be implemented or configured to any security service in the users' environment.
15	Application-layer cryptographic technology for key management, encryption, and digital signing of classified data	PEPs and SSGs can be implemented or configured to any security service in the users' environment.
16	Use of cryptographic mechanisms and critical management.	PEPs can be implemented or configured to any security service in the users' environment.
17	Use of object-level encryption for transport and storage.	PEPs can be implemented or configured to any security service in the users' environment. This capability depends on the types of data stores the user employs.
18	Tools and processes for management and validation of the correctness of security measures	Testing and analysis tools are currently beyond the scope of the IEF-RA.
19	Application-layer encryption, signatures, and key management for sensitive data	PEPs can be implemented or configured to any security service in the users' environment.

1.10.2 IEF Alignment with Zero-Trust Architecture

Figure 3 Illustrates the allocation of IEF services within a Zero Trust Architecture (ZTA). ZTA is an enterprise cybersecurity architecture based on zero-trust principles and design. It is intended to prevent data breaches and limit lateral movement within the environment. ZTA is a security model that eliminates the notion of trusting users within protected enclaves, networks, and systems. With Zero Trust, all users are presumed untrustworthy.

This approach contrasts with the traditional perimeter security model, which presupposes that bad actors are always on the network's untrusted side (outside) and trustworthy users are always on the trusted side (inside).

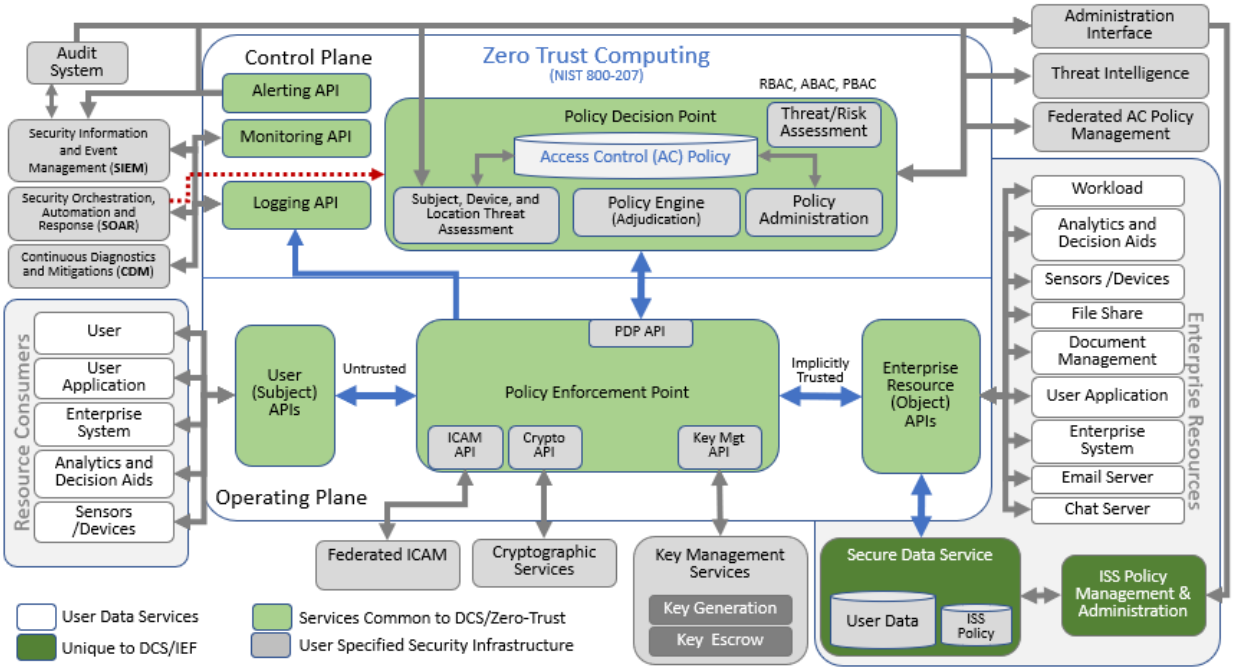


Figure 3 - IEF in Zero Trust Architecture

The following table outlines IEF-RA alignment to ZTA Tennents.

Table 3 - IEF-RA - Zero-Trust Alignment		
#	Zero Trust Tennent (NIST 200-207)	IEF-RA Alignment
1	All data sources and computing services are considered resources	The IEF-RA seeks to treat data as resources independent of the systems, applications, and services that operate on that data
2	All communication is secured regardless of network location	IEF-RA services can encrypt all communications between IEF-RA components and systems, applications, and devices seeking to access their protected data
3	Access to individual enterprise resources is granted on a per-session basis.	IEF-RA services can protect data and metadata at the attribute level if the underlying data structures, protocols, and technologies enable attribute-level control.
4	Access to resources is determined by dynamic policy, which includes the observable state of client identity, application/service, and the requesting asset	IEF-RA PEPs are controlled by a PDP adjudicating user-defined policies

Table 3 - IEF-RA - Zero-Trust Alignment		
#	Zero Trust Tennent (NIST 200-207)	IEF-RA Alignment
	and may include other behavioral and environmental attributes.	The PPS is controlled by user-specified exchange and semantic policies governing the processing, storage, and packaging of all data for release
5	The enterprise monitors and measures the integrity and security posture of all owned and associated assets.	The IEF-RA services log all transactions to the level defined in the user-specified component configuration. Users can route these log records to monitoring and auditing services.
6	All resource authentication and authorization are dynamic and strictly enforced before access is authorized.	The IEF strictly enforces authentication and authorization by the PEPs, adjudicated by the PDP. For structured data, the PPS and its bound metadata at the PEP adjudicate and enforce authentication and authorization to the attribute level.
7	The enterprise collects as much information as possible about the current state of assets, network infrastructure, and communications and uses it to improve its security posture.	The Enterprise specifies the configuration of each IEF-RA component The PAP maintains status for each IEF-RA component under its control (note that the PAP is a message-based service that can be integrated into the user's security and system management services) The TLS interface can be integrated into the user's security information, event management (SIEM) system, or logging service

DCS specifically enforces Zero-Trust (least access) for data assets and communications, including:

1. Chat messages,
2. Email messages,
3. Shared files and
4. System-to-System message exchanges.

1.10.3 IEF Alignment with Cloud Security Practices

The implementation of zero-trust computing within an enterprise network coincides with the organization's desire to increase its control of access to resources (e.g., systems, applications, devices, and data) for external and internal users. Zero-trust computing establishes boundaries at the resource's access point rather than the network's boundaries. Integrating controls for each resource and asset, tailored to their unique sensitivities, better balances access controls and unintended lateral movement within the network.

Increasingly, organizations seek to leverage the benefits of cloud (private, public, and hybrid) computing. Many of these benefits derive from multi-tenant infrastructures, platforms, and software. The sharing and reuse of these resources provide cost benefits that dedicated data centers cannot deliver. However, it comes at the cost of relinquishing physical, personnel, and network (/boundary) security to the Cloud service provider (e.g., the shared responsibility model). From the user's perspective, traditional security does not apply in the cloud.

`

Organizations that have moved workloads to the cloud have systems, applications, and data spread across multiple networks, infrastructures, and locations. They are losing insight into:

Where workloads are executed:

1. Who is accessing their workloads, applications, and data,
2. What devices (e.g., smartphones, tablets, and laptops) access workload, systems and data, and
3. How are data assets being used and shared?

The IEF explicitly treats data as a resource – a pillar of ZTA. As illustrated in Figure 4, the IEF

As illustrated (Figure 4), the IEF-RA separates the data from all users (e.g., individuals, applications, systems, and devices) from the data assets. It provides the architecture for integrating services that provide defense-in-depth and zero-trust computing for data and information assets and sharing and safeguarding them per user-defined policy (rules and constraints). It provides the logging required to regain institutional knowledge and understand how data assets are accessed, used, and shared. As described in the previous clause, the IEF directly aligns with the core tenets of zero trust.

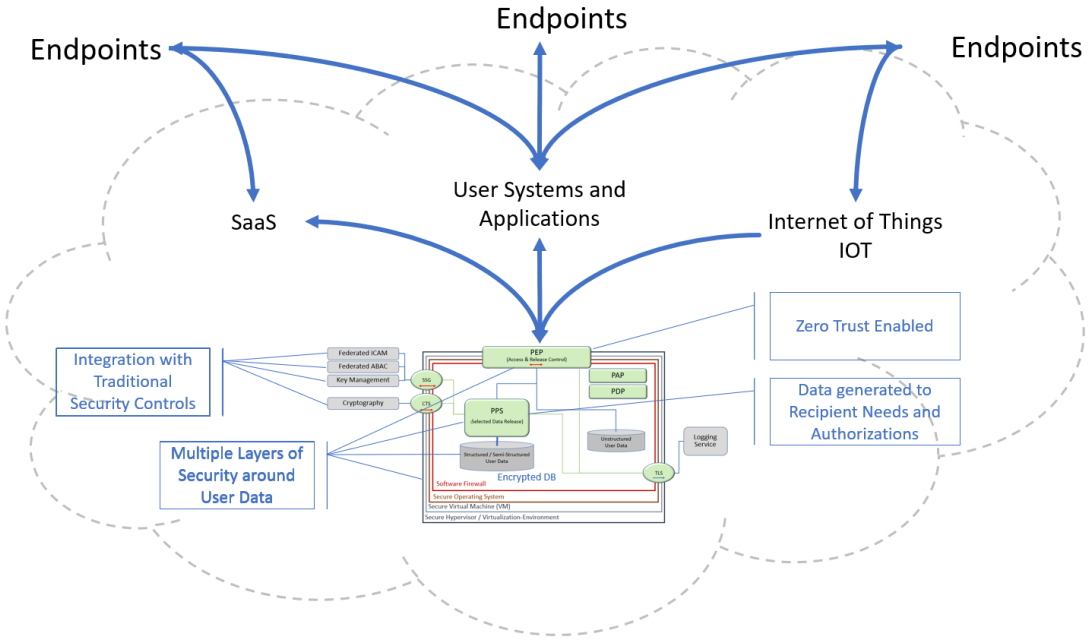


Figure 4 - IEF in the Cloud – All interaction with Data Resources Protected by a PEP

1.10.4 IEF Alignment to Data Centricity

Transforming organizations into data-centric operations is critical to improving performance and creating information to decision advantage at all enterprise levels, from operations to the board room, ensuring competitive advantage. To accelerate the digital transformation efforts, leaders must ensure all data is visible, accessible, understandable, linked, trustworthy, interoperable, and secure. Delivering data centricity involves:

1. Maximizing data sharing and safeguarding for data use: all data is an enterprise resource,
2. Publishing data assets using federated data catalogs employing standard interface specifications,

3. Deploying automated data interfaces that are externally accessible and machine-readable, ensure interfaces use industry-standard, non-proprietary, preferably open-source technologies, protocols, and payloads,
4. Storing data in a platform- and environment-agnostic manner, uncoupled from hardware or software dependencies and
5. Employing industry best-practices for secure authentication, access management, encryption, monitoring, and data protection at rest, in-transit, and in-use.

Employing the IEF Secure Data Service Configuration (Section 1.12.6), the IEF enables users to deploy data as a service that is:

1. Using a PPS that employs policy-driven data sharing and safeguarding:
2. Populating (automatically) data catalogs per data sharing and safeguarding policy,
3. Employing standards-based interfaces, protocols, and payloads,
4. Employing secure data services agnostic of its environment, uncoupled hardware, software, and operational dependencies and
5. Using PEPs to employ secure authentication, access management, encryption, monitoring, and data protection at rest, in-transit, and in-use.

Figure 5 Illustrates a high-level data-centric architecture employing IEF Secure Data Services.

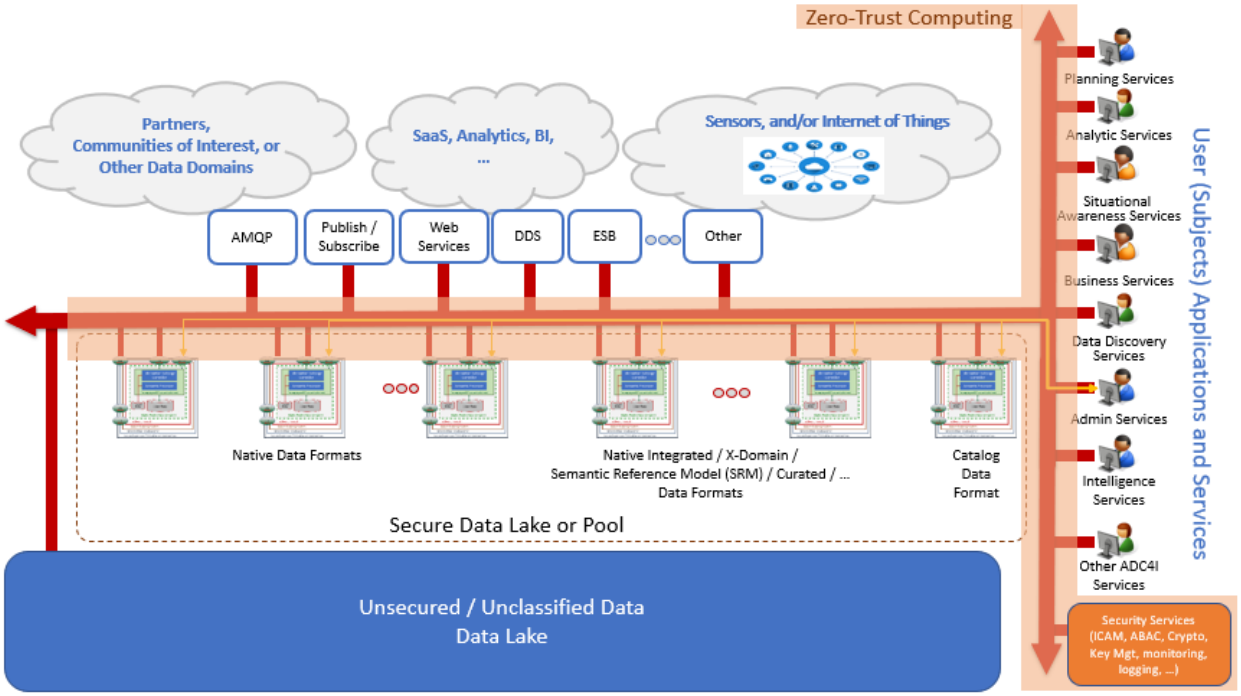


Figure 5 - Data-Centric Architecture

1.10.5 IEF in Data Centricity, Digitization, and Digital Transformation

The current focus of most public and private sector organizations is the ability to exploit all source or pan-domain data to develop and sustain an information advantage over competitors and adversaries. Developing this capability is not a destination but a journey. The following outlines several steps in this journey:

1. **Siloed Enterprise**: Represents the starting state for many organizations:
    a. Enterprise data, information systems, and workloads are siloed or enclaved and unable to communicate their data to decision-makers who need it,
    b. Slow to identify and react to challenges and changes (e.g., threats, risks, technical, business/operations, partners, and adversaries) in the environment,

`

c. Lacking in flexibility, agility, and adaptability to take advantage of opportunities in the environment and
d. Vulnerable to advances and innovation in competitor's or adversary's capability.
e. Struggling with aging technologies and capabilities that were never designed for change.

2. **Digitization**: Represents the transformation of data holdings into a digital form:
    a. To improve access to enterprise data and
    b. To lower the risk and costs of information and data management programs.
3. **Digitalization**: Represents the transformation of enterprise policy, practices, processes, standards, tools, and infrastructure to exiting operations through the exploitation of enterprise data:
    a. To enable the efficient and effective use of enterprise data,
    b. To protect sensitive (private, confidential, legally significant, and classified) data governed by legislation, regulation, international agreements, security, and data policy,
    c. To improve enterprise data quality,
    d. To improve information quality for decision-makers,
    e. To improve trust in enterprise data and information,
    f. To exploit innovation in analytics, artificial intelligence, machine learning, business intelligence, and technology,
    g. To develop single sources of truth for enterprise data and information,
    h. To improve information and data management, Quality of Service (QoS) and
    i. To lower operating and lifecycle costs.
4. **Digital Transformation**: Deliver new value to strategic, operational, and tactical organizations, including:
    a. Generate new capability,
    b. Nimble and able to pivot,
    c. Create blue ocean markets,
    d. Fend off innovative competitors and
    e. Expand ecosystem.
5. **Data-Centric Enterprise**: Maximize the utility of enterprise data in the delivery of outcomes and desired effects, including:
    a. To develop and sustain information and decision advantage:
        i. Maximize the availability of quality information to decision-makers and
        ii. Protect sensitive data and information assets and workloads from unauthorized access, release, use, appropriation, and manipulation:
    b. To develop new and enhanced capabilities from all-source and pan-domain data:
    c. To scale the transformed enterprise to meet the new and evolving mission and operating needs,
    d. To leapfrog competitor and adversary capabilities,
    e. To accelerate the achievement of objectives and outcomes,
    f. To integrate capability and innovation at mission speed and maximize its benefit to the enterprise and
    g. To future-proof the enterprise.

1.10.5.1 Role In Digitalization

The IEF can be used to expedite the digitalization process by providing secure receptacles in which to place the newly digitized data element. Because the policies (rules and constraints) governing access to data are a separate and independent process, the IEF approach enables an evolutionary approach to digitization and digitalization activities and later digital transformation. The IEF allows the enterprise to implement an ISS and DCS infrastructure for dynamic and continual change.

As illustrated in Figure 6, the IEF approach separates policy development from software development. This separation adds several benefits:

1. It enables the enterprise to place the responsibility and authority for developing, testing, and deploying ISS and DCS policy to the IM and DM organizations and governance structure,

2. It enables IT organizations to implement, test, and deploy ISS and DCS infrastructure without waiting for the business to define and implement ISS and DCS policies,
3. It enables the business to evolve ISS and DCS capability over time,
4. It provides for an agile, flexible, and adaptive enterprise,
5. It eliminates the large, complex Information System project that rarely meets enterprise needs and project budget and schedule, and
6. It places the enterprise on the path to exploiting agile DevSecOps practices, processes, standards, and tools for software and policy as part of separate lifecycles (Figure 6).

The IEF operates on existing digital data. Its use and benefits begin after the data are digitized. User data is digitized and iteratively integrated into the secure data lake (see Figure 16).

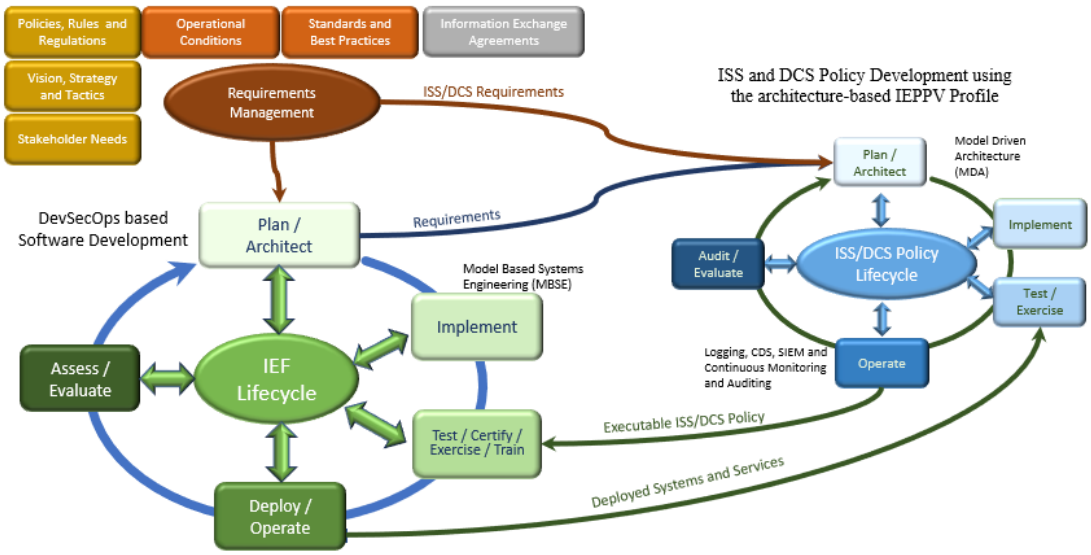
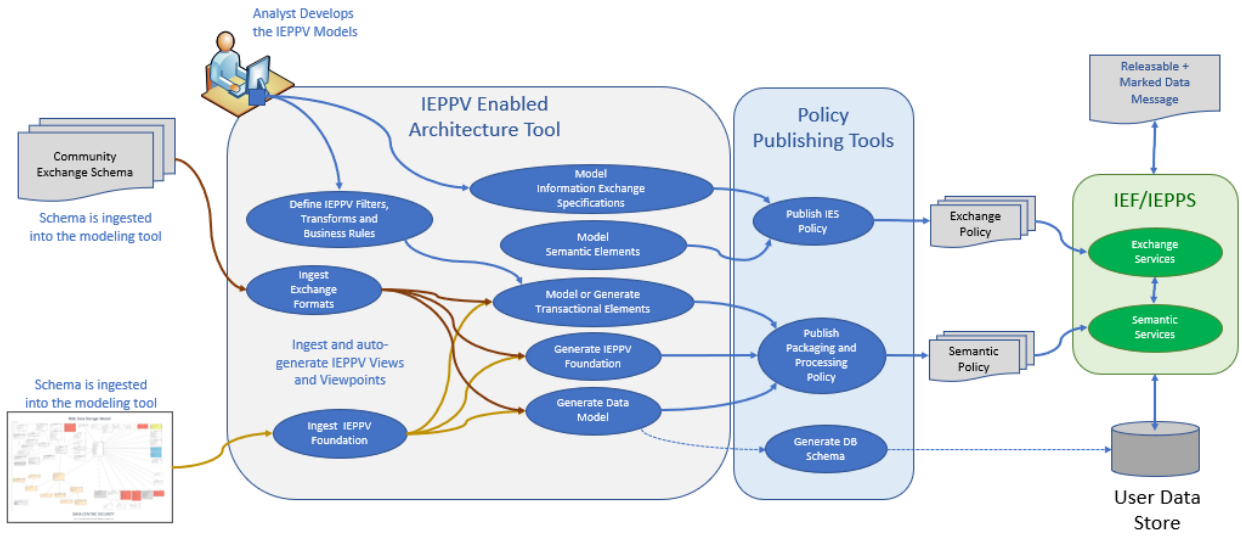


Figure 6 - Separation of Policy and Software Lifecycles

The IEF aligns architecture-based approaches to modeling and autogenerating runtime policy environments (Figure 15). Model-driven architecture (MDA) and Model-Based Systems Engineering (MBSE) practices and tools can dramatically shorten the time required to deliver ISS and DCS capability to operations.



`

Figure 7 - Architecture to Operations

1.10.5.2 Digital Transformation

Digital transformation extends digitalization into new domains of operation and opportunity. Where digitalization seeks to use enterprise data to improve existing practices, procedures, processes, and decisions, digital transformation seeks new uses for enterprise data and new sources of data (e.g., the Internet of Things). As with digitalization, the IEF will enable the enterprise to rapidly explore and exploit new and existing data sources to achieve the following:

1. New data capabilities in the strategic, operational, and tactical domains,
2. Increased decision flexibility, agility, and adaptability,
3. Increase the speed of the mission and
4. Increased information and decision advantage.

1.10.5.3 Role in a Data-Centric Enterprise

As the enterprise moves from application-centric to data-centric operations, the IEF will enable:

1. The continued separation of concerns between the business (information Requirements), data management (data policy), and technology organizations,
2. The continued evolution of data-centric capabilities and
3. The continued improvement of data access, quality, and protection.

1.10.6 IEF Alignment with Secure Relationship Protocol

Secure Relationship Protocol (SRP) is a development initiative of the Industrial IoT Consortium (IIC7 - https://www.iiconsortium.org/) – Secure Communications and Infrastructure by Design (SCID). The initiative provides an architecture to resolve security, communication, and management issues common to most communication and information management architectures. The SRP then extends policy-based security controls into the transport and networking layers of the architecture. As illustrated in Figure 5, the IEF components operate in the data services layer and sit on top of the exchange services, which may provide policy-based security services (e.g., DDS). In combination, IEF, DDS, and SRP provide user-controlled policy-driven security controls for all data and information-sharing aspects.

7 The Industry IoT Consortium is a program of the Object Management Group®, Inc. (OMG®), a not-for-profit 501(c)(6) tax-exempt organization.

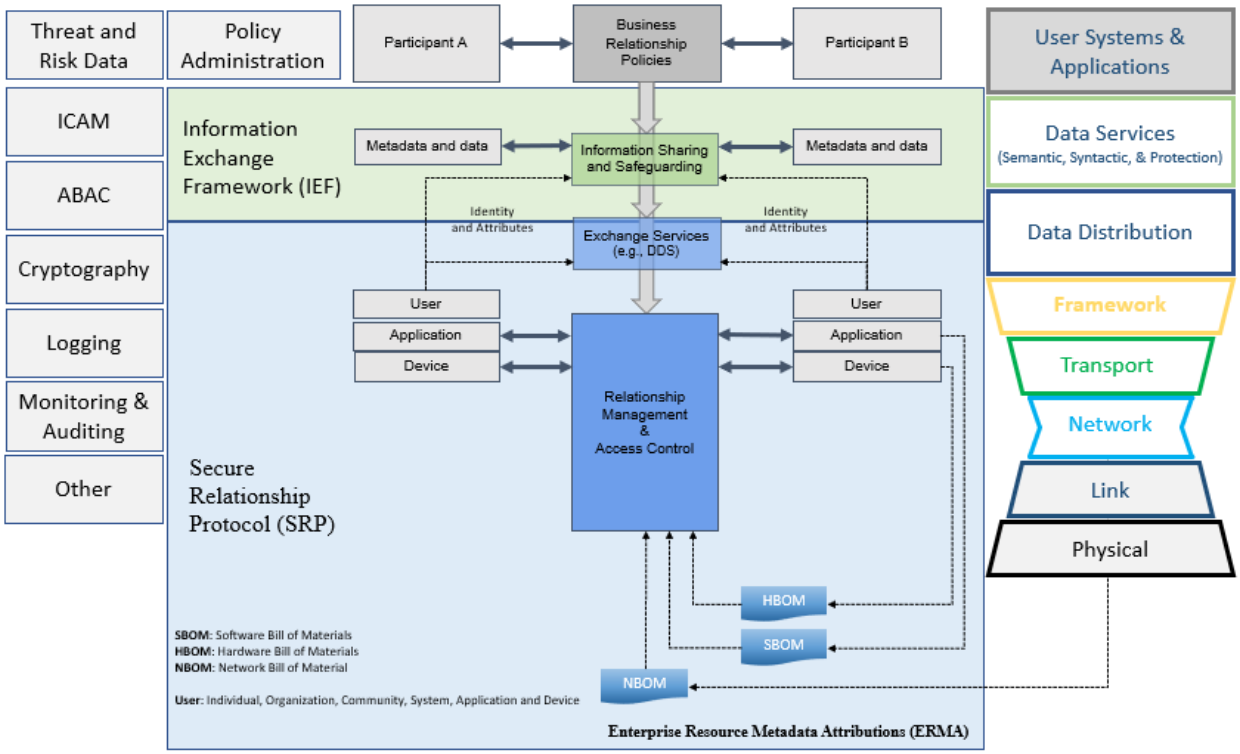


Figure 8 - IEF Alignment to SRP

The Bills of Materials (BOM)8 Describe the hardware, software, and network needed to authenticate and adjudicate communications across the network.

1.11 Alternate Configurations of IEF Services

The following clauses outline alternate configurations of the IEF components.

1.11.1 Secure Data Service (SDS)

The Secure Data Service (SDS) integrates IEF-specified services within traditional Hypervisor, Operating System, and Virtual Firewall security controls (Figure 9). The objective is to provide layers of security around the data and data services by placing them within a virtual container and affixing the requisite layers of defense for the data based on its sensitivity.

Data-centric security adds layers of security around the data in a defense-in-depth configuration. As illustrated, the SDS adds the PPS and PEP, which enforce semantic exchange and access control policies for every data exchange.

The SDS completely wraps User data in IEF and traditional (virtual) security. The SSG, CTS, and TLS provide PEP-defined functionality when interfacing with user infrastructure components. In addition to providing integration points to the user's security infrastructure, the SSG, CTS, and TLS provide many of the PEP functions needed to protect the IEF components from unauthorized access.

8 Refer to the Enterprise Resource Metadata Attributions (ERMA) Request For Proposal (https://www.omg.org/cgi-bin/doc.cgi?ad/2008-9-15)

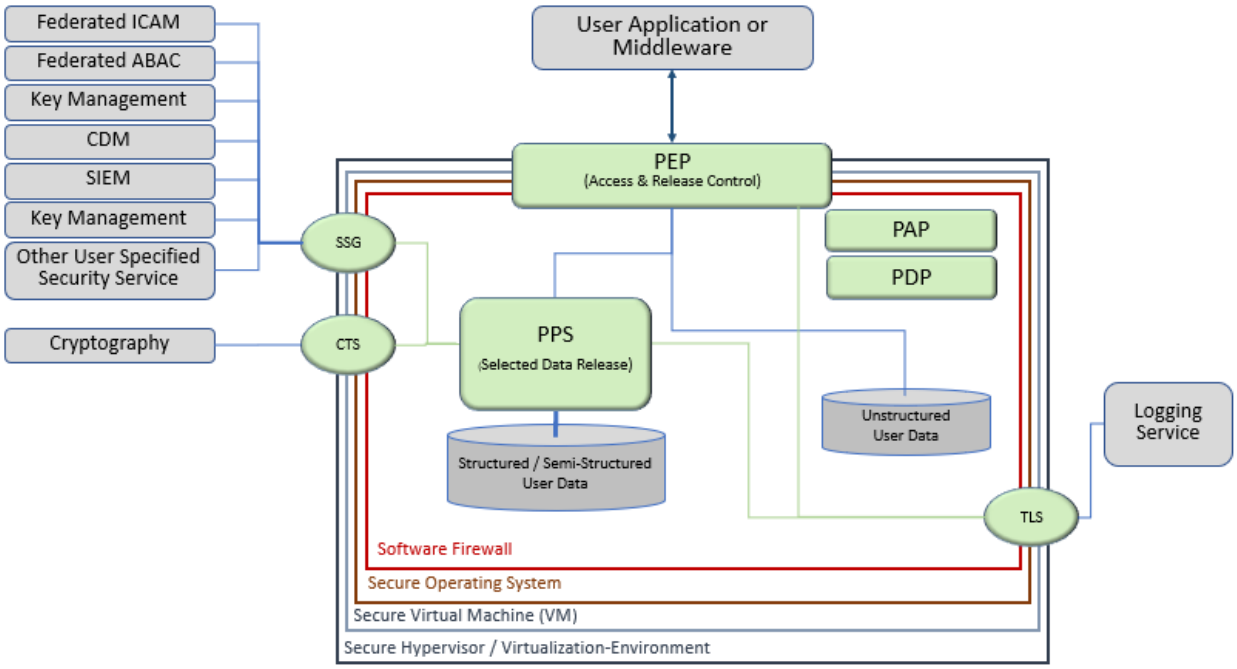


Figure 9 - Secure Data Service (SDS)

1.11.2 Direct PEP Access to Security Services

IEF testing of the configuration in Figure 7 uncovered that the IEF-RA design requiring each IEF service to access user security infrastructure through the SSG presented a significant overhead and latency. The PEP (Figure 8) was refactored to communicate directly with the requisite security services (e.g., ICAM, ABAC, key management, and cryptography). The refactoring reduced the latency identified in the original architecture and enabled the PEP to integrate with federate ICAM and ABAC servers.

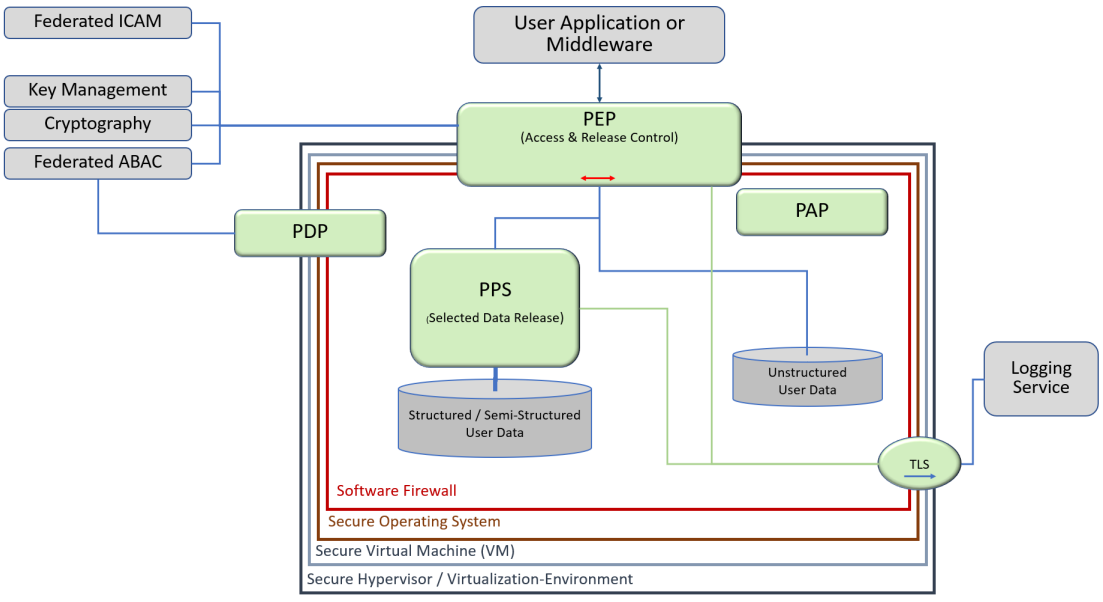


Figure 10 - Extended PEP

The latter configuration (Extended PEP) displayed some performance benefits over SSG and CTS configurations.

As illustrated in Figure 9, the PEP provides all the features, functions, and APIs required to deliver data-centric security in a zero-trust environment.

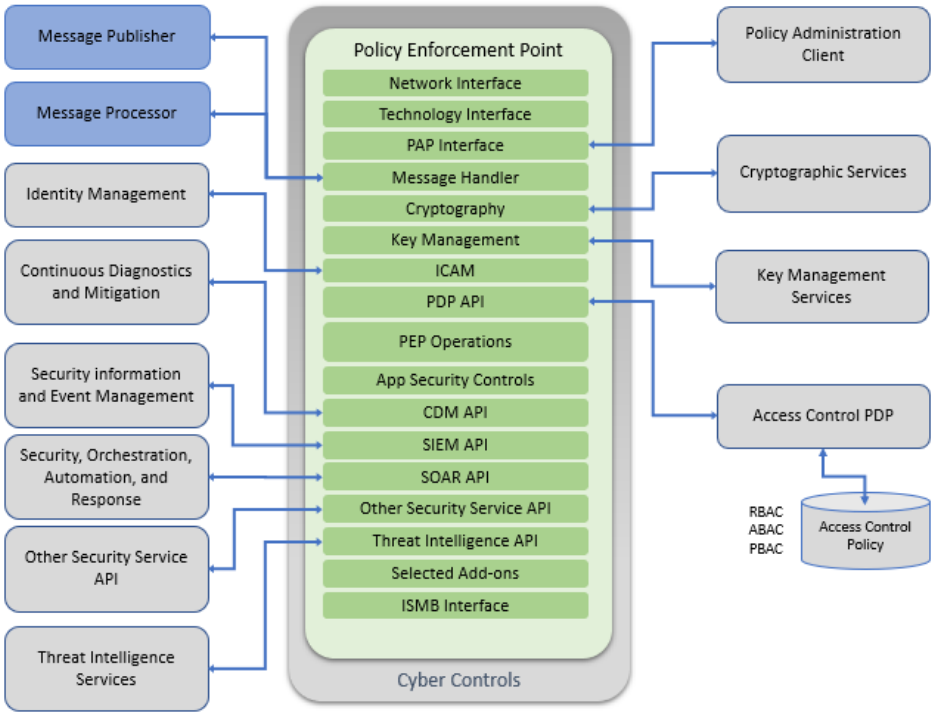


Figure 11 - PEP Extensions

A second configuration change used a separate PEP for each technology stack. This change enabled the team to support multiple user (/community) requirements to share data using a variety of protocols and technologies (e.g., SOAP, REST, and DDS), syntaxes, and communications links. It enabled the development of a set of core reusable PEP services while isolating the technology differences to the specific PEP. PEPs are included as necessary to support partner testing without adding complexity (processing differences within each technology combination) to an individual PEP implementation; user-selected variations of security services further exacerbate potential PEP complexity.

PEP features, functions, and requirements are provided in Clause 9.

1.11.3 Streaming Data Configuration

Not explicitly addressed in the IEF-RA V1.0 Specification, Figure 12 Illustrates a configuration of the IEF SDS that delivers DCS for full-motion video (FMV) streams. The IEF SDS splits the stream and selectively releases portions to recipients based on their needs and authorizations.

This DCS solution for FMV comprised configuration and policy changes to standard IEF components – showing the versatility of the IEF-RA. As illustrated, video streams were labeled and fed into a secure data service comprising a set of IEF-RA-defined components. The SDS stripped off the confidentiality labels and

`

used the metadata in the labels. The user's attributes (e.g., clearance, location, and data authorizations) to receive or not receive the video feed.

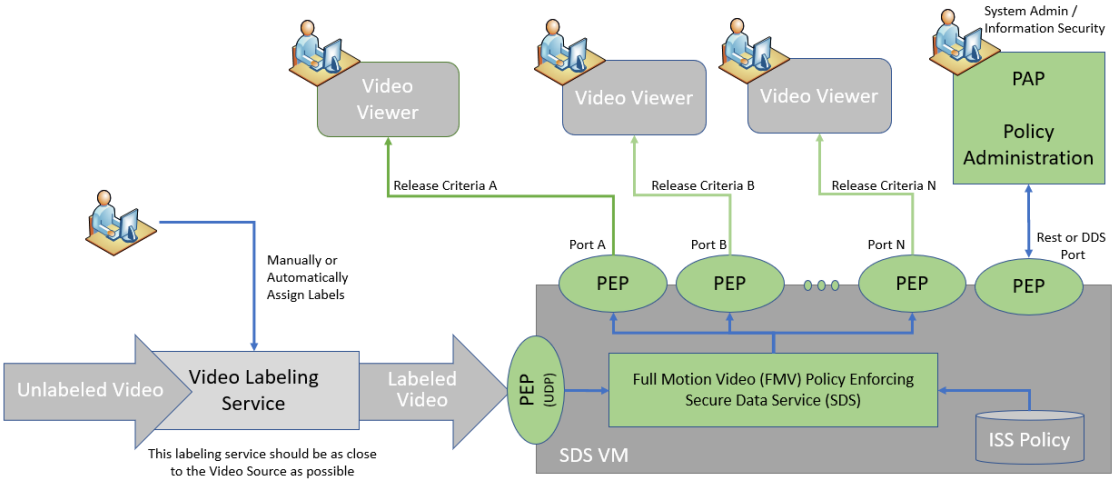


Figure 12 - IEF /DCS for Full Motion Video

1.11.4 External PAP

As illustrated in Figure 13, the Policy Administration Point (PAP) can be outside the SDS perimeter and integrated through a dedicated PEP. This variation enables the PAP functionality to be hosted by other user-developed system administration applications – a recommendation from several test partners.

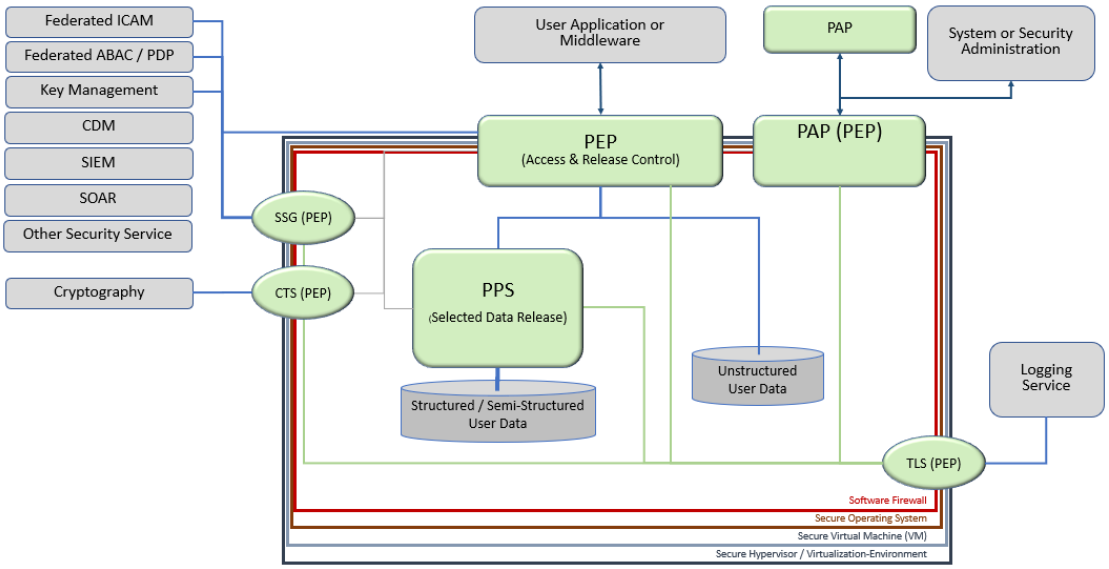


Figure 13 - External Administration Point

1.12 IEF Deployment Configurations

1.12.1 Micro-Service-Based PPS

Progressive development of the PPS identified the potential for increasing policy and configuration control levels and user control (Figure 14). As illustrated in the most recent test configurations:

1. The Policy Administration Point (PAP – Clause 8) is configured as an external service interacting with IEF services through a PEP and
2. Much of the policy and business logic is extracted from and managed separately from the enforcement services (e.g., PEP and PPS), including:
    a. Message Schema,
    b. Parser Libraries,
    c. Transformation Libraries,
    d. Business Logic (e.g., filters),
    e. Data Mappings,
    f. Semantic Policies,
    g. Exchange Policies and
    h. Publisher Libraries and
3. This configuration allowed greater reuse of the IEF services and maximized the users' ability to extend capabilities to address evolving business and data protection needs.

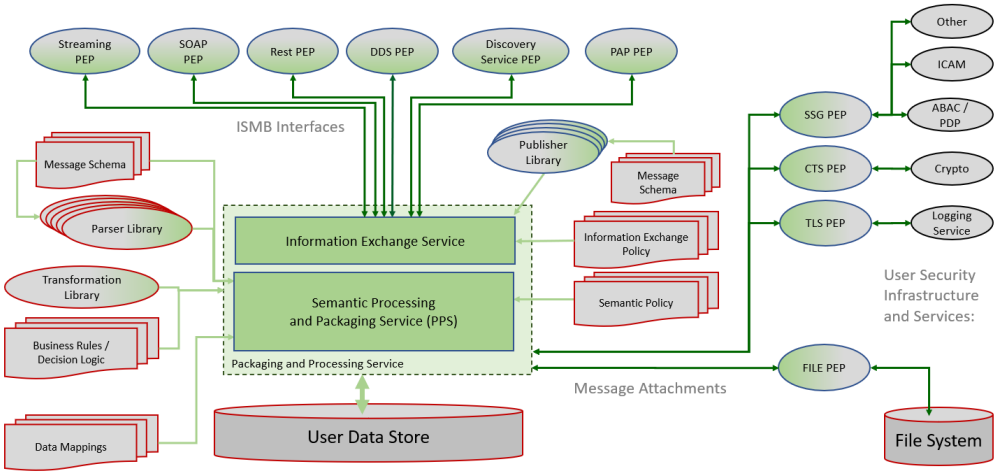


Figure 14 - Packaging and Processing Service Exemplar

1.12.2 Secure Data Service (SDS)

Figure 11 illustrates the wrapping of IEF and traditional virtual security services around a data store (e.g., RDBMS, Object Store, or File Store). The SDS (Figure 12) wraps IEF and traditional (virtual) security around the user data store to ensure recipients can only access data that they need (defined by approved Information Sharing Agreements) and are authorized (user information sharing and safeguarding [security] policy) to access. DCS controls are applied by:

1. The PEPs/PDP that enforce access control policy using labels bound to each data object received or released by the SDS and
2. The PPS that enforces:
    a. The exchange policy that governs the release of data and metadata content aggregated for formatting and release and
    b. The semantic policy governs the aggregation, transformation, labeling, and redaction (filtering) of available data.

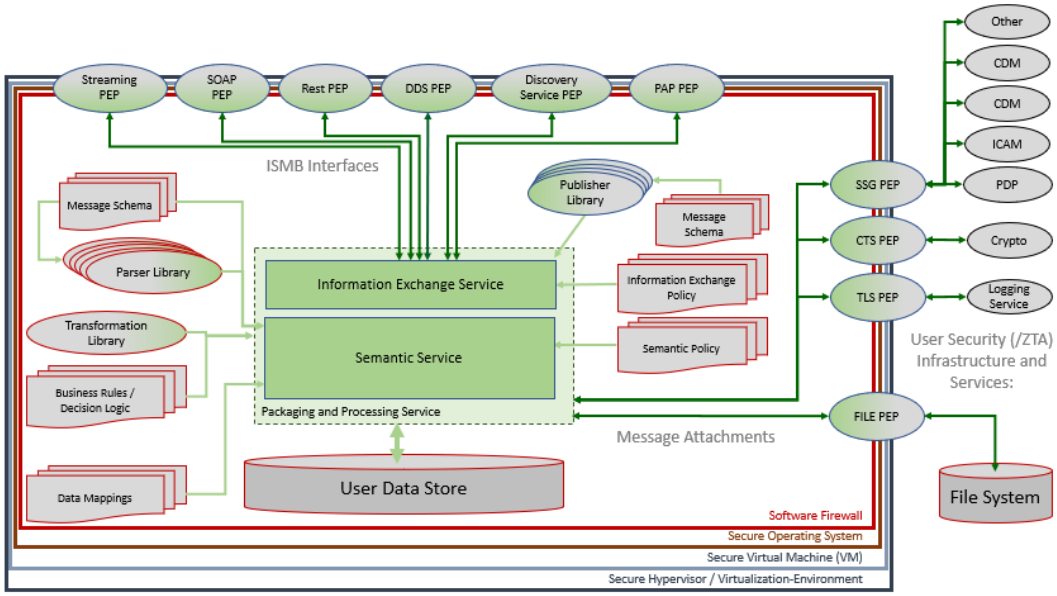


Figure 15 - Secure Data Services Exemplar

1.12.3 DCS Enable Data Lake Configuration

As illustrated in Figure 16, users can deploy multiple SDSs to the environment. In this example, individual SDSs are deployed to:

1. Capture and store data in its native schema,
2. Capture and store data post-curation or in a standard Semantic Reference Model (SRM) and
3. Store metadata in a catalog to facilitate data discovery.
4. Each SDS enforces its policy environment independently. These policies are tailored to protect data by the SDS.

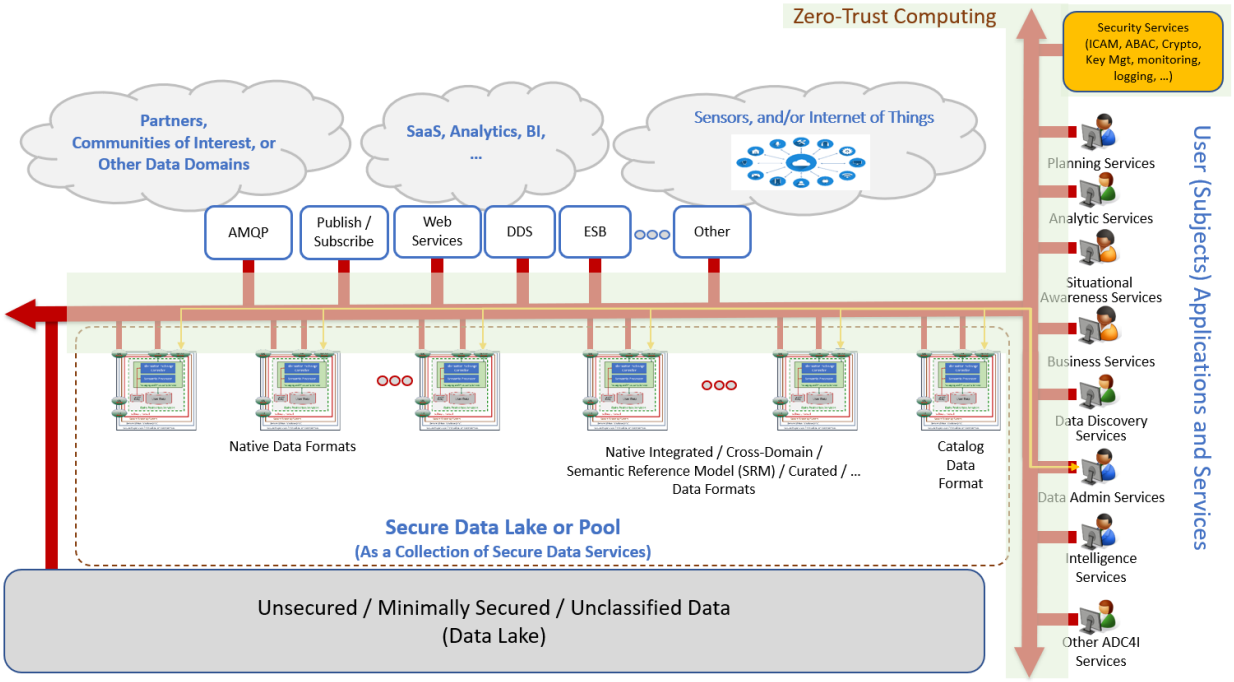


Figure 16 – DCS Enabled Data Lake

`

2 Conformance

The conformance clause identifies which specification clauses are mandatory (or conditionally mandatory) and which are optional for an implementation to claim conformance to the specification.

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

Table 4 - Compliance		
Email Compliance	Description	Cause
Email Compliance	This compliance point applies to implementors seeking to deliver data-centric security for email exchange.	2.2.1
File-Share Compliance	This compliance point applies to implementors seeking to deliver data-centric security for email exchange.	2.2.2
Instant Messaging Compliance	This compliance point applies to implementors seeking to deliver data-centric security for email exchange.	2.2.3
Data Messaging Compliance	This compliance point applies to implementors seeking to deliver data-centric security for email exchange.	2.2.4
Alternate Configurations	These compliance points describe alternate integrations of functions that deliver equivalent data sharing and safeguarding controls as compliance in 2.2.4 with increased use of traditional cyber and zero-trust controls.	2.3

The following causes describe Email, File-Share, Instant Messaging, and Data Messaging compliance.

2.1 Selecting a Compliance Point

This specification defines four compliance points. A user, vendor, or integrator may select and implement one or more compliance points described below.

2.2 Compliance Point Descriptions

For an implementation to demonstrate compliance, it must provide the functionality specified for at least one Policy Enforcement Point, the six supporting components, and APIs conforming to ISMB messages (Clause 16). The Policy Enforcement Points (PEP) include:

1. Email PEP (Clause 2.2.1),
2. File Share PEP (Clause 2.2.2),
3. Instant Messaging PEP (Clause 2.2.3) and
4. Messaging PEP and PPS (Clause 2.2.4 and 2.2.5, respectively).

Note: The user, implementor, or integrator is responsible for implementing the APIs needed to enable the PEP to interoperate with external security services (e.g., ICAM, Access Controls, Key Management, and Administration services). The PEP is an integration point between IEF services and the users' infrastructure.

The six supporting components for each configuration are:

1. Policy Administration Point (PAP, Clause 8),
2. Policy Decision Point (PDP, Clause 10**Error! Reference source not found.**),
3. Security Services Gateway (SSG, Clause 12),
4. Cryptographic Transformation Service (CTS, Clause 13),
5. Trusted Logging Services (TLS, Clause 14), and
6. Secure Message Bus (Clause 15).

Figure 17 Outlines the core function provided by each of the IEF components.

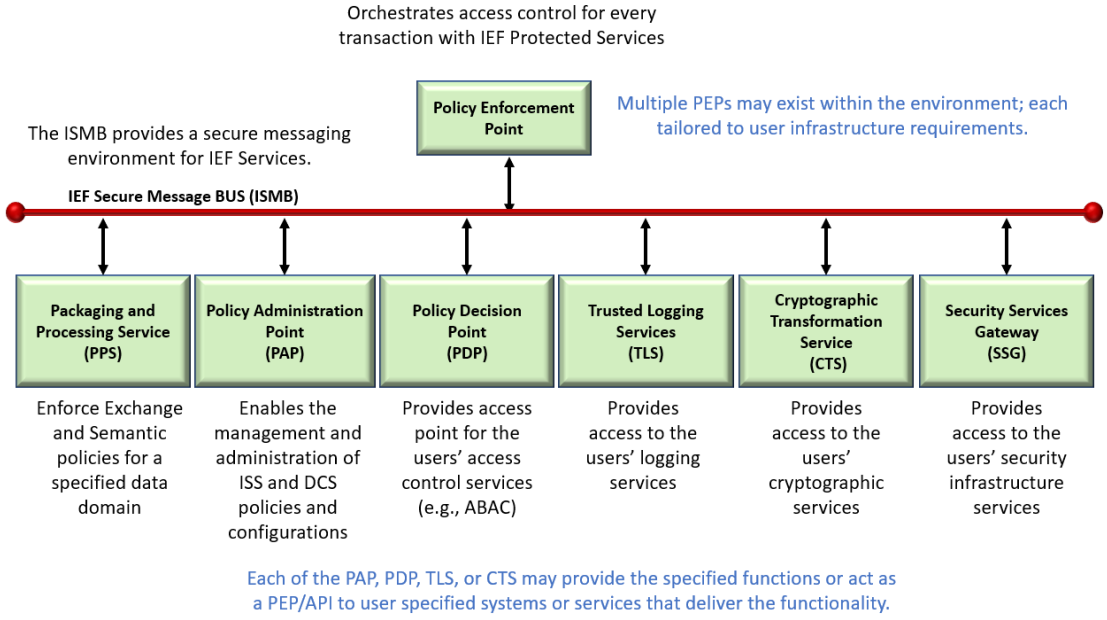


Figure 17 – IEF Components

Figure 18 Illustrates an IEF configuration where the implementor integrates the specified functions into the described application.

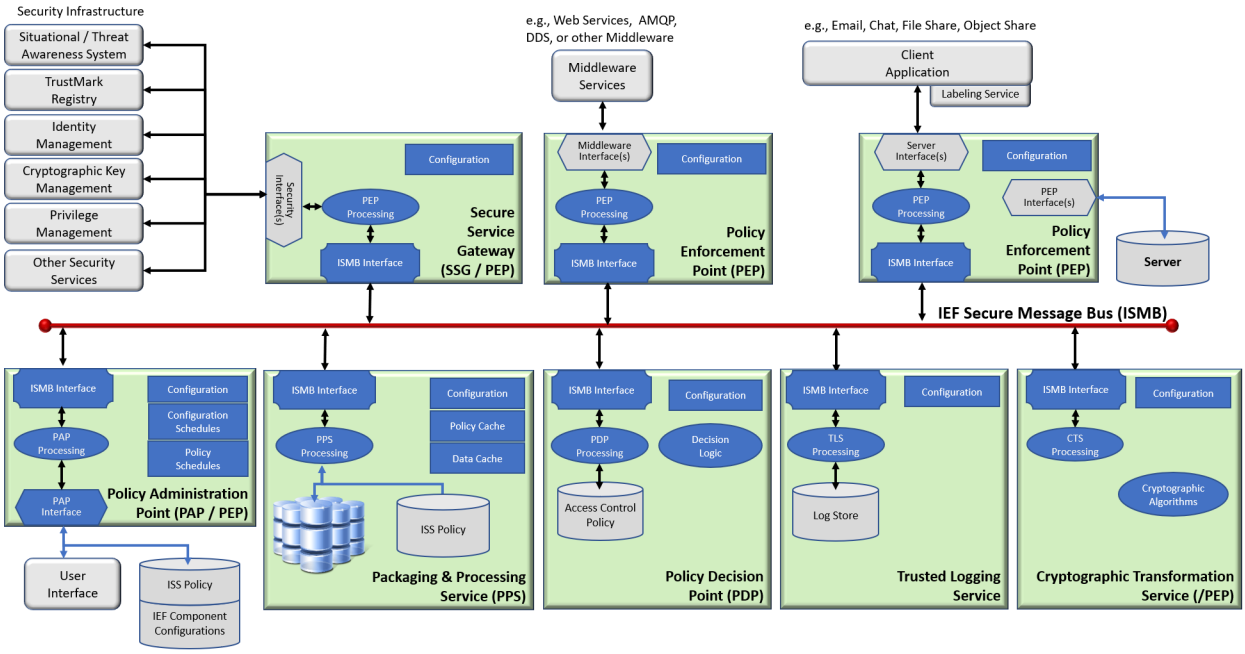


Figure 18 – IEF Service Topology

Alternatively, the implementor can implement the components (e.g., PAP, PDP, TLS, and CTS) as PEPs that provide the IEF-specified services using off-the-shelf or Software-as-a-Service components.

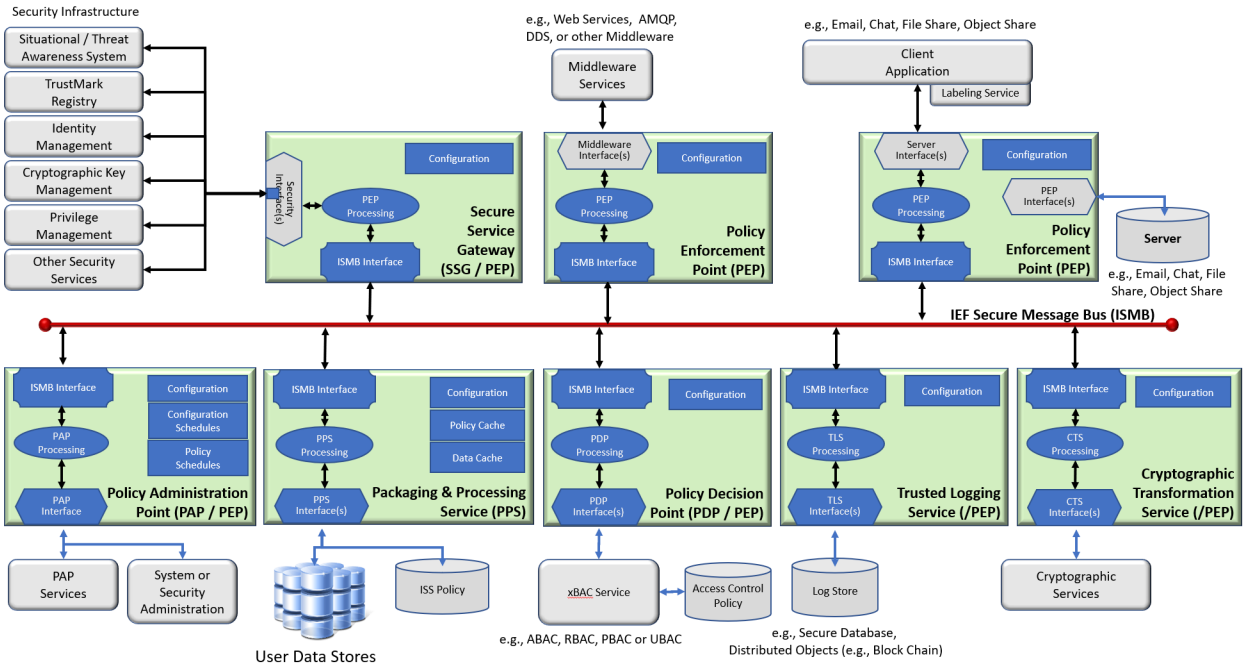


Figure 19 - Alternate Component Architecture

2.2.1 Email Compliance Point

Email compliance requires the implementation of an Email PEP (Clause 10.2.1) tailored to intercept emails transiting between an Email Client or User Mail Agent and the Email Server (Transfer or Delivery Agent). The Email-PEP is a proxy between the email client and the email service. The Email-PEP validates and verifies the receipt or release of each email to each recipient. For each transaction, the Email-PEP must verify that the sender is authorized to release each information element (email body and attachments) and that the recipients can receive and access those elements.

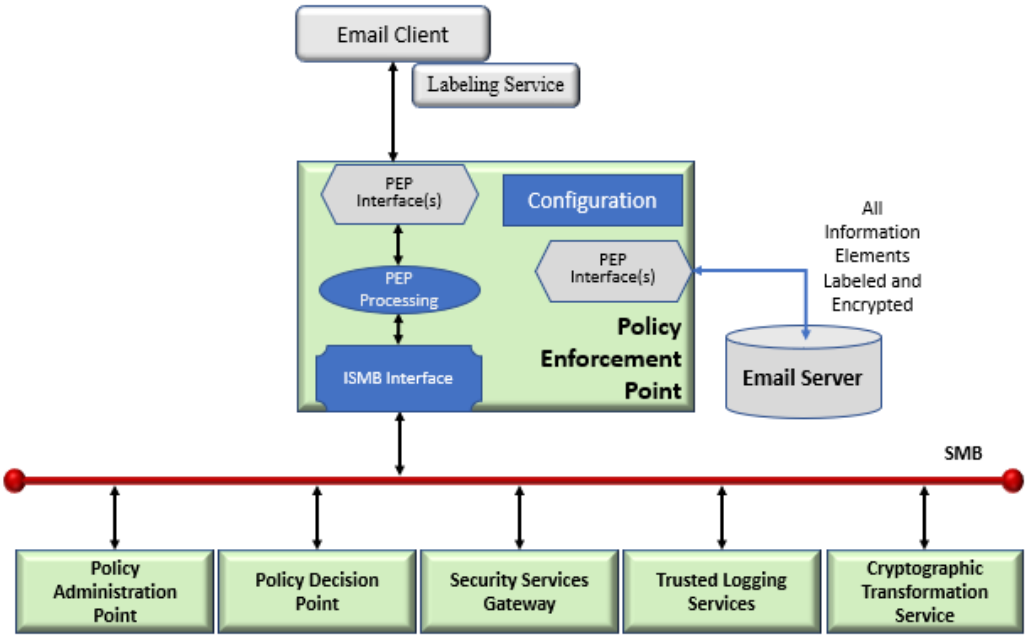


Figure 20 - Email PEP

As illustrated (Figure 20), the Email-PEP is an integration point and a proxy between the user's chat client and the server. On the release of an email from an email client, the Email-PEP:

1. Intercepts each email from the email client(s),
2. Extract the information elements contained within the email message,
3. Extract the metadata markings (/labels/tags) bound to each information element (e.g., body and attachments) in the email,
4. Retrieves the sender's and recipient's attributes (authorizations),
5. Stages each authorization for adjudication by the PDP:
    a. The sender is authorized to release the information element and
    b. Each Recipient is authorized to receive each information element.
6. Enforces the PDP determination (e.g., release, no-release, or indeterminant); in response, the PEP may:
    a. Release the email to the server,
    b. Block the entire email for release from All recipients,
    c. Block the entire email for a specific recipient (from the TO, CC, and BCC lists),
    d. Redact a specific information element the sender is not authorized to release or

`

    e. Redact a specific information element the recipient is not authorized to receive,
7. Releases the email to the users' email client,
8. Issue a warning to the sender, identify issues or redactions to the email, and
9. Log the transaction.

On receipt of an email from a user, the Email-PEP:

1. Intercepts each email received from the server,
2. Extract the information elements contained within the email message.
3. Extract the metadata markings (/labels/tags) bound to each information element,
4. Gathers the recipient's attributes (authorizations),
5. Stages the authorization process (e.g., PDP/ABAC) for each information element,
6. Enforces the PDP determination (e.g., release, no-release, or indeterminant); in response, the PEP may:
    a. Release the email to the email client,
    b. Block the Email message, or
    c. Redact any offending elements from the email and release the rest to the email client,
7. Release the email to the server,
8. Issue a warning to the sender, identify issues or redactions to the email, and
9. Log the transaction.

2.2.2 File Share Compliance Point

File share compliance requires the implementation of a File-PEP (Clause 10.2.2) tailored to intercept exchanges between a user application (e.g., office application, file manager, or file browser) and the protected file server. The File-PEP verifies that the user is authorized to access and perform any requested transactions with specified files.

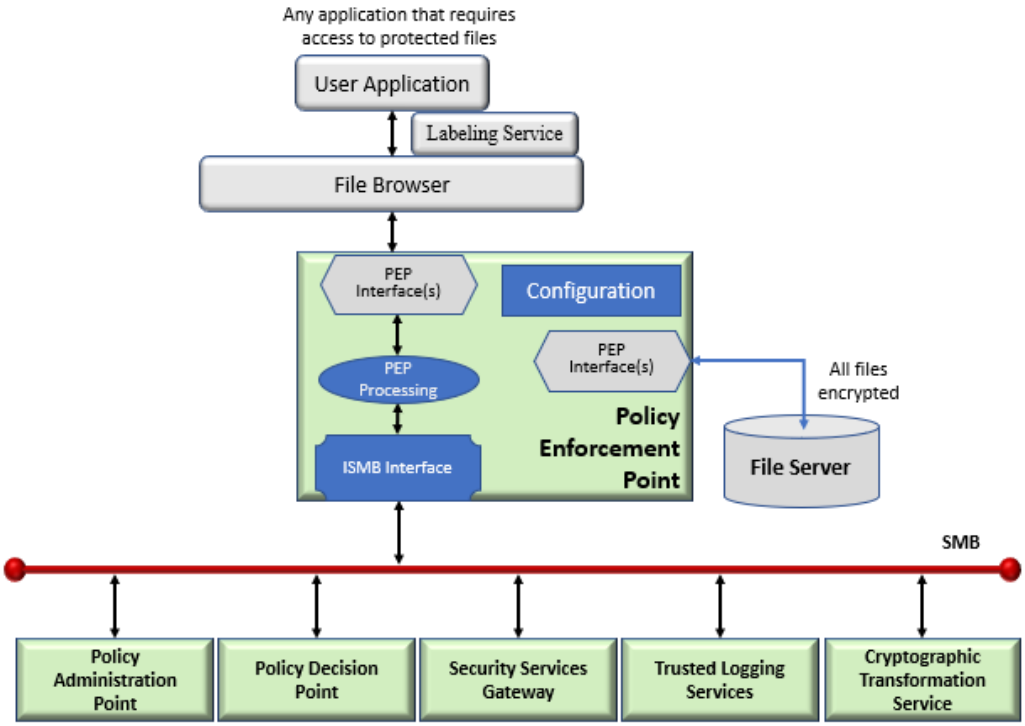


Figure 21 - File Sharing PEP

As illustrated in Figure 21, the File-PEP acts as an integration point and a proxy between the user's application and secure file share. The File-PEP interacts with other IEF services to perform the following functions. When retrieving files or information about a file (e.g., metadata) from the store:

1. Intercepts each file request from a user application to the protected file share,
2. Authenticates each user(s)/application/device making the request,
3. Retrieves the file(s) from the store,
4. Extract the metadata (/labels/tags) bound to each requested file,
5. Retrieves recipient (e.g., user, application, and device) attributes (e.g., authorizations) for requesting user,
6. Stages 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),
7. Receives the release determination from the PDP (e.g., release, no-release, or indeterminant) and, in response and
8. Log the transaction.

When receiving a file from a user's application:

1. Intercept each operation from a client application to the protected file share,
2. Authenticate each user(s)/application making the request,
3. Extract the relevant metadata (/labels/tags) bound to each file sent by the user,
4. Retrieves recipient (e.g., user, application, and device) attributes (e.g., authorizations) for requesting user,

`

5. 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),
6. Receives the release determination from the PDP (e.g., release, no-release, or indeterminant) and, in response, may:
    a. Decrypt the file (as protected files are encrypted at rest in the file store) and release it to the user's application or
    b. Rejects the request and terminates the request,
7. Encrypts and stores authorized files and
8. Log the transactions.

2.2.3 Instant Messaging or Chat Compliance Point

Instant messaging or chat compliance requires a chat PEP (Clause 10.2.3) tailored to intercept each message between the IM or chat client and the message-oriented middleware or server. The Chat-PEP verifies that the user can request a specific operation (e.g., create a chatroom, join a chatroom, list chatrooms, receive a message, send a message).

The Chat-PEP (Figure 22) is an integration point between the IM or chat client, the IEF services, and the Users' 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.

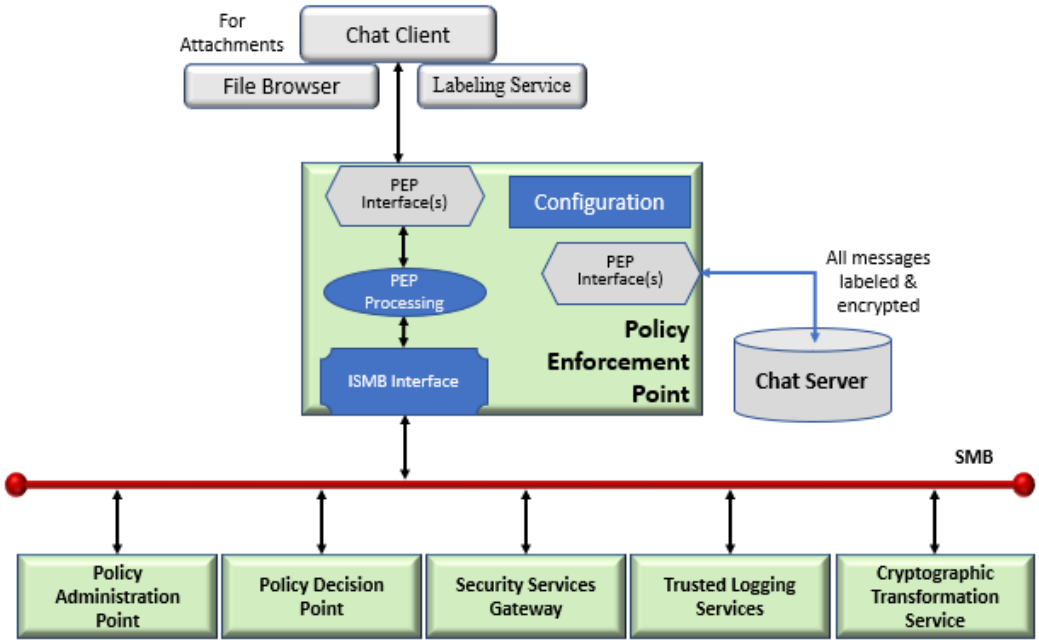


Figure 22 – Chat-PEP

2.2.4 Data Messaging Compliance Point

Integrating the user client to the PPS requires a Messaging-PEP (Clause 10.2.4) tailored to intercept communications between a user application and a PPS that operates in a server configuration. The Messaging-PEP authenticates each recipient and enforces the PDP's determination regarding each recipient's authorization to request and receive information from the associated PPS.

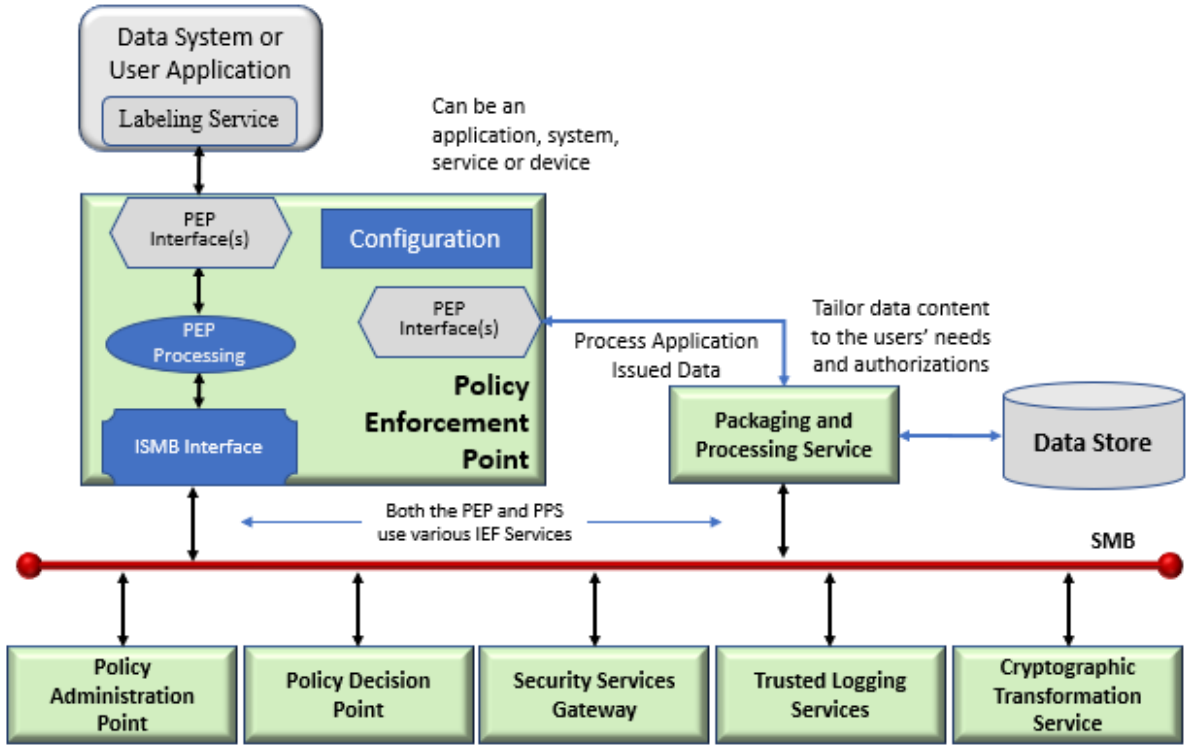


Figure 23 - Messaging PEP

The Messaging-PEP is the integration point between IEF services and the Users' information system application environment. The Messaging-PEP must be able to:

1. Intercept messages between each user application and the PPS, e.g.:
    a. Data Message,
    b. Data Request,
    c. PAP Administrative Message (assuming and External PAP Configuration), or
    d. Streaming Data,
2. For a **Data Message**, the Messaging-PEP:
    a. Extracts the metadata markings (/labels/tags) bound to each message,
    b. Authenticates the user,
    c. Gathers recipient and PPS authorizations to specifically marked information,
    d. Stages the authorization process for each recipient,
    e. Enforce policy determination for the release of the PPS (is the PPS authorized to process and store the content of the message?),
    f. Releases authorized data to the PPS for processing and packaging, and
    g. Logs the transaction,

3. For a **Data Request**, the Messaging-PEP:
    a. Extracts the metadata markings (/labels/tags) bound to each message,
    b. Authenticates the user,
    c. Gathers Requestor and PPS Attributes,
    d. Stages the authorization process for each requestor,
    e. Enforces PDP policy determination for the Request of the PPS (*is the Requestor authorized to request data from the PPS?*),
    f. Releases authorized data requests to the PPS to package data content authorized for the requestor, and
    g. Logs the transaction,
4. For an **Administration Message**, the Messaging-PEP:
    a. Extracts the metadata markings (/labels/tags) bound to each message,
    b. Authenticates the PAP (e.g., PAP, System Administration Application, Data Security Application),
    c. Gathers PAP attributes,
    d. Stages the authorization process,
    e. Enforces PDP policy determination (the specific PAP authorized to administer the PPS),
    f. Releases authorized administration messages to the PPS,
    g. Logs the transaction,
5. For **Streaming** Data, the Messaging-PEP:
    a. Authenticates the Stream source,
    b. Gather recipient and PPS authorizations,
    c. Stages the authorization process for each recipient,
    d. Enforces policy determination for the release of the PPS,
    e. Accepts the authorized stream for its duration:
        i. Extracts the metadata markings (/labels/tags) bound to the stream,
        ii. Releases authorized Metadata and associated Stream to the PPS for Processing and Packaging (the PPS determines the releasability of the stream to active Information Exchange Specifications (IES[9]) based on the extracted metadata), and
    f. Logs the transaction.

2.2.4.1 Packaging and Processing Service (PPS)

The Packaging and Processing Service (PPS: Clause 11) governs the receipt and release of data content per user-defined exchange and semantic policy. Where the PEP is responsible for governing access control to the PPS, the PPS governs:

1. The processing of message content:
    a. Inbound:

[9] See Information Exchange Packaging Policy Vocabulary (IEPPV) for additional details, https://www.omg.org/spec/IEPPV.

`

i. Deconstructs the message,

ii. Extracts the data and metadata elements,

b. Outbound:

i. Verify data releasability based on data and metadata content (see IEPPV "Dynamic Filters"),

ii. Format data and metadata elements,

iii. Construct message,

iv. Bind metadata to message and

v. Release and route message,

2. The processing of received data and metadata content:
    a. Decrypts the message elements (as necessary),
    b. Parses the data and metadata elements,
    c. Transforms the data and metadata elements to user data semantics,
    d. Maps data to internal data semantics (e.g., schema) and
    e. Marshals the elements and attributes,
3. The packaging and release of data and metadata content:
    a. Execute release watchpoints and requests,
    b. Aggregate data elements authorized to the recipient(s),
    c. Transform attributes and aggregates to exchange semantics,
    d. Label aggregates (as required),
    e. Redact attributes and aggregates (as required)
    f. Provide data and metadata for formatting and release.

The Messaging-PEP and the PPS combine to deliver ISS and DCS for structured and semi-structured data elements.

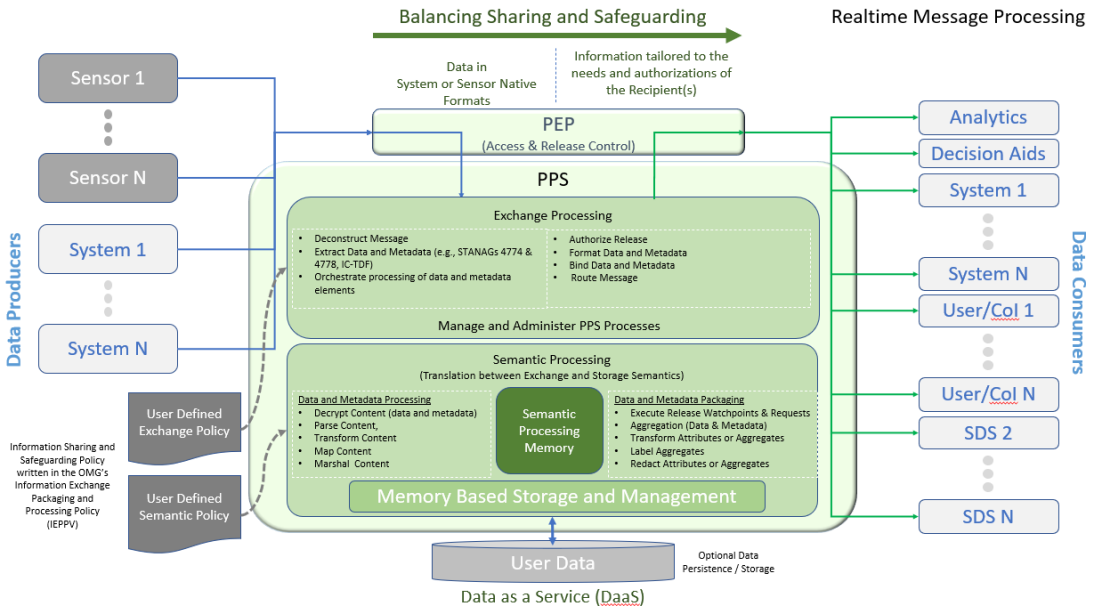


Figure 24 - PPS Features

2.2.4.1.1 Information Exchange Packaging Policy Vocabulary

The PPS must allow users to ingest information sharing and safeguarding (or DCS) policies conforming to the Information Exchange Packaging Policy Vocabulary (IEPPV), which governs the processing and packaging of received data messages for release.

2.2.4.1.2 Exchange Processing Services

The Exchange Processing Services (EPS) execute and enforce the Information Exchange Specification (IES) policy (rules and constraints) governing the format, protocols, and routing of messages specified in an agreement between partners and systems, e.g.:

1. Information Sharing Agreement (ISA),
2. Memorandum of Understanding (MoU),
3. Information Exchange Requirement (IER) or
4. API Specification.

The PPS enforces the IES elements of ISS and DCS policies defined using the IEPPV, including:

1. The semantics associated with each exchange specification,
2. The guard filters to limit or protect against the release of unauthorized content and
3. The rules that govern the formatting and routing of messages.

For more information about IES policies, refer to Normative Reference K.

Upon receipt of a message, the PPS:

1. Deconstructs the message,
2. Extract the message metadata and payload and
3. Passes the metadata and message data to Semantic Processing Services to be:
    a. Decrypted (if required),

`

b. Parsed,

c. Transformed to storage semantics and schema,

d. Mapped to the storage semantics and schema, and

e. Marshaled to the Memory-based Data Management services.

Upon receipt of the data and metadata objects from the Semantic Processing Services, the PPS:

1. Verifies that the data is releasable through the Information Exchange Specification (Guard Function),
2. Formats the outbound message (encrypting as necessary),
3. Binds metadata to the data message and
4. Routes the message to the PEP for authorization and release.

2.2.4.1.3 Semantic Processing Services

The Semantic Processing Services execute and enforce the Information Exchange Specification (IES) policy (rules and constraints) governing the processing and packaging of data content received by the PPS.

On receipt of message data and metadata, the PPS:

1. Receives the data and metadata content from the exchange service,
2. Parses the data and metadata content from the structures,
3. Maps the data and metadata to the specified data store(s),
4. Marshals the data elements (/attributes) to the Memory Based Data Management Services (MBDMS).

On receipt of the data from the Memory Based Data Management services, identify if policies require PPS to share the data with other systems. If so, the semantic services schedule the execution of semantic policies that govern the preparation and release of data tailored to assigned needs and authorizations. The Semantic Processing services build (aggregate, transform, label, and redact) the data and metadata element required for each IES and route the data and metadata object to the Exchange Processing services.

2.2.4.2 Memory-Based Data Management Services

The Memory-Based Data Management Services (MBDMS) bridge the PPS and the user-specified storage technology. The MBDMS maintains a memory-based data environment that services semantic processing and persisted data to the user-specified store. Upon receiving data elements, the MBDMS:

1. Produces the necessary Wrapper Elements (see IEPPV) and persists the data elements to the user-specified store,
2. Identifies all WatchPoint (see IEPPV) triggers by the release of data to involved IESs,
3. Schedules for the execution of the appropriate semantic policies and
4. Routes associated data and metadata objects to the EPS to be formatted and released.

2.3 Alternate PEP Configuration Structured Data Messaging Compliance Points

The following clauses outline alternate configurations for IEF Services, extending the flexibility of the IEF and enabling continuous development of IES capability.

2.3.1 Multiple PEP for System-to-System Exchanges

The IEF services can be configured to deliver a transformation gateway between two or more systems. In this configuration, data received from one system can be transformed (restructured, formatted, and routed) to any number of interoperating systems. The ISS and DCS policy, enforced by the PPS, governs data transformation, with the PEP providing the integration to the technology stack specified for the interchange. Figure 25 Illustrates using the PPS and IEPPV to employ policy to exchange data between systems using different protocols and technologies to exchange data (SOAP/XML and REST/JSON and DDS, respectively).

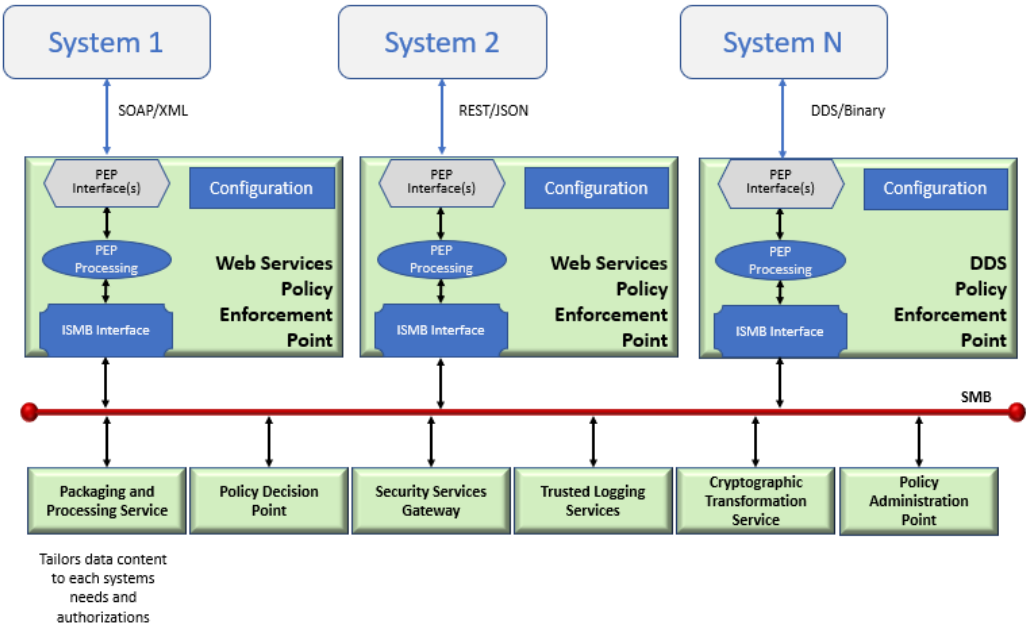


Figure 25 - Additional System-to-System Compiance Options

During the processing and repackaging data elements, the PPS can transform the data to meet the variations of the exchange semantics and enforce DCS controls (e.g., data transformation, data masking, data redaction, and data labeling) as required by policy. Typically, this configuration would operate in memory and not persist the data elements.

2.3.2 Extended PEP Services

The Integrated PEP incorporates the functionality and interfaces of the SSG, PDP, and CTS. This configuration is most useful when employing off-the-shelf and SaaS solutions to deliver ICAM, access control, cryptography, and key-management functions.

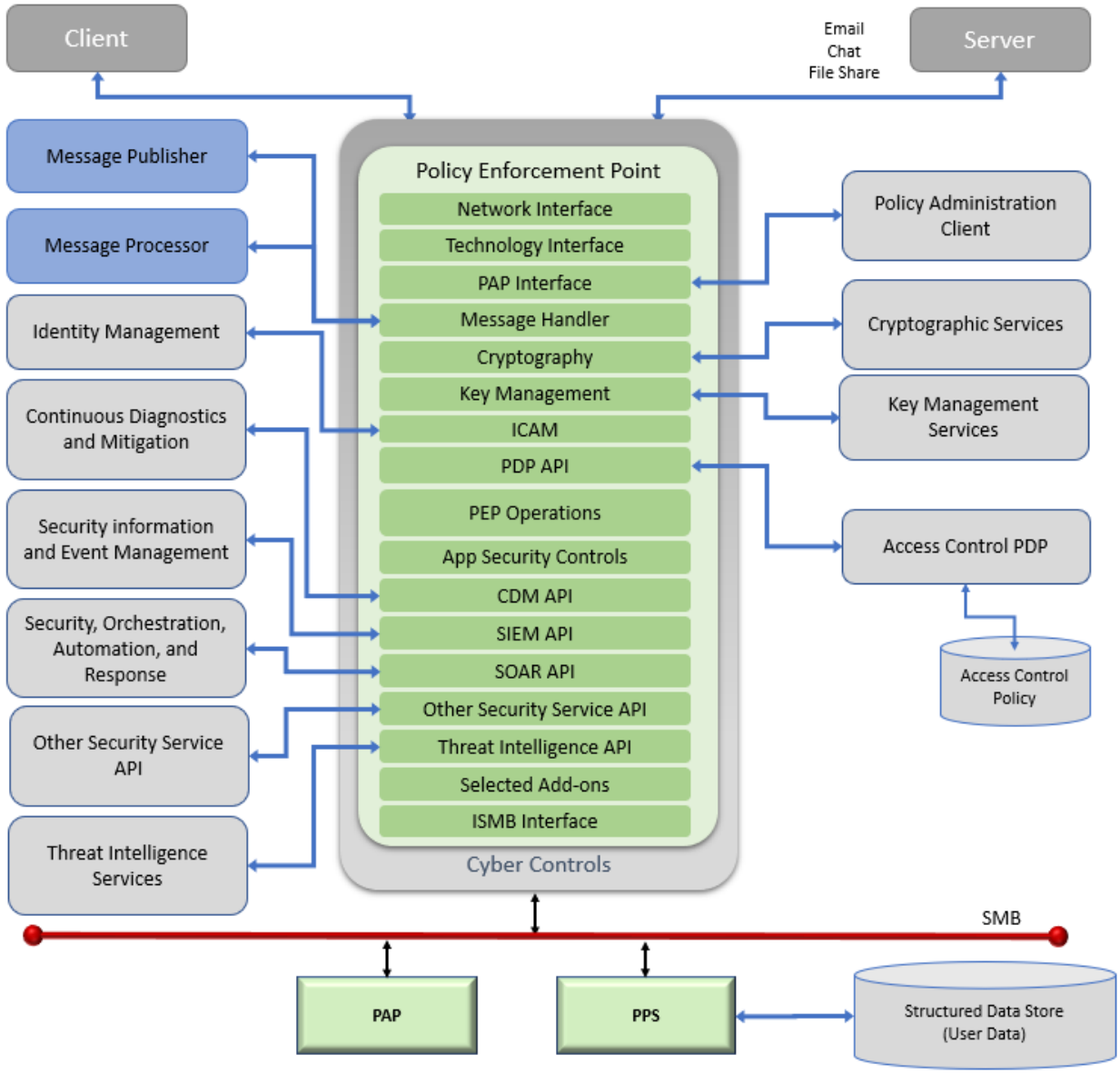


Figure 26 - Integrated PEP

The PPS will require access to identity information, key management, and cryptographic functions through the PEP in this configuration. This reference architecture defers the allocation of IEF-specified functions to the PEP, CTS, SSG, or external services of the implementor (user, vendor, or integrator).

2.3.3 Secure Data Service

The Secure Data Service (SDS) seeks to implement a data enclave that can reside on any network where the boundary protections surround the data. As illustrated in Figure 27, the IEF services are implemented as policy enforcement points to user-specified systems and services.

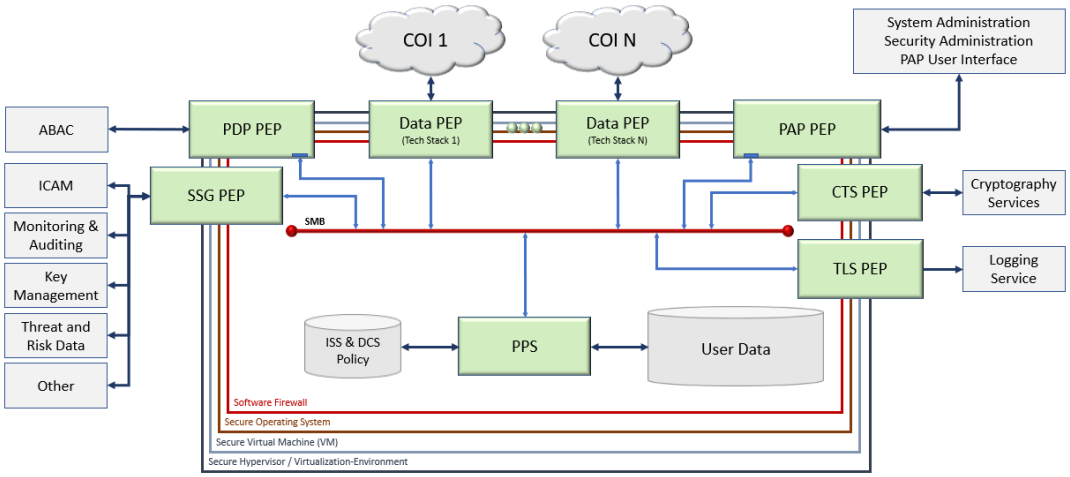


Figure 27 - Secure Data Service

All interactions with the SDS are through a PEP that authenticates and authorizes the user (e.g., individual, system, service, or device) initiating that transaction. All user data, policy, and configuration are encrypted within the VM, operating system, and firewall boundary. For PPS-initiated exchanges, the Data PEPs authorize the PPS to release the data and each recipient to receive that information based on the following:

1. The PPS's attributes retrieved from the Identity Credential and Access Management (ICAM) services,
2. The recipient's attributes retrieved from ICAM services and
3. The labels (metadata) are bound to the data message.

Every exchange is authenticated and verified. The PEP prepares an authorization request and issues it to the PDP for adjudication. The PEP then enforces the PDP determination of the exchange.

As illustrated, security measures encapsulate user data as specified by DCS, and through labeling and cryptography, security is carried by the data wherever it is.

`

3 Normative References

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

3.1 Normative Specifications

The following normative documents contain provisions directly applicable to this specification. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply.

NREF1. Information Exchange Packaging Policy Vocabulary, htpps://www.omg.org/spec/IEPPV

NREF2. eXtensible Access Control Markup Language (XACML), http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.pdf

NREF3. XML Schema, https://www.w3.org/standards/techs/xmlschema#w3c_all

NREF4. Data Distribution Service™ (DDS™), http://www.omg.org/spec/DDS/

NREF5. DDS Security™ Specification (DDS-SECURITY™), http://www.omg.org/spec/DDS-SECURITY/

NREF6. Information Exchange Packaging Policy Vocabulary™ (IEPPV™), http://www.omg.org/spec/IEPPV/

NREF7. Unified Modeling Language™ (UML®), http://www.omg.org/spec/UML/

3.2 Reference Materials (Informational)

The following comprises the background to this specification.

REF1. Secure Data Service (SDS) Operational Context Document, https://www.omg.org/news/whitepapers/20201223-ASMG_DCS_SDS_Operating-Concept-v1-01.pdf

REF2. Data-Centric Security for System-to-System Information Sharing and Safeguarding Policy Development, https://www.omg.org/news/whitepapers/20210125-ASMG_DCS_S2S_Policy-Modelling-v1-02.pdf

REF3. https://go.omgprograms.org/l/658223/2019-05-17/41hhs

REF4. Information Assurance Architecture, K. Willet, CRC Press, 2008

REF5. The Security Development Lifecycle, Howard & Lipner, Microsoft Press, 2006

REF6. Building a Global Information Assurance Program, Curtis & Cambell, Auerbach Publications, 2003

REF7. Adaptive Information, Pollock & Hodson, Wiley-InterScience, 2004

REF8. Information Security Management, Raggad, CRC Press, 2010

REF9. Core Security Patterns, Steel et al., Prentice Hall, 2006

REF10. Information Security Architecture, Killmeyer, Auerbach Publications, 2006

REF11. Cloud Security and Privacy, Mather et al., O'Rilley Books, 2009

REF12. Asset Protection and Security Management Handbook, POA Publishing, 2003

REF13. Information Security Management Handbook, Tipton et al., Auerbach Publications, 2007

REF14. SOA Security, Kanneganti et al., Manning Publications, 2008

REF15. http://www.telegraph.co.uk/news/worldnews/wikileaks/10210236/WikiLeaks-five-things-we-learned-from-the-Bradley-Manning-case.html

REF16. http://www.computerworld.com/s/article/9242493/Snowden_s_role_provided_perfect_cover_for_NSA_data_theft

REF17. http://www.theglobeandmail.com/news/national/convicted-spy-delisle-sold-csis-names-to-russians-court-told/article8030374/

REF18. 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

REF19. Understanding the Security, Privacy, and Trust Challenges of Cloud Computing, Debabrata Nayak: http://riverpublishers.com/journal/journal_articles/RP_Journal_2245-1439_127.pdf

REF20. MILS: Architecture for High-Assurance Embedded Computing, Luke, Taylor, Uchenick: http://www.crosstalkonline.org/storage/issue-archives/2005/200508/200508-Vanfleet.pdf

REF21. XACML, ABAC, Privacy-preserving access-controls, Bodriagov: http://www.csc.kth.se/~buc/PPC/Slides/accesscontrololeksandr.pdf

REF22. What Can You Do with XMPP? Barrett, 2009:http://fyi.oreilly.com/2009/05/what-can-you-do-with-xmpp.html

REF23. SAMSON Technology Demonstrator Architectural Design Document Phase IV SD002, Dr Daniel Charlebois et al., Defence Research and Development Canada, 2013

REF24. SAMSON Technology Demonstrator Detailed Design Document Phase IV SD004, Dr Daniel Charlebois et al., Defence Research and Development Canada

REF25. NIST 800-207 – Zero Trust Architecture, https://csrc.nist.gov/publications/detail/sp/800-207/final

REF26. NIST 800-53 - Security and Privacy Controls for Information Systems and Organizations, https://csrc.nist.gov/publications/detail/sp/800-53/rev-5/final

3.3 Additional Specifications and Standards

The following specifications and standards implementations of the IEF-RA and services:

REF27. Logical Entity eXchange Specifications 4.0 (LEXS), https://lexs.codeplex.com/

REF28. Unified Profile for the Department of Defense Architecture Framework (DoDAF) and the Ministry of Defence Architecture Framework (MODAF), http://www.omg.org/spec/UPDM/

REF29. Unified Architecture Framework, https://www.omg.org/spec/UAF

REF30. SAML V2.0, http://saml.xml.org/saml-specifications

REF31. Extensible Messaging and Presence Protocol (XMPP), Source: http://www.ietf.org/rfc/rfc3920.txt, Source http://xmpp.org/

REF32. Instant Messaging and Presence Protocol (IMPP), https://www.ietf.org/rfc/rfc2779.txt

REF33. XACML - eXtensible Access Control Markup Language - http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.html

REF34. IC-TDF (Intelligence Community Trusted Data Format) - https://www.dni.gov/index.php/who-we-are/organizations/ic-cio/ic-cio-related-menus/ic-cio-related-links/ic-technical-specifications/trusted-data-format-base

REF35. ADatP-4774 - Confidentiality Metadata Label Syntax (Source NATO)

REF36. ADatP-4778 - Metadata Binding Mechanism (Source NATO)

REF37. ADatP-5636 - NATO Core Metadata Specification (NCMS), (Source NATO)

REF38. ADatP-5644 - Web Service Messaging Profile (WSMP), (Source NATO)

REF39. ACP-240 - Data-Centric Interoperability Concepts and Design Guidance

REF40. DOD Reference Architecture Description - DoD CIO

REF41. National Information Exchange Model (NIEM), http://www.NIEM.gov

4 Terms and definitions

The terms and definitions for this specification have been included in Annex E – Glossary.

`

5 Symbols

This specification does not define additional symbols; all symbols conform to the standard UML.

6 Additional Information

6.1 Intended Audience

This specification will interest end users, analysts, and integrators using this profile to define information exchange specifications and tool vendors interested in developing tools to support the development and sustainment of information interoperability solutions. End users, auditors, and developers will understand the semantic and business rules (sharing and safeguarding) for information exchange.

6.2 Acknowledgements

The following organizations are the direct submitters to this specification:

1. Advanced Systems Management Group (ASMG) Ltd. and
2. Defence Research and Development Canada (DRDC), Centre for Security Sciences (CSS).
3. ASMG personnel acted as the principal authors and editors of the specification and models.
4. 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 specification.

Contributors (/Contributing Entities)

In particular, the submitter would like to acknowledge the participation and contribution of the following individuals: Michael Abramson (ASMG), Jean Claude Lecomte (ASMG), Eric Penwill (ASMG), Sebastian Schneider (ASMG), Jonathan Austin (ASMG), Vijay Mehra (KYM Advisors), and Dr Daniel Charlesbois (DRDC/CSS).

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

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

The XSDs and Additional data types used for testing the IEF-RA during the NATO CWIX exercises (2018-2023) have been included as informational elements in Annexes A, B, and C. These elements were specifically developed for CWIX and should be included in separate technical specifications (e.g., PAP, PEP, PPS, SSG, CTS, and TLS); they are only informational elements in this reference Architecture.

6.4 IEF RA Objective

The IEF Reference Architecture gives readers an overview of the software services and their integration patterns that deliver policy-driven, data-centric ISS. The architecture provides specialized definitions for standard security services that conform to the protection needs of information elements (i.e., File, Email, Text/Instant Message, and Structured Message). These services include:

1. Policy Enforcement Point (PEP),
2. Policy Decision Point (PDP), and
3. Policy Administration Point (PAP).

XACML architecture standards form the basis for many IEF concepts. The IEF specializes in XACML architectural building blocks to protect data and information elements.

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 will provide a single point of integration to client services, including:

`

1. Identity, Credential and Access Management Services (ICAM),
2. Access Management Components (e.g., ABAC),
3. Cryptographic Key Management Services (key generation and escrow),
4. Directory Services (e.g., Active Directory or LDAP), and
5. Situational Awareness or Incident Management Services (providing operational context).

As illustrated in Figure 28, the user's information processing applications and information exchange (file-share, email, text messaging, and middleware) services connect through a PEP (proxy server). The PEP intercepts each transaction and provides it to the specialized PEP to enforce user-specified access and release control policies.

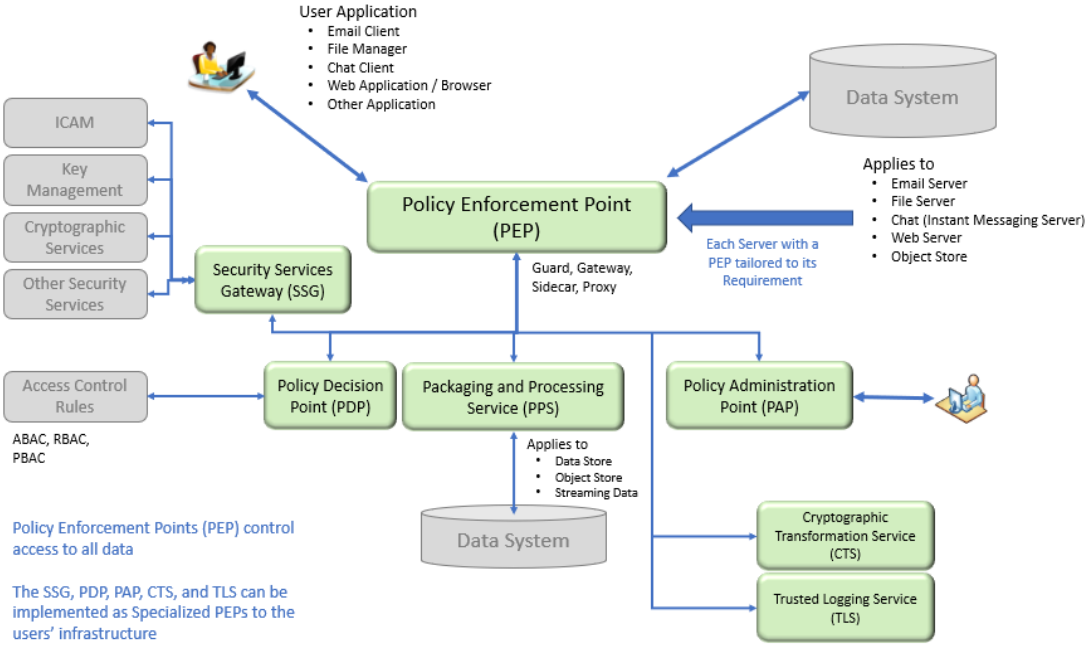


Figure 28 - All Connections through a PEP

6.5 Modelling Conventions

The specification uses standard UML and UML Profiles to model IEF components, including UML Class and Data diagrams.

6.6 OMG Related Work

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

1. IEPPV: The Information Exchange Packaging Policy Vocabulary (formal/2015-05-06) defines the policy vocabulary and UML profile for developing PPS policies.
2. UAF: The Unified Architecture Framework (DTC/2016-08-01) enables users to specify the IEF deployment in conjunction with the rest of their enterprise architecture. The IEPPV is aligned with the UAF, which further aligns these efforts.
3. UPDM: See UAF
4. DDS: The Data Distributions Service acts as a PSM for the ISMB or a data distribution capability in the User's environment
5. DDS-Security: DDS Security fortifies the security of IES and User data distribution capability.

6. C4I, MARS, Data Residency, Systems Assurance, and others seek tagging and labeling specifications. If any specifications are realized, the metadata definitions in Clause 17 and Annex C and the enumerations in Annex D can be replaced with the new specifications.
7. SRP: Secure Relationship Protocol fortifies the network connections that enable DDS.
8. ERMA is the Enterprise Resource Management Attribute specification for software, hardware, and network bills of material (BOM).

`

7 IEF Reference Architecture Specification

7.1 IEF-RA Scope

This specification defines the Information Exchange Framework's architecture (e.g., services, APIs, and messages).

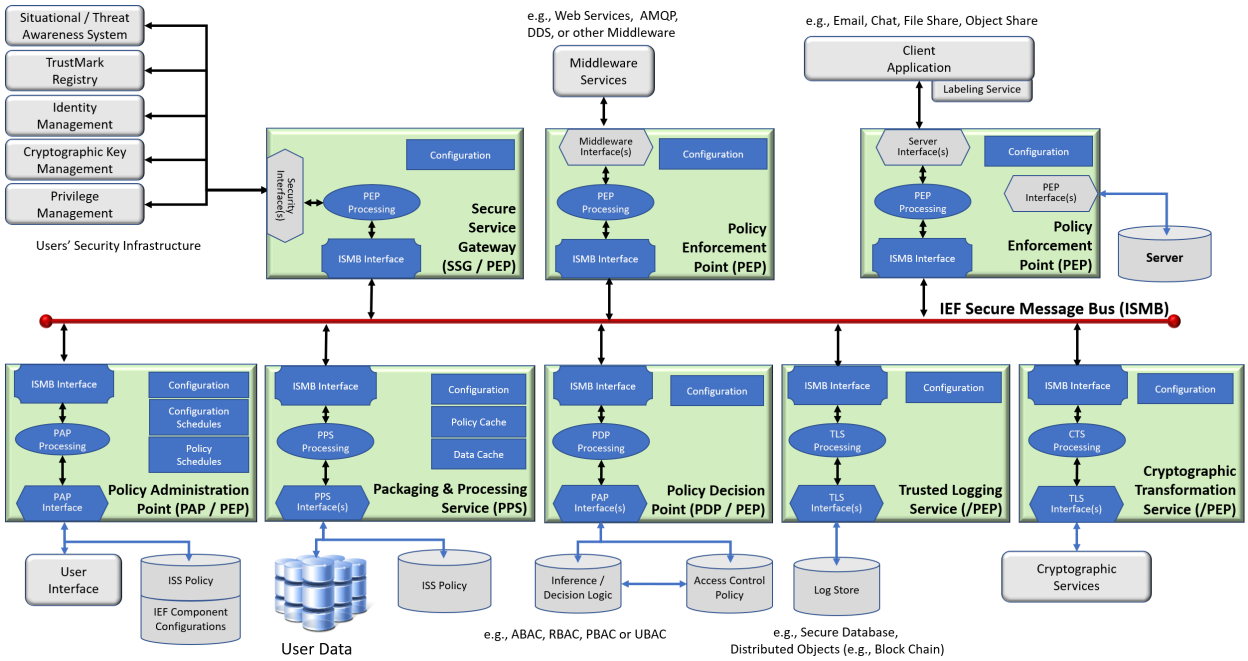


Figure 29 - IEF Service Elements

As illustrated in Figure 29, this specification defines seven services that combine providing an information-sharing and safeguarding capability:

7.1.1 The IEF in Operations

1. A core design principle of the framework is that the ownership/control of policies rests with the owner/steward of the data. This principle implies that:
2. Data owners/stewards have complete control over the access to and release of data/information under their data and information custody,
3. The Data Owner develops, tests, deploys, manages, and administers and manages data policy (rules /instructions/constraints),
4. Data policies are independent of the processes, services, and technologies used to enforce them and
5. Data policies (executable or machine-actionable) reflect and are traceable to the policy instruments endorsed by each data owner/steward.

The IEF separates 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. The IEF defines a machine-readable serialization for ISS policy that governs PPS operations. IEF components ingest user-defined ISS policy at runtime to enable mission-specific operations and real-time administration. IEF conformant environments enable stakeholders (data owners) to develop, test, and deploy ISS policies under an independent lifecycle, separated from the software lifecycle used to develop the enforcing services.

The PEPs also provide lower-grained access and release control using PDP controls and traditional access management based on metadata bound to data objects.

The PAP provides user access to the IEF policy environment to manually or automatically adapt ISS (or DCS) policy and component configurations to changing environmental or mission requirements. The PAP uses a messaging interface to communicate user commands to the individual IEF components and monitor their status and operating state.

The IEF identifies and defines service interface specifications for policy administration, decision, and enforcement services that will enable the integration of one or multiple vendor components into a desired ISS capability.

7.1.2 IEF Component Details

The following clauses outline core and supporting IEF services:

1. Core components:
    a. Policy Enforcement Points (PEP – Clause 10):
        i. Email-PEP (Clause 10.2),
        ii. File-PEP (Clause 10.3),
        iii. Instant Messaging (CHAT)-PEP (Clause 10.4), and
        iv. Structured Messaging PEP (Clause 10.5), and
    b. Policy Decision Point (PDP – Clause 9),
    c. Packaging and Processing Service (Clause 11),
2. Supporting components:
    a. Policy Administration Point (PAP – Clause 8),
    b. Security Services Gateway (SSG – Clause 12),
    c. Cryptographic Transformation Service (CTS – Clause 13).
    d. Secure Messaging Bus (ISMB – Clause 14), and
    e. Trusted Logging Services (TLS – Clause 15),

Additional Components:

1. PAP-PEP implementation enables IEF components to interoperate with external PAP or administration systems within the security infrastructure.
2. CTS-PEP implementation enables IEF components to interoperate with external user-specified cryptographic services as if operating on the SMB.
3. TLS-PEP implementation enables IEF components to interoperate with external logging services as if operating on the SMB.
4. PDP-PEP implementation that enables IEF components to interoperate with external PDP services as if operating on the SMB.
5. Integrated PEP implementation communicates directly with the users' security infrastructure vs using the services on the SMB.

7.2 IEF Component Characteristics

The following clauses outline IEF components' core characteristics and specializations (or services). Subsequent clauses describe these characteristics and specializations in increasing detail.

7.2.1 IEF Component Common Operations

The Policy Enforcement Point (PEP – Clause 10) intercepts each information element transiting between a data producer and a data consumer using Email, Instant Messaging and File Share, and Data-message (e.g., XML, JSON, and BSON) exchanges. The PEP ensures that each user can perform the requested action on the specified information element(s). The PEP requires that each information element contains security, privacy, and other user markings required by the PDP. As illustrated, there are PEPs tailored to the specific protocols of the exchange.

`

In addition, the SSG, PDP, CTS, and TLS can also be configured as PEPs to integrate the IEF and user-specified security infrastructure that performs the requisite functions. The components now form integration points to external capabilities (e.g., ICAM, ABAC, continuous monitoring, and logging) and are identified as PEP specializations. This change derives from evolving requirements (e.g., Zero Trust Architecture) that require all services to authenticate and authorize each exchange of resources. The IEF-RA has adopted this best practice.

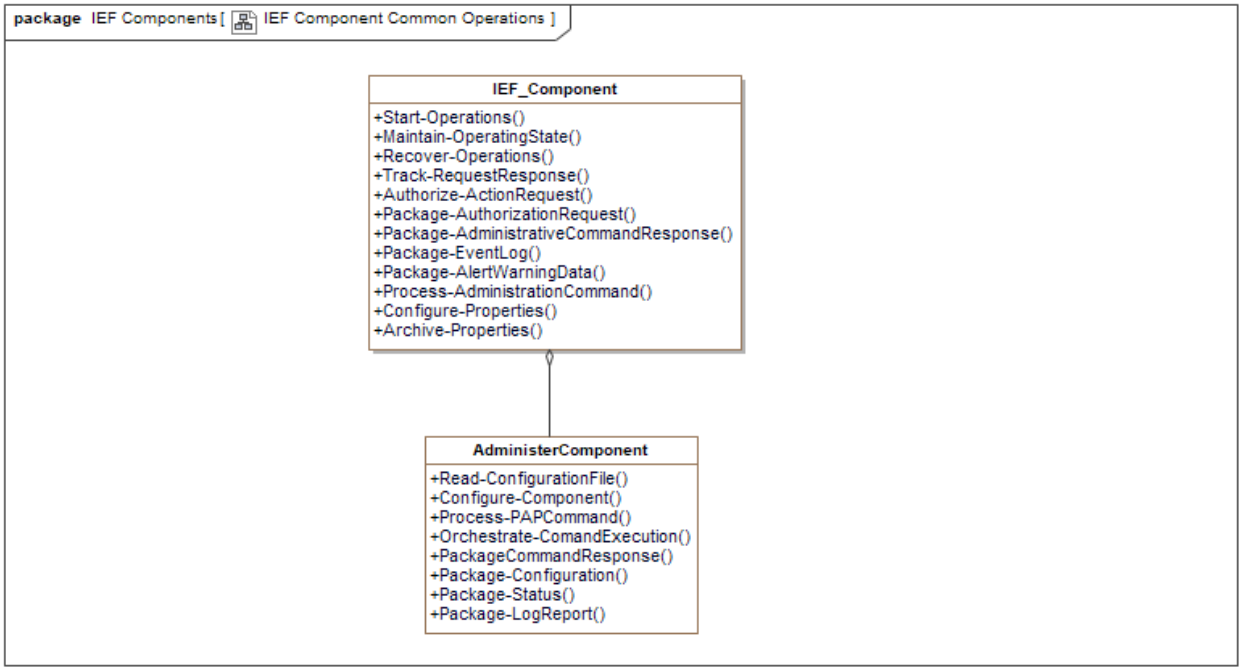


Figure 30 -IEF Component Common Operations

The following table identifies and describes the elements and operations illustrated in the previous figure - IEF Component Common Operations.

Table 5 - IEF Component Common Operations Elements	
Element Name	**Element and Operation Descriptions**
AdministerComponent	Each IEF component provides features to administer its configuration and respond to PAP commands.
	Element Type: Class Owned Operations: Read-ConfigurationFile: The IEF component must provide features that access, parse and map the content to its configuration parameters. Configure-Component:

Table 5 - IEF Component Common Operations Elements	
Element Name	**Element and Operation Descriptions**
	The IEF component must provide features that configure operations based on the parameters in the configuration file, or provided by a PAP. Process-PAPCommand: The IEF component must provide features that parse, map PAP Command messages. Orchestrate-ComandExecution: The IEF component must provide features that execute PAP command Messages. PackageCommandResponse: The IEF component must provide features that aggregate and format configuration and status data resulting from the PAP commands. Package-Configuration: The IEF component must provide features that aggregate and format its running configuration for export or archive. Package-Status: The IEF component must provide features that aggregate and format its operating status for export or archive. Package-LogReport: The IEF component must provide features that aggregate and format a log report.
IEF_Component	The IEF-Component is a generalization of the individual components and identifies the ore software functions (methods/operations) inherited by each component, including: 1. Policy Administration Point (PAP); 2. Policy Decision Point (PDP); 3. Policy Enforcement Point (PEP); 4. Packaging and Processing Service (PPS); 5. Cryptographic Transformation Services (CTS) and 6. Security Services Gateway (SSG). The IEF components represent independent services that communicate using standardized SMB messages. These services can be operated as independent applications using data exchange middleware (e.g., DDS, XMPP, or Service Bus) or as software containers (e.g., Kubernetes).
	Element Type: Class Owned Operations: Start-Operations: Each IEF component must provide features to startup (/power-up /restart), access and ingest the user-specified configuration

`

Table 5 - IEF Component Common Operations Elements	
Element Name	**Element and Operation Descriptions**
	(/property) files, and start the component per the specified default configuration. It is up to the user, vendor, or integrator to decide if the ComponentConfigurationFile comprises one or more files. The ComponentConfigurationFile should be encrypted to prevent unauthorized users from tampering. Maintain-OperatingState: Each IEF component must provide features that continuously monitor, maintain, and record the component's operating state and configuration to recover the component operations after a service failure, network failure operation after a service failure, or power interruption. Recover-Operations: Each IEF component must provide features that re-establish operation from the last known valid operating state and configuration on resumption of operations. Track-RequestResponse: Each IEF component must provide features that track the relationships, fulfillment, and logging of requests to the components. Authorize-ActionRequest: Each IEF component must provide features that stage the authorization of a requested action. These features ensure that each requesting component is authorized to perform the requested operation (/function). Package-AuthorizationRequest: Each IEF component must provide features that gather data the PDP requires to adjudicate the authorization request against the user-specified policies for the specific environment (e.g., mission or operation). (see PDP-AuthorizationRequest Message) Package-AdministrativeCommandResponse: Each IEF component must provide features that gather the data required to publish the reports for the PAP. (see PAP-CommandResponse Message). Package-EventLog: Each IEF component must provide features to gather event or transaction reporting data, prepare a report, and send it to the TrustedLoggingService(s). (see TLS-LogMessage) Each action, event, or transaction is recorded as an entry in the logging service to enable real-time monitoring and forensic auditing. Package-AlertWarningData:

Table 5 - IEF Component Common Operations Elements	
Element Name	**Element and Operation Descriptions**
	Each IEF component must provide features to gather data to package and send an AlertWarning message to the IEF administrator (/PAP) or other user-designated recipients. (See PAP-AlertWarning Message) Alerts and warnings provide indications of issues within the operation of the IEF or the users' information-sharing environment, e.g.: • Persistent attempts to access or release sensitive information to unauthorized recipients; • Receipt of unauthorized information elements; • Unknown message types; • Unknown metadata; • Unknown operational states and contexts and • Unknown decision states received from the PDP. Process-AdministrationCommand: Each IEF component must provide features to parse, process, and execute PAP commands directed at the component. (see PAP-Command Message). Configure-Properties: Each IEF component must provide features to enable the PAP to modify the value of one or more configuration properties. These 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 administrative commands executed by one or more IEF components. Archive-Properties: Each IEF component must provide features that gather current operating and configuration properties, package them as a configuration file, and store them in a specified location in the IEF persistent store.

7.2.2 IEF Component Specializations

The following figure identifies the IEF components defined by this specification.

`

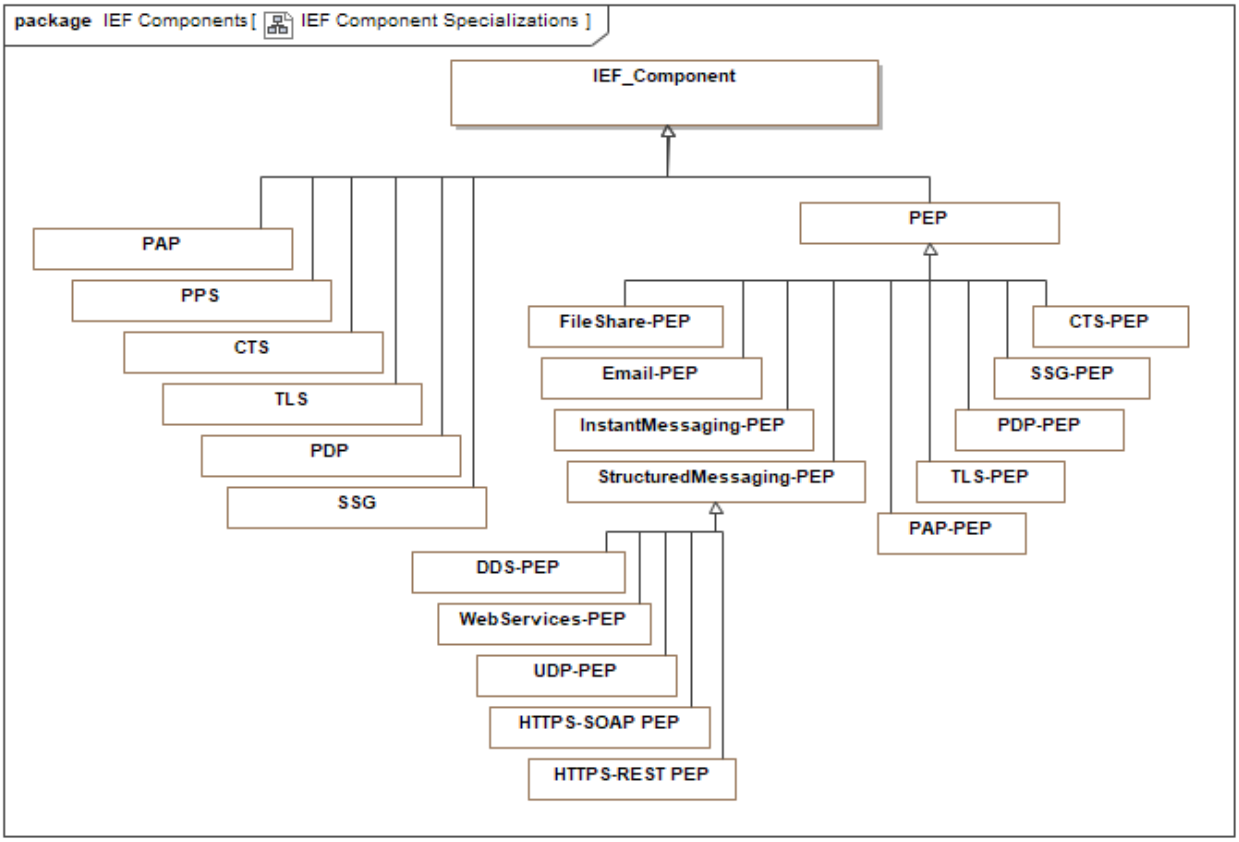


Figure 31 -IEF Component Specializations

The following table describes the elements illustrated in the previous figure - IEF Component Specializations.

Table 6 - IEF Component Specializations Elements	
Element Name	**Element Descriptions**
CTS Type: Class	The Cryptographic Transformation Service (CTS) provides a common interface to cryptographic (encryption and decryption) services required by IEF components. The CTS may utilize internal encryption or decryption services or provide an interface to a user-specified system or services. All encryption services are expected to be FIPS-complaint software modules, services, or appliances.
CTS-PEP Type: Class	This CTS-PEP connects the IEF Services to user-specified cryptographic services through a ZTA-enabled interface. The CTS operates in the same fashion as the SSG, and it is tailored to cryptographic operations.
DDS-PEP Type: Class	The DDS-PEP extends the Messaging PEP by providing the features, functions, and interfaces needed for the IEF to interoperate with Real-time Data Distribution Services (DDS).

Table 6 - IEF Component Specializations Elements	
Element Name	**Element Descriptions**
Email-PEP Type: Class	The Email-PEP operates as a proxy between the email client and the mail server.
FileShare-PEP Type: Class	The File-PEP operates between user applications and the file server. The PEP: 1. Intercepts each file-based action (e.g., open, save, save-as, move, and delete); 2. Authenticates the user; 3. Extract the metadata for each involved file; 4. Gathers user attributes; 5. Composes a PDP adjudication request; 6. Issues the adjudication request to the PDP; and 7. Enforce the PDP receipt and release determinations. (Optional) The File-PEP may redact or interact with the PPS to tailor the content to the recipients' attributes.
HTTPS-REST PEP Type: Class	The DDS-PEP extends the Messaging PEP by providing the features, functions, and interfaces needed for the IEF to interoperate with HTTPS-REST services.
HTTPS-SOAP PEP Type: Class	The DDS-PEP extends the Messaging PEP by providing the features, functions, and interfaces needed for the IEF to interoperate with HTTPS-SOAP services.
IEF_Component Type: Class	The IEF-Component is a generalization of the individual components and identifies the ore software functions (methods/operations) inherited by each component, including: 1. Policy Administration Point (PAP); 2. Policy Decision Point (PDP); 3. Policy Enforcement Point (PEP); 4. Packaging and Processing Service (PPS); 5. Cryptographic Transformation Services (CTS) and 6. Security Services Gateway (SSG). The IEF components represent independent services that communicate using standardized SMB messages. These services can be operated as independent applications using data exchange middleware (e.g., DDS, XMPP, or Service Bus) or as software containers (e.g., Kubernetes).
InstantMessaging-PEP Type: Class	The IM or chat PEP operates between user applications and the IM server. The PEP performs the following: 1. Intercept each message;

`

Table 6 - IEF Component Specializations Elements	
Element Name	**Element Descriptions**
	2. Authenticate the sender and recipients; 3. Extract the metadata from the message of request; 4. Compose and issue an adjudication request to the PDP or ABAC system; 5. Enforce the PDP receipt and release determination for each message or request and 6. Log the transaction to the TLS. The InstantMessaging-PEP specializes in protecting chat or IM Messages. Clause 10.3 provides the details for this interface.
PAP Type: Class	The Policy Administration Point (PAP) provides the functionality to manage and administer IEF components during operation. The IEF defines a general architecture for a PAP, identifying its core sub-components, functions, and interfaces (e.g., protocols and content). The PAP may be implemented as a component within a secure operating environment or as an external capability operating through a PEP. The PAP provides authorized users with an interface and the tools to manage and administer IEF components and policies. The PAP uses standards (Clause 16) messaging to communicate with IEF components through the SMB within its environment. Messages include: 1. PAP Command Message; 2. PAP Command Response Message; and 3. PAP AlertWarning Message. PAP functionality may be delivered as a core IEF Function or as part of the users' system administration capability. The latter must interface with a data messaging PEP connected to the SMB.
PAP-PEP Type: Class	The PAP-PEP must provide features that integrate a user-specified administration system (or Policy Administration Point) with the users' security infrastructure. This administration system would provide the features described in Clause 8.
PDP Type: Class	The Policy Decision Point provides access control services (e.g., RBAC, ABAC, PBAC, or UBAC) to adjudicate data access to or release of resources to a specified user based on: 1. The sensitivity of the resource; 2. The clearances and attributes for each user; 3. (Optional) The operational context (e.g., location, device, mission, threat environment, and role) and

Table 6 - IEF Component Specializations Elements	
Element Name	**Element Descriptions**
	4. (Optional) Other security and data protection considerations of the user. The PDP may provide access control decision logic as native services within the PDP or as an interface to a user-specified decision system.
PDP-PEP Type: Class	This PEP (/proxy) connects the IEF Services to access control adjudication services provided by the user or the local infrastructure (e.g., Cloud Service Provider (CSP)). The PDP-PEP employs external services (e.g., RBAC, ABAC, PBAC, or UBAC) to adjudicate access to or release of resources to a specified user. (Clause 9)
PEP Type: Class	The Policy Enforcement Point (PEP) intercepts all user and system requests to access or release data elements protected by IEF services. The PEP gathers information about the data elements, sender, recipients, and communication channel, packages an adjudication request to the PDP, and enforces the PDP access or release determination. In a zero-trust environment, the PEP interface authenticates and authorizes each transaction with an external cryptographic service. The PEP must provide one or both of the feature sets defined by the SMB or Direct Connect options.
PPS Type: Class	The Policy-based Packaging and Processing Service (PPS) transitions structured Information Elements (e.g., NIEM, EDXL, and HL7) between data stores and information exchange services following local information sharing, safeguarding, and DCS policies. The IEF PPS enforces policies that conform to the specification of the Information Exchange Packaging Policy Vocabulary (IEPPV). The PPS allows users to selectively package (aggregate, transform, mark, filter, structure, and format) information elements for publication to authorized recipients. It also allows processing (parsing, transforming, mapping, and Marshalling) structured data and integrating the data elements into user-specified data stores.
SSG Type: Class	The Security Services Gateway provides a generalized integration point between IEF Components and user-specified security services, including ICAM, ABAC, Key Management, Cryptography, SIEM, SOAR, and CDM. The SSG conforms to ZTA principles. The following features apply to each combination of the IEF component and security system (/service), requiring implementations tailored to the combination or services. The SSG may be implemented as a single integration point for all security systems and services or as an integration point tailored to each user-specified security system (/service) API.

`

Table 6 - IEF Component Specializations Elements	
Element Name	**Element Descriptions**
SSG-PEP Type: Class	The SSG-PEP is a hybrid between the SSG and a PEP, providing ZT security elements to the interfaces between the IEF and the user's security infrastructure. It connects the IEF Services to Security services provided by the user or the local infrastructure (e.g., Cloud Service Provider (CSP)). It enables the IEF components to interoperate with user-specified security services. (e.g., Identity, credential, access management services, access control services, and essential management services) and infrastructure using a standardized SMB Interface. (Clause 12)
StructuredMessaging-PEP Type: Class	As a specialization of the PEP, the StructuredMessaging-PEP provides external interfaces between: 1. System or user application and a protected data store or 2. systems and applications. The StructuredMessaging-PEP acts as a proxy service that authenticates each user and authorizes the receipt or release of each data message. Clause 10.4 provides details on this interface.
TLS Type: Class	The Trusted Logging Service (TLS) provides logging services for all IEF components. In this configuration, the TLS provides logging as a native service within the IEF configuration.
TLS-PEP Type: Class	This PEP (/proxy) connects the IEF Services to logging services provided by the user or the local infrastructure (e.g., Cloud Service Provider (CSP)). The TLS-PEP is the integration point between IEF components and external logging services. (Clause 15)
UDP-PEP Type: Class	The DDS-PEP extends the Messaging PEP by providing the features, functions, and interfaces needed for the IEF to interoperate with UDP services.
WebServices-PEP Type: Class	The WebServices-PEP extends the Messaging PEP by providing the features, functions, and interfaces needed for the IEF to interoperate with Web services.

7.2.3 IEF Internal Interface

The following figure outlines the internal interface to the IEF Secure Message Bus (SMB). The user, implementor, or integrator determines the preferred messaging infrastructure.

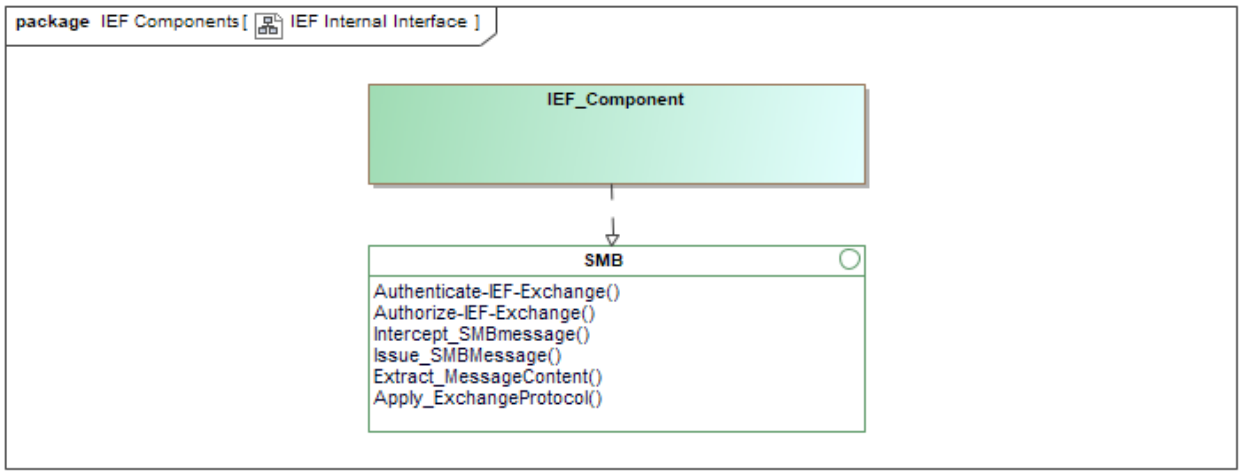


Figure 32 -IEF Internal Interface

The following table identifies and describes the elements and operations illustrated in the previous figure - IEF Internal Interface.

Table 7 - IEF Internal Interface Elements	
Element Name	**Element and Operation Descriptions**
SMB	The Secure Message Bus (SMB) interface is the standard communication path between IEF components. The IEF does not specify a platform-specific implementation of this component, enabling the implementor or integrator to select the technology that best suits the operational environment, performance requirements, and data sensitivity. Within the architecture, we identify several PSM options (e.g., DDS, REST, and Containers). These options do not preclude the use of alternate technology options.
	Element Type: Interface Owned Operations: Authenticate-IEF-Exchange: The SMB Interface must provide features that authorize the receipt or release of a message over the SMB. If IEF components and user data are securely isolated from the operational network behind the PEP(s), authorization can be disabled to enhance performance. Authorize-IEF-Exchange: The SMB Interface must provide features that authorize the receipt or release of a message over the SMB. If IEF components and user data are securely isolated from the operational network behind the PEP(s), authorization features can be disabled to enhance performance. Intercept_SMBmessage: The SMB Interface must provide features that listen to the SMB and intercept messages directed at the component. The interface then parses and extracts data and metadata elements

`

Table 7 - IEF Internal Interface Elements	
Element Name	**Element and Operation Descriptions**
	from the message and passes them on to component services for processing. Alternatively, this operation can be a poling mechanism that requests messages from the resources API. Issue_SMBMessage: The SMB interface must provide features that apply the SMB Messaging protocol and write the formatted message to the SMB (middleware) queue, topic, or channel. Extract_MessageContent: The SMB-Interface must provide features that extract the content of the message from its messaging and network Protocol. Apply_ExchangeProtocol: The SMB Interface must provide features that bind the message headers and metadata within the messaging and network Protocol.

7.3 Basic Deployments

The following clauses outline the basic configuration of IEF services for deployment.

7.3.1 IEF Unstructured Data Deployment

For unstructured data types, the IEF specifies an interoperable set of services that are combined to:

1. Authenticate participants to a data exchange,
2. Authorize that the provider is authorized to release the data, and
3. Authorize the release of the data to specified recipients.
4. All data elements transiting through the IEF PEP must be labeled appropriately and have the labeling (metadata) bound to the data elements per the Secure Access Container, STANAG 4778 bindings, or IC-TDF. In most cases, the producers of the data object (file or message) label the data.
5. As illustrated (Figure 33), the IEF services operate between the client application (recipient) and the data server (provider). On receipt, the PEP will:
6. Authenticate the provider and recipients,
7. Extract the metadata from the exchange message or file,
8. Retrieve the provider and recipient (s) attributes (e.g., security level, role, mission affiliations, data authorizations),
9. Prepare an adjudication request for the PDP and
10. Enforce the PDP determination.

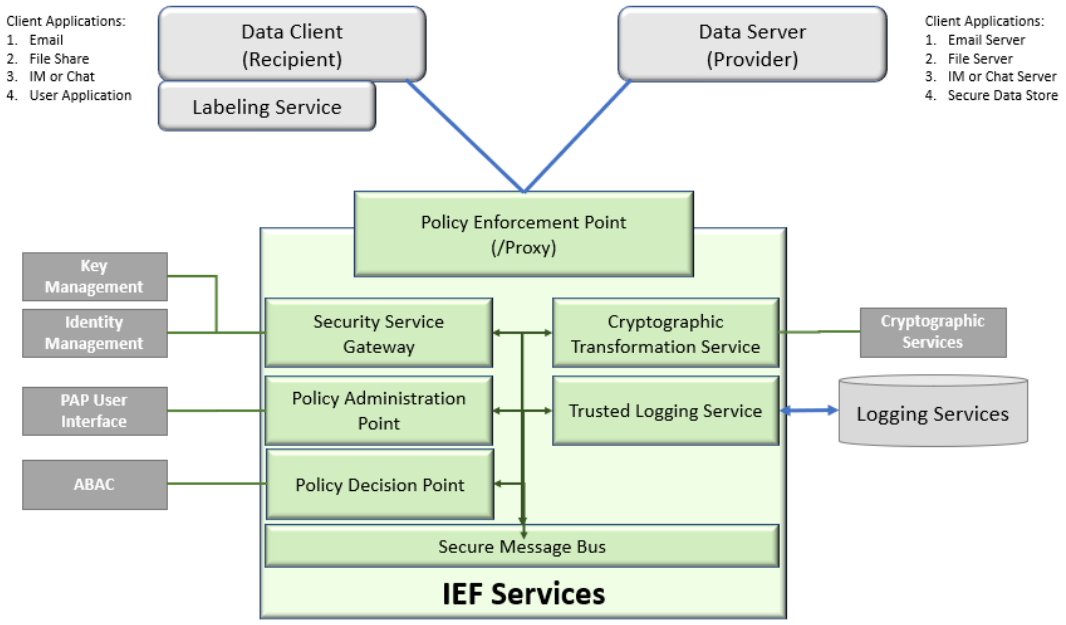


Figure 33 - IEF Deployment for Unstructured Data

7.3.2 IEF Structured Data Deployment

The IEF deployment for structured data or system-2-system environments employs the same components as for unstructured data, plus one, the Packaging and Processing Service (PPS). As illustrated (Figure 30), the IEF services operate between the client system, application, middleware, and device. However, in this case, the PPS selectively packages data tailored to the needs and authorizations of each recipient or community:

1. Process received data:
    a. Receive data from user systems, applications, and devices,
    b. Unpack and parse the data,
    c. Map data to the User's internal semantics,
    d. Marshal data to memory and persistent storage, and
    e. Trigger the requirement to package data for release per existing Information Exchange Specifications,
2. Package Data for release:
    a. Aggregate for release to the specified recipient(s),
    b. Transforms name-value pairs per agreed exchange specifications,
    c. Label data aggregates per business rules and the agreed exchange specifications,
    d. Redact data elements and aggregates exceeding the recipients' authorizations,
    e. Format the data for release, and
    f. Bind metadata per the agreed exchange specifications and
    g. Release the data to the specified application, systems, middleware, and device.

The PPS enforces user-specified semantic and exchange specifications defined using the IEPPV specification.

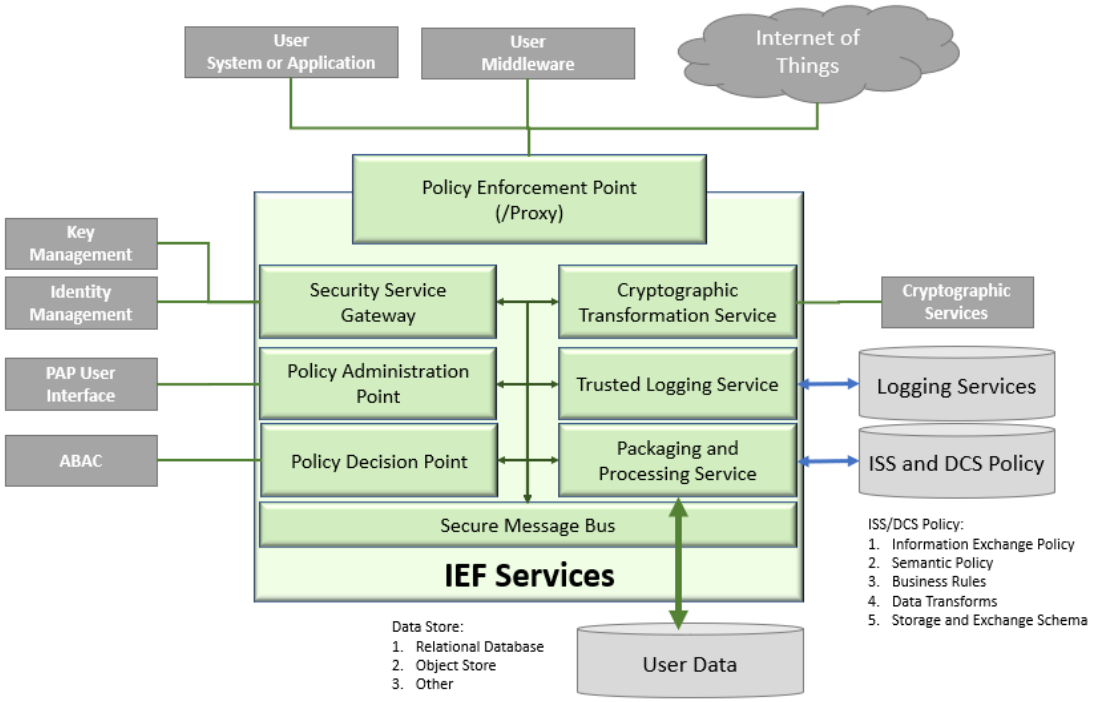


Figure 34 - IEF Deployment for Unstructured Data

7.4 Alternate Deployments

The following clauses outline acceptable alternatives to the configurations described above.

7.4.1 Direct PEP Integration

Direct PEP integration or connection enables the PEP to interface directly with existing security infrastructure rather than access these features through the SMB and the IEF-defined components (e.g., SSG, PDP, CTS, TLS). This configuration reduced areas of latency discovered in the implementation of IEF-RA Version 1.

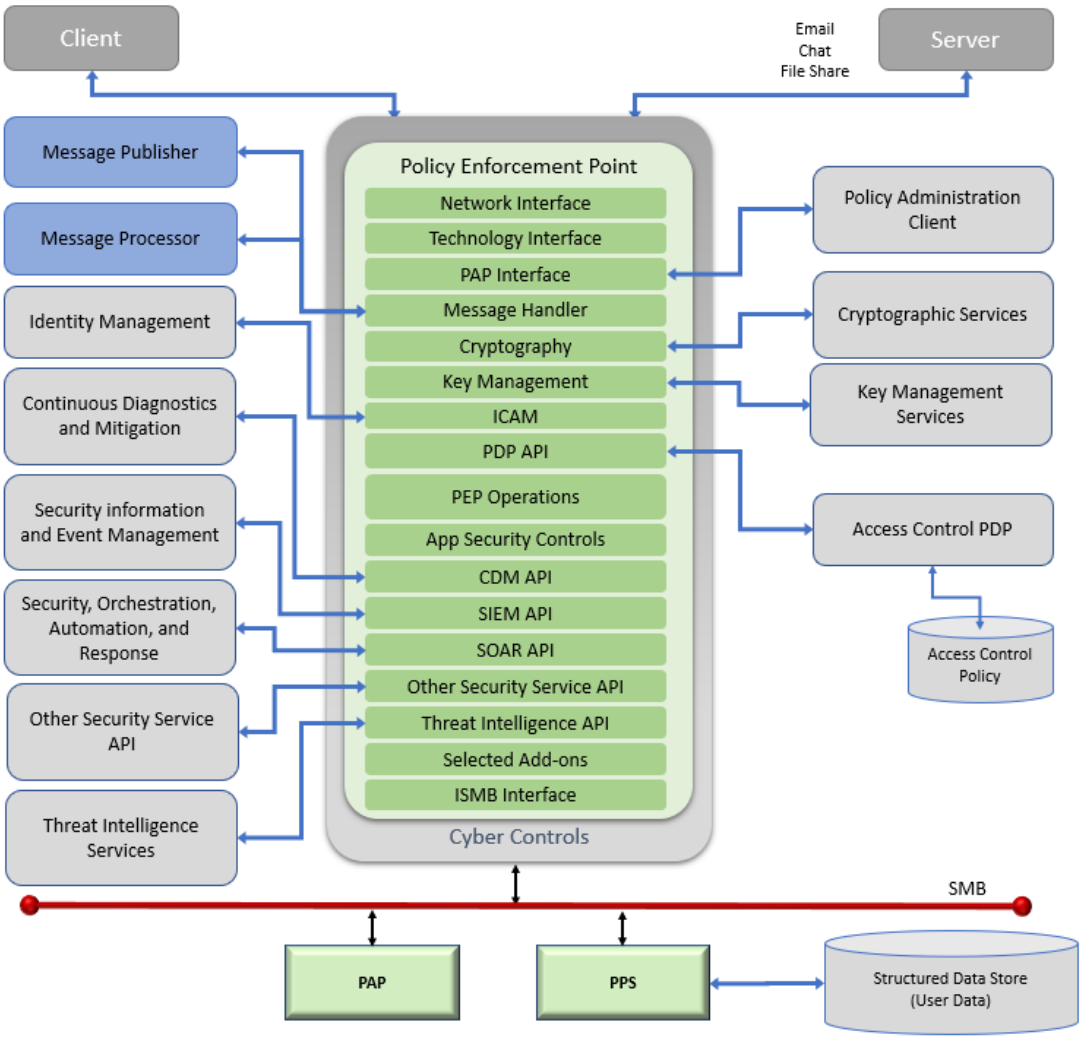


Figure 35 - Direct Connection of PEP with Security Services

Note that the SMB retains the interface between the PEP and some IEF components (e.g., PAP and PPS) in this configuration. The implementor or integrator determines which components the PEP accesses through the SMB or direct interfaces. All transactions must be authenticated and authorized by PEP functions in the PEP, SSG, CTS, PDP, and TLS.

7.4.2 External Policy Administration Point

During IEF and DCS testing, users were interested in integrating administration functions (Clause 8) into existing systems (e.g., system, network, or security administration solutions). Where required, the PAP functions were integrated through a PEP with specialized functions that translated administration messages into SMB formats and protocols.

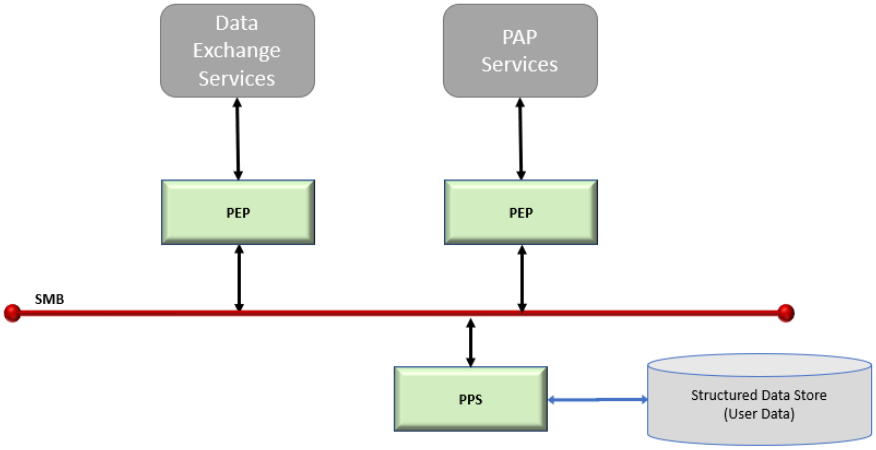


Figure 36 - External PAP

7.4.3 Secure Data Service (SDS)

The Secure Data Service configurations wrap the previously identified configurations in traditional virtual security controls.

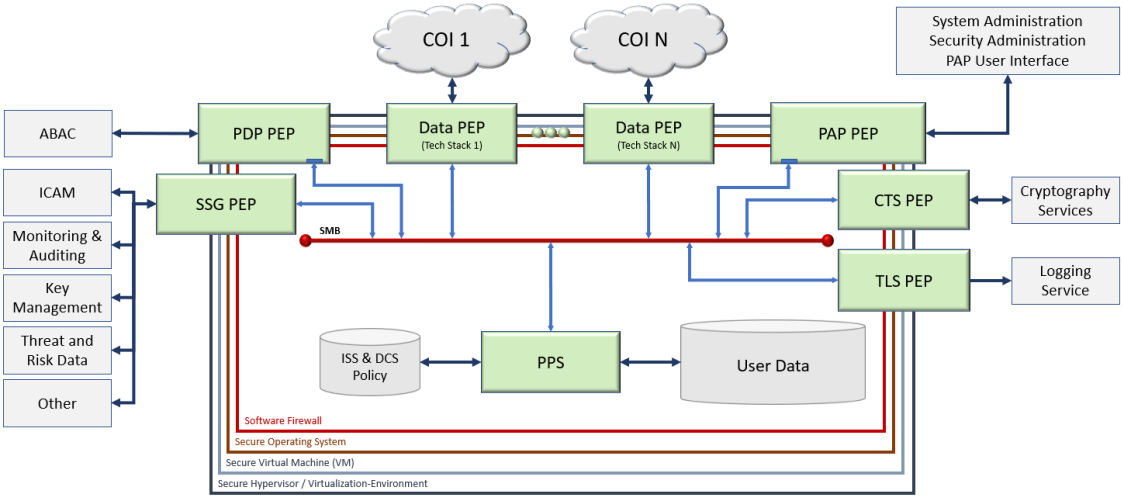


Figure 37 - SDS with SMB Connected Services

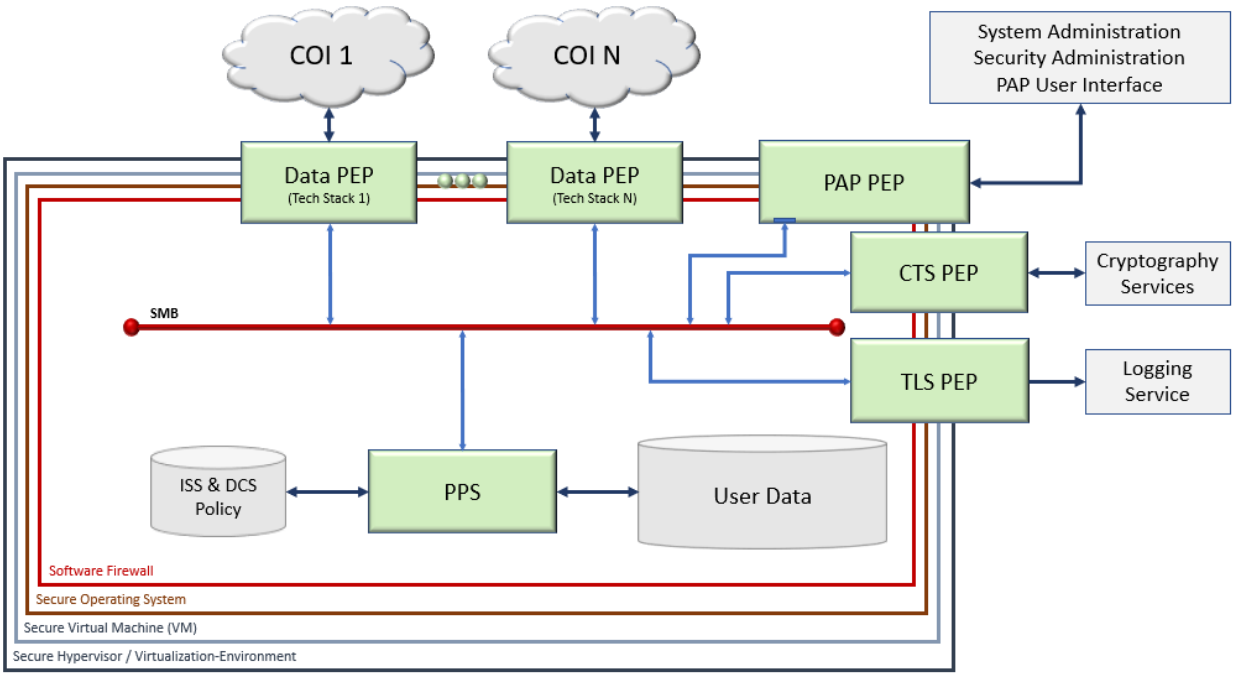


Figure 38 - SDS with Direct Connect PEP

7.5 IEF Component Core Functions

The following table briefly describes the core functions of the IEF Components.

Table 8 – Core Functions

IEF Service	Core Functions
Policy Enforcement Point (PEP)	The PEP: Intercepts all data exchanges between providers and consumers, Authenticates the producers and consumers to the exchange, Orchestrates the data collection needed to adjudicate access control adjudication for each data element, Issues the access control data to the PDP for adjudication, Enforces access control decisions rendered by the PDP (redacting unauthorized data elements where applicable), Translates message semantics between external data services and the SMB and Provides a standardized integration point for IEF services to interoperate with user infrastructure. Alternatives - The PEP can be configured to access supporting security services (Identity, Attribute Management, Access Control [e.g., ABAC], and logging) using: The SMB and IEF-defined services (e.g., SSG, CTS, PDP and TLS), The user-specified APIs to services within the security infrastructure or A user-approved combination of the above.

`

Table 8 – Core Functions	
IEF Service	Core Functions
Policy Decision Point (PDP)	The PDP: Receives access control requests from the PEP, Adjudicates: The provider's authorization to release the data and metadata elements, and Each recipient has the authorization to receive and use the data and metadata elements contained in the exchange, Returns access control decisions to the PEP, Translates messages between external services (e.g., ABAC) and the SMB and Provides a standardized integration point to access control rules engines (e.g., RBAC, ABAC, PBAC, UBAC). Alternatives: The PDP configurations: A PEP interface to user-specified services (e.g., RBAC, ABAC, PBAC, or UBAC) or An IEF service that delivers the PDP functionality described in Clause 9.
Policy Administration Point (PAP)	The PAP provides an authorized user with the interfaces and tools to manage and administer ISS, Data-Centric Security (DCS) and Access Control policies during operations, The PDP translates messages between external services and the SMB and Provides a standardized integration point that allows IEF services to interoperate with a user-specified PAP. Alternately: The PAP provides a PEP interface to user-specified services that provide the functionality above. The configuration or the PDP provides implementors and integrators with a standard integration point to IEF services.
Packaging and Processing Service (PPS)	The PPS: Enforces exchange and semantic policies expressed in IEPPV semantics for a specified data domain, Orchestrates: The processing (the receipt, decryption, parsing, transformation, and marshaling) of received data to a specified data store, The packaging (aggregation, transformation, labeling, redaction, formatting, and routing) of recipient-authorized data, and Routes the releasable data to the PEP for specified exchange service, The PPS acts as a mediation and interface service between user data stores containing structured and semi-structured data.

Table 8 – Core Functions

IEF Service	Core Functions
Trusted Logging Service (TLS)	The TLS: Provides access to a trusted data store (e.g., block-chain or hashed database) for IEF transaction reports and logs, Translates messages between SMB components and logging infrastructure and Provides a standardized integration point between IEF services and user infrastructure. Alternates: The TLS The TLS provides a PEP interface to user-specified services that provide the functionality above. The configuration or the TLS provides implementors and integrators with a standard integration point to IEF services.
Security Services Gateway (SSG)	The SSG: Provides a specialized PEP interface that enables IEF services to access user-specified security services through the SMB. Translates messages between security services and the SMB, Provides a standard interface for IEF services to acquire requisite security data and Provides a standardized integration point between IEF services and user infrastructure.
Cryptograghic Transformation Service (CTS)	The CTS provides cryptographic services for the PEP and PPS. The CTS provides a standardized integration point between IEF and cryptographic services. Alternatives: The CTS provides a PEP interface to user-specified services that provide the functionality above. The configuration or the CTC provides implementors and integrators with a standard integration point to IEF services.
Secure Message Bus (SMB)	The SMB provides secure and consistent messaging between the IEF services. Alternatives for SMB implementation include: DDS XMPP, and Container Infrastructure (e.g., Docker and Kubernetes) Note: See Clause 16 for the permitted message semantics for the SMB.

7.6 IEF Interfaces

In so much as the IEF leverages a set of user defined/supplied external services, the architecture defines a set of bridging components named after the external service to which they connect. These components provide an interface which the IEF uses to deliver it's internal message data to. The component then interfaces with the

`

external service after performing any data transformations required. The IEF only defines the internal facing interface and uses the naming convention suffix 'internal' to indicate the relative position within the design.

7.6.1 SMB Interfaces and Component Interactions

The following clauses identify the core interactions between the IEF components operating on the SMB.

Note: the implementor of the IEF components may extend the interactions described below to enhance performance, functionality, or security.

7.6.1.1 SMB Interfaces

The SMB enables secure communications between IEF components using the standard messages described in Clause 16. The following figure identifies the IEF component interfaces to the SMB—the clause corresponding to the component that realizes each interface provides the details on the interface.

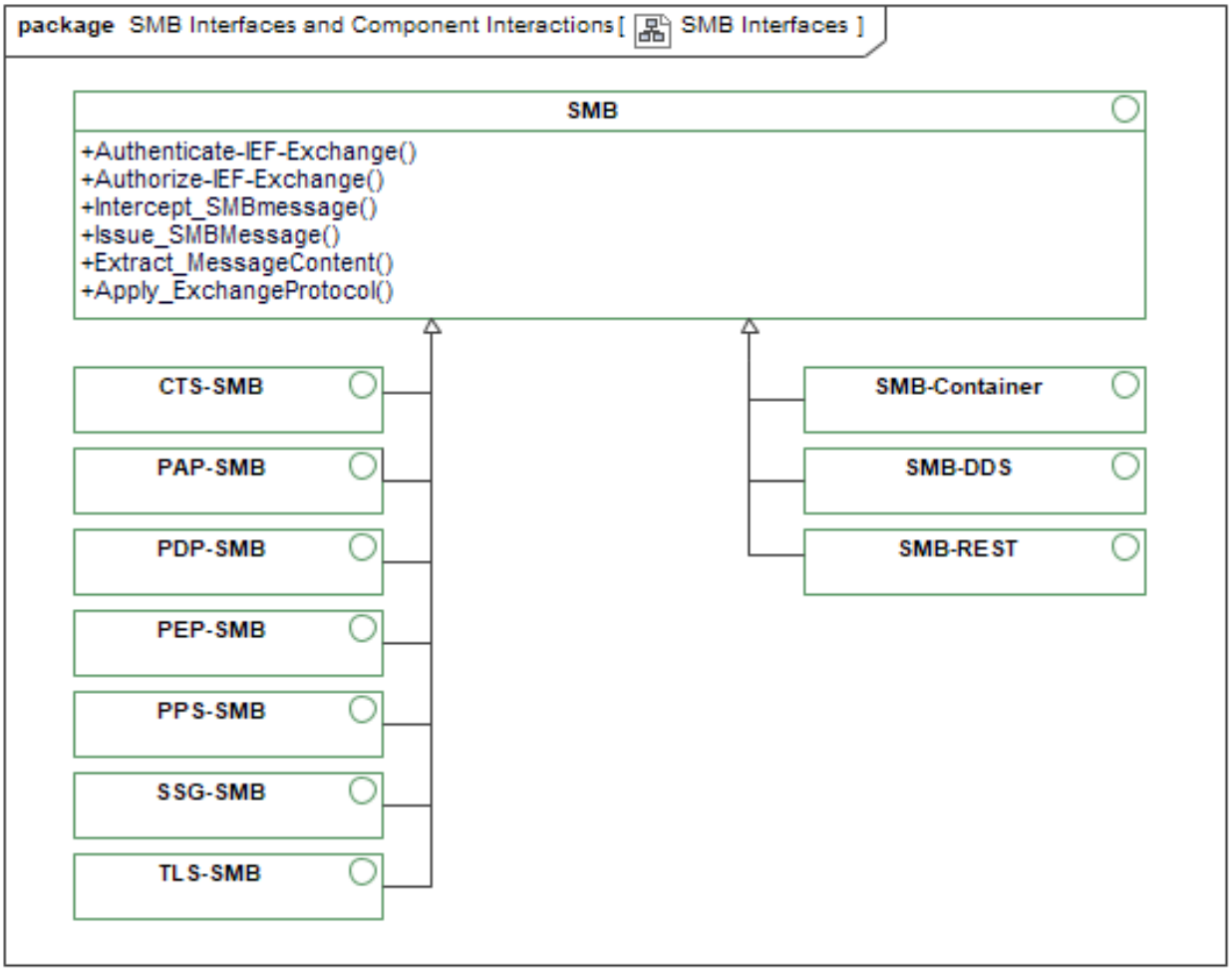


Figure 39 -SMB Interfaces

The following table describes the elements illustrated in the previous figure - SMB Interfaces.

Table 9 - SMB Interfaces Elements	
Element Name	**Element Descriptions**
CTS-SMB Type: Interface	The CTS-SMB interface enables the CTS to interoperate with other IEF components (e.g., PAP, PEP, PPS, SSG, PDP, TLS, and CTS) in the configuration. In a Zero-Trust implementation, it is responsible for enforcing access and release control policy between IEF components. Messages (Clause 16) applicable to the CTS-SMB interface include: • PAP-Command (receive); • PAP-Command-response (send); • CTS-Request (receive); • CTS-Response (send); and • CTS-LogMessage (send).
PAP-SMB Type: Interface	The SMB interface provides features that enable the PAP to interoperate with other IEF components (e.g., PEP, PPS, SSG, PDP, TLS, and CTS) in the configuration. In a Zero-Trust implementation, it is responsible for enforcing access and release control policy between IEF components. Messages (Clause 16) applicable to the PAP-SMB interface include: • PAP-Command (send); • PAP-CommandResponse (receive); • PAP-AlertWarning (receive); • PDP-OperationAuthorizationRequest (send); • PDP-OperationAuthorizationResponse (receive); • SSG-Request (send); • SSG-Response (receive); and • TLS-LogMessage (send). Refer to Clause 7 for additional details on PAP Interfaces.
PDP-SMB Type: Interface	SMB interface enables the PDP to interoperate with other IEF components (e.g., PEP, PPS, SSG, TLS, and CTS) in the configuration. In a Zero-Trust implementation, it is responsible for enforcing access and release control policy between IEF components. Messages (Clause 16) applicable to the PDP-SMB interface include: 1. PDP-Request (receive); 2. PDP-Response (send); 3. PAP-Command (receive); 4. PAP-CommandResponse (send); 5. PAP-AlertWarning (send);

`

Table 9 - SMB Interfaces Elements	
Element Name	**Element Descriptions**
	6. SSG-Request (send); 7. SSG-Response (receive) and 8. TLS-LogMessage (send).
PEP-SMB Type: Interface	The SMB interface enables the PEP to interoperate with other IEF components (e.g., PEP, PPS, SSG, PDP, TLS, and CTS) in the configuration. In a Zero-Trust implementation, it is responsible for enforcing access and release control policy between IEF components. Messages (Clause 16) applicable to the PEP-SMB interface include: • PAP-Command (receive); • PAP-CommandResponse (send); • PAP-AlertWarning (send); • PDP-Request (send); • PDP-Response (receive); • PPS-Publish (receive); • PPS-Receive (send); • PPS-Request (send); • SSG-Request (send); • SSG-Response (receive); • CTS-Request (send); • CTS-Response (receive); and • TLS-LogMessage (send).
PPS-SMB Type: Interface	SMB interface enables the PPS to interoperate with other IEF components (e.g., PEP, SSG, PDP, TLS, and CTS) in the configuration. In a Zero-Trust implementation, it is responsible for enforcing access and release control policy between IEF components. Messages (Clause 16) applicable to the PPS-SMB interface include: • PPS-Receive (receive); • PPS-Publish (send); • PPS-Request (receive); • PAP-Command (receive); • PAP-CommandResponse;

Table 9 - SMB Interfaces Elements	
Element Name	**Element Descriptions**
	• PAP-AlertWarning (send); • SSG-Request (send); • SSG-Response (receive); • CTS-Request (send); • CTS-Response (receive); and • TLS-LogMessage (send).
SMB Type: Interface	The Secure Message Bus (SMB) interface is the standard communication path between IEF components. The IEF does not specify a platform-specific implementation of this component, enabling the implementor or integrator to select the technology that best suits the operational environment, performance requirements, and data sensitivity. Within the architecture, we identify several PSM options (e.g., DDS, REST, and Containers). These options do not preclude the use of alternate technology options.
SMB-Container Type: Interface	The SMB-Container interface enables the deployment of IEF components as containerized workloads using Kubernetes, Docker, and other alternatives.
SMB-DDS Type: Interface	An SMB-DDS Interface enables the deployment of IEF components using DDS services (e.g., Readers and Writers).
SMB-REST Type: Interface	An SMB-REST interface enables the deployment of IEF components using REST services.
SSG-SMB Type: Interface	SMB interface enables the SSG to interoperate with other IEF components (e.g., PEP, PPS, PAP, PDP, TLS, and CTS) in the configuration. In a Zero-Trust implementation, it is responsible for enforcing access and release control policy between IEF components. Messages (Clause 16) applicable to the SSG-SMB interface include: • SSG-Request (receive); • SSG-Response (send); • PAP-Command(receive); • PAP-CommandResponse (send); • PAP-AlertWarning (send); and • TLS-LogMessage (send).

`

Table 9 - SMB Interfaces Elements	
Element Name	**Element Descriptions**
TLS-SMB Type: Interface	SMB interface enables the TLS to interoperate with other IEF components (e.g., PEP, SSG, PDP, PPS, and CTS) in the configuration. In a Zero-Trust implementation, it is responsible for enforcing access and release control policy between IEF components. Messages (Clause 16) applicable to the TLS-SMB interface include: • TLS-LogMessage (receive); • PAP-Command (receive); and • PAP-CommandResponse (send).

7.6.1.2 SMB Component Interface Interactions

The following figure illustrates the dependencies between the core IEF components. Each IEF component uses the IEF Secure Messaging Bus (SMB) to communicate with and obtain services from other IEF components. The SMB 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;
- SSG-Request;
- SSG-Response;
- CTS-Request;
- CTS-Response; and
- TLS Log Entry.

Clause 16 and Annex A provide content descriptions for each message.

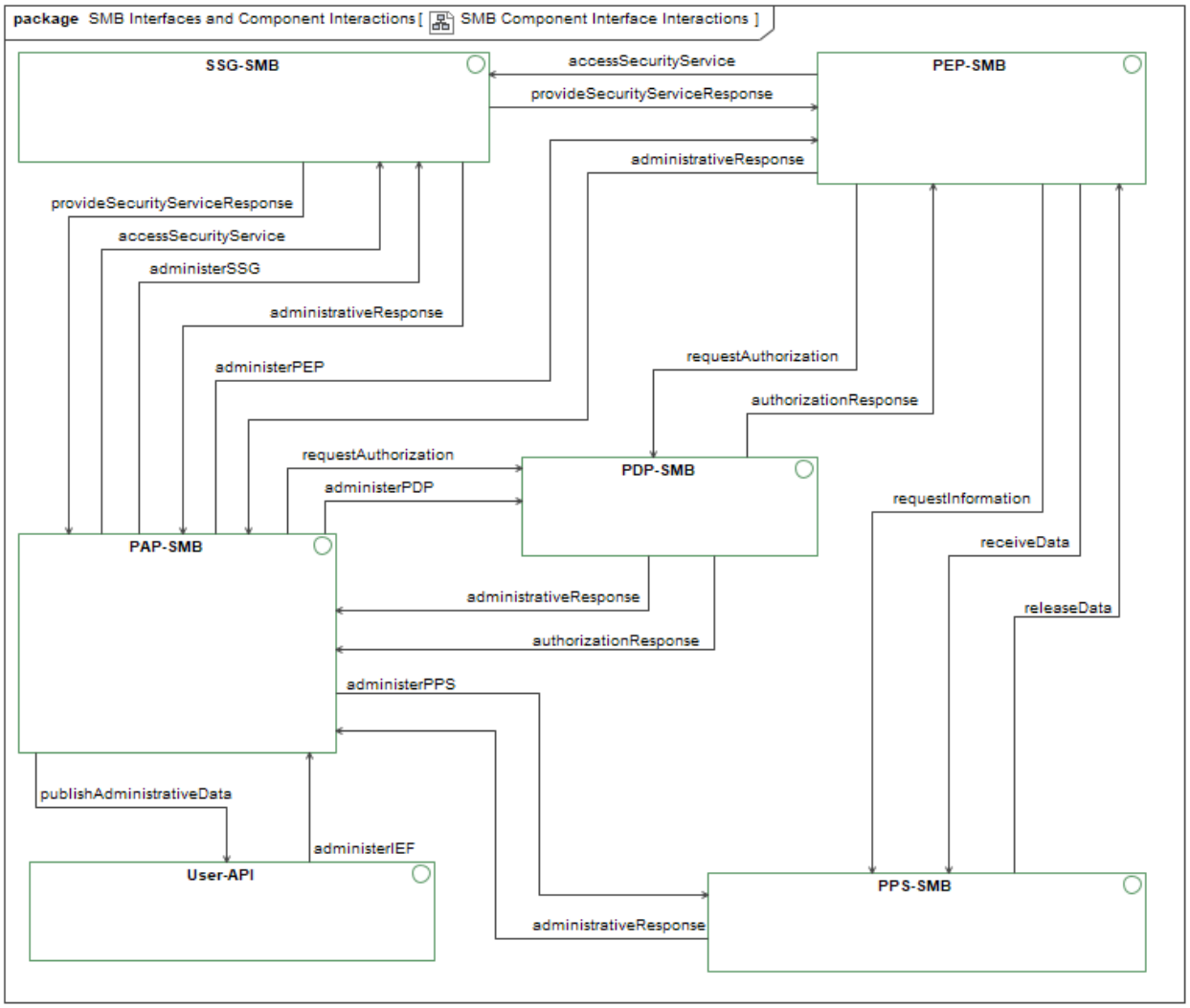


Figure 40 -SMB Component Interface Interactions

The following table describes the interfaces in the previous figure - SMB Component Interface Interactions.

Table 10 – SMB Component Interface Interactions Interfaces

Element	Interface Interactions
PAP-SMB Type: Interface	The PAP-SMB enables the component to access the following services needed to perform its assigned function: accessSecurityService: The PAP uses an SSG-Request Message to the SSG to retrieve data from the user's security services (/infrastructure). The SSG translates the request into a message to a specific security service. administerSSG: The PAP uses the PAP-Command Message to direct an SSG to alter its operating characteristics or configuration. administerPEP:

`

Table 10 – SMB Component Interface Interactions Interfaces	
Element	Interface Interactions
	The PAP uses the PAP-Command Message to direct a PEP to alter its operating characteristics or configuration. administerPDP: The PAP uses the PAP-Command Message to direct a PDP to alter its operating characteristics or configuration. requestAuthorization: The PAP issues a PDP-Request to the PDP to obtain authorization for User commands for IEF components. administerPPS: The PAP uses the PAP-Command Message to direct a PPS to alter its operating characteristics or configuration. publishAdministrativeData: The PAP packages administrative data from IEF components and then provisions it for viewing on the user interface or application.
PDP-SMB Type: Interface	The PDP-SMB enables the component to access the following services needed to perform its assigned function: administrativeResponse: Upon completing a PAP-directed administrative action, the PDP packages a PAP-CommandResponse and issues it to the PAP. authorizationResponse: Upon completion of the adjudication process. the PDP packages and issues a PDP-Response message and issues it to the PEP. authorizationResponse: The PDP adjudicates the authorization request using user-specified policies, then packages its determination as a PDP-Response message and issues it to the PAP.
PEP-SMB Type: Interface	The PEP-SMB enables the component to access the following services needed to perform its assigned function: accessSecurityService: The PEP issues an SSG-Request to retrieve data (e.g., user authentication and attributes) from the user's security services (/infrastructure). The SSG translates the request into the message protocol expected by the specific security service. requestInformation: The PEP receives an information request message and verifies that the user is authorized to receive the information. If authorized, the PEP issues a PPS-Request to the PPS. Upon completing the request, the PPS packages the requested

Table 10 – SMB Component Interface Interactions Interfaces	
Element	Interface Interactions
	information as a PPS-Publish and issues it to the PEP for dissemination. requestAuthorization: The PAP issues a PDP-Request to the PDP to obtain authorization: • To a user's request for data; • For the user to release specified data elements and • To release the specified data elements to specified users. receiveData: The PEP receives information from the users' exchange services or applications, extracts the metadata elements, and verifies that the local PPS is authorized to ingest the message content. If authorized, the PEP packages the message and the metadata as a PPS-Receive message and issues it to the PPS for processing. administrativeResponse: Upon completion of the PAP-directed administrative action, the PEP packages a PAP-CommandResponse message and issues it to the PAP.
PPS-SMB Type: Interface	The PPS-SMB enables the component to access the following services needed to perform its assigned function: releaseData: The PPS packages and issues PPS-Publish messages containing information exchange messages (/payloads) and message metadata for release to the recipient. The PEP validates and verifies that each recipient's information is relevant and authorized. administrativeResponse: Upon completing a PAP-directed administrative action, the PPS packages a PAP-CommandResponse and issues it to the PAP.
SSG-SMB Type: Interface	The SSG-SMB enables the component to access the following services needed to perform its assigned function: provideSecurityServiceResponse: Upon receipt of the data from the Security service, the SSG translates it into an SSG-Response message and issues it to the PAP for processing. administrativeResponse: Upon completion of a PAP-directed administrative action, the SSG packages a PAP-CommandResponse message and issues it to the PAP. provideSecurityServiceResponse: Upon receipt of requested data from a security service, the SSG translates it into an SSG-Response and issues it to the PEP.

`

Table 10 – SMB Component Interface Interactions Interfaces	
Element	Interface Interactions
User-API Type: Interface	The User-API enables the component to access the following services needed to perform its assigned function: administerIEF: This interface represents an integration point for implementors to develop an administrator interface that suits their operations and users. The interface receives messages from a user application that directs the PAP to authorize changes to IEF components.

7.6.1.3 Additional SMB Component Interface Interactions

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

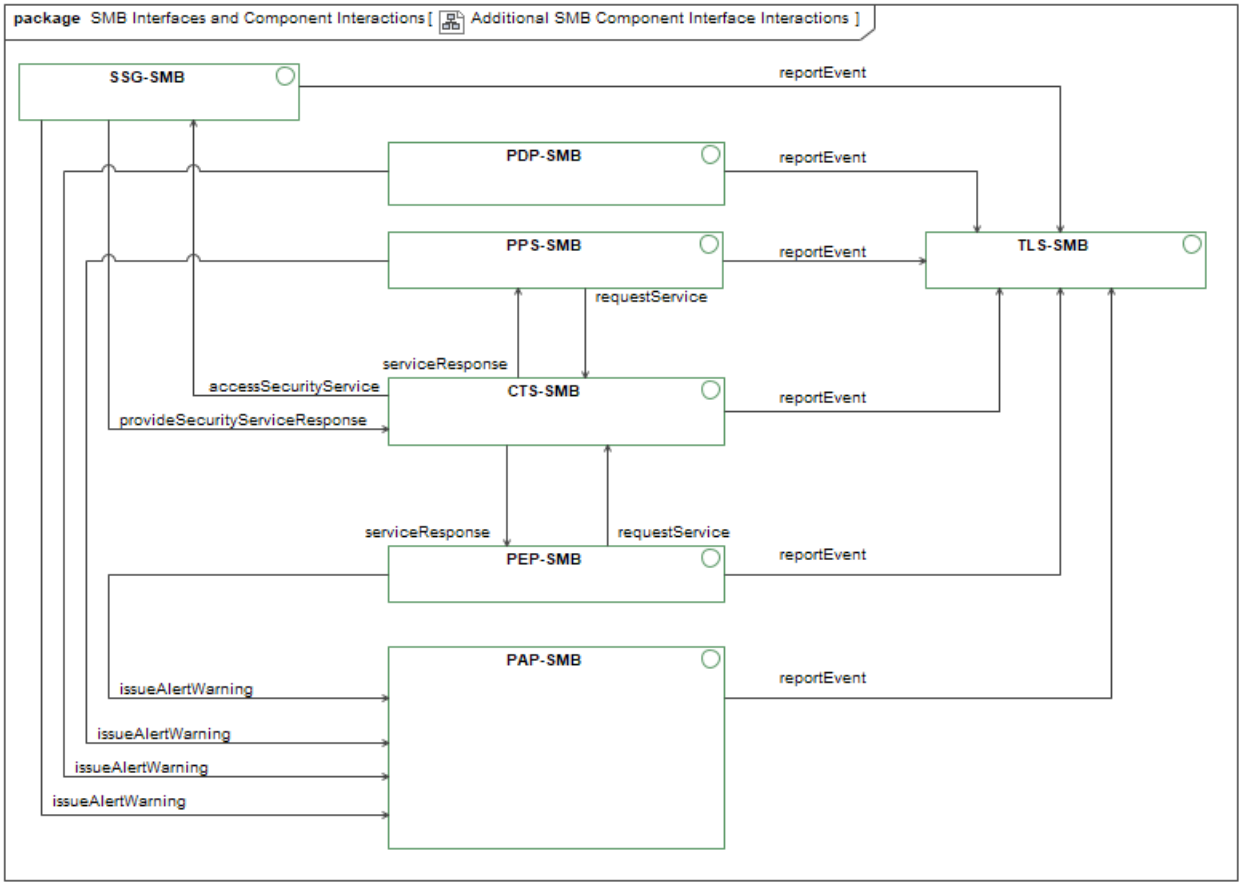


Figure 41 -Additional SMB Component Interface Interactions

The following table describes the interfaces in the previous figure - Additional SMB Component Interface Interactions.

Table 11 – Additional SMB Component Interface Interactions Interfaces

Element	Interface Interactions
CTS-SMB Type: Interface	The CTS-SMB enables the component to access the following services needed to perform its assigned function: reportEvent: The CTS uses a TLS-LogMessage to report events/transactions, alerts, and warnings to the Trusted Logging Service. serviceResponse: Upon completing the PEP requested action, the CTS uses a CTS-Response message to provide the encrypted or decrypted objects to the PEP. serviceResponse: Upon completing the PPS requested action, the CTS uses a CTS-Response message to provide the encrypted or decrypted objects to the PPS. accessSecurityService: The CTS issues an SSG-Request to retrieve data (e.g., crypto keys) from the user's security services (/infrastructure). The SSG translates the request into the message protocol expected by the specific security service.
PAP-SMB Type: Interface	The PAP-SMB enables the component to access the following services needed to perform its assigned function: reportEvent: The PAP uses a TLS-LogMessage to report events/transactions, alerts, and warnings to the Trusted Logging Service.
PDP-SMB Type: Interface	The PDP-SMB enables the component to access the following services needed to perform its assigned function: reportEvent: The PDP uses a TLS-LogMessage to report events/transactions, alerts, and warnings to the Trusted Logging Service. issueAlertWarning: The PDP uses PAP-AlertWarning to issue alerts, warnings, and error messages to the PAP in the event of unauthorized activities or requests.
PEP-SMB Type: Interface	The PEP-SMB enables the component to access the following services needed to perform its assigned function: reportEvent:

`

Table 11 – Additional SMB Component Interface Interactions Interfaces	
Element	Interface Interactions
	The PEP uses a TLS-LogMessage to report events/transactions, alerts, and warnings to the Trusted Logging Service. issueAlertWarning: The PEP uses PAP-AlertWarning to issue alerts, warnings, and error messages to the PAP in the event of unauthorized activities or requests. requestService: The PEP uses a CTS-Request message to request the encryption or decryption of data elements.
PPS-SMB Type: Interface	The PPS-SMB enables the component to access the following services needed to perform its assigned function: reportEvent: The PPS uses a TLS-LogMessage to report events/transactions, alerts, and warnings to the Trusted Logging Service. issueAlertWarning: The PPS uses PAP-AlertWarning to issue alerts, warnings, and error messages to the PAP in the event of unauthorized activities or requests. requestService: The PPS sends a CTS-Request message to the CTS to encrypt or decrypt data elements.
SSG-SMB Type: Interface	The SSG-SMB enables the component to access the following services needed to perform its assigned function: reportEvent: The SSG uses a TLS-LogMessage to report events/transactions, alerts, and warnings to the Trusted Logging Service. issueAlertWarning: The SSG uses PAP-AlertWarning to issue alerts, warnings, and error messages to the PAP in the event of unauthorized activities or requests. provideSecurityServiceResponse: Upon receipt of requested data from a security service, the SSG translates it into an SSG-Response and issues it to the CTS.
TLS-SMB Type: Interface	The TLS-SMB 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.

Table 11 – Additional SMB Component Interface Interactions Interfaces	
Element	Interface Interactions

7.6.2 PEP External Interfaces and Integration Points

The following classes outline the IEF interfaces to external systems and Infrastructure.

7.6.2.1 PEP External Security Infrastructure Interfaces

The PEP allows access to external interfaces to the user security services, as illustrated by the DirectConnect-PEP specialization. Using the SMB-PEP configuration, the PEP accesses the security service using the SMB interface to the SSG, PDP, CTS, and TLS attached to the SMB. Clause 9 provides interface details.

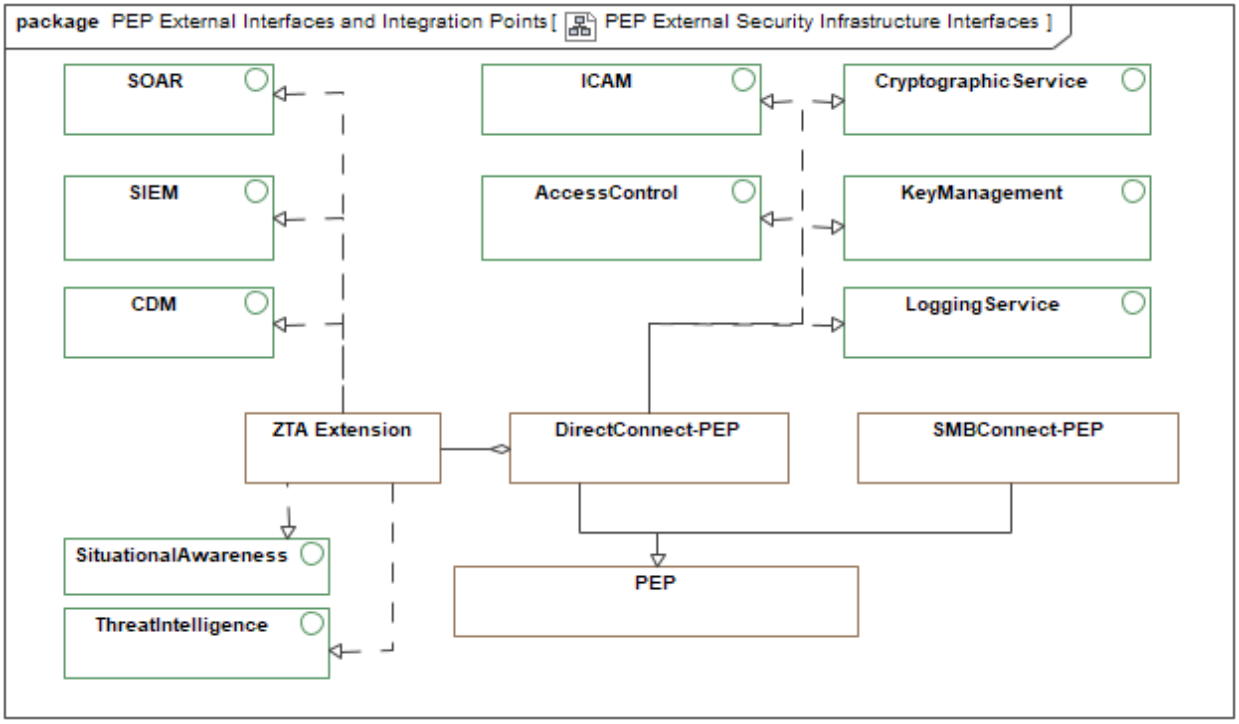


Figure 42 -PEP External Security Infrastructure Interfaces

The following table describes the elements illustrated in the previous figure - PEP External Security Infrastructure Interfaces.

Table 12 - PEP External Security Infrastructure Interfaces Elements	
Element Name	**Element Descriptions**
AccessControl Type: Interface	This PDP interface must provide features that enable the PEP to interoperate with user-specified Access Control and adjudication services, e.g.: • Attribute Based Access Control (ABAC);

`

Table 12 - PEP External Security Infrastructure Interfaces Elements	
Element Name	**Element Descriptions**
	• Role Based Access Control (RBAC); • Policy Based-Access Control (PBAC) and • User-Based Access Control (UBAC). The PEP prepares an adjudication request to the access control system (e.g., ABAC) and enforces its response. The requirements and functions provided by this interface are left to the user (/implementer/integrator) to define based on the capabilities and API of the specified Access Control (PDP) system.
CDM Type: Interface	This optional interface enables the PEP to interoperate with the user-specified Continuous Diagnostic Monitoring (CDM) systems. It aligns the IEF PEP with ZTA requirements. The requirements and functions provided by this interface are left to the user (/implementer/integrator) to define based on the capabilities and API of the specified CDM system.
CryptographicService Type: Interface	The cryptographic service interface provides features that enable the PEP or CTS to interoperate with user-specified cryptographic services. The requirements and functions provided by this interface are left to the user (/implementer/integrator) to define based on the capabilities and API of the specified cryptographic system.
ICAM Type: Interface	This optional (see notes below) interface enables the PEP to interoperate with the user-specified Identity Credential and Access Management (ICAM) system. It also provides functions to authenticate users of IEF services and request and receive attributes (metadata) about a user (system/application/device) seeking to access IEF components, services, and data. The PEP may request user attributes, including: • Security Level; • Role; • Operational Associations; • Data attributes; • Equipment being used; and • Location. The requirements and functions provided by this interface are left to the user (/implementer/integrator) to define based on the capabilities and API of the specified ICAM system.
KeyManagement Type: Interface	The SSG (or PEP) may provide interfaces that enable IEF components to interoperate with user-specified policy Key Management Services (KMS), which manage (e.g., generate, escrow, and provision) the cryptographic keys.

Table 12 - PEP External Security Infrastructure Interfaces Elements	
Element Name	**Element Descriptions**
LoggingService Type: Interface	This TLS interface (see notes below) must provide features that enable the PEP or TLS to interoperate with the user-specified log management service (/system). The requirements and functions provided by this interface are left to the user (/implementer/integrator) to define based on the capabilities and API of the specified logging system.
SIEM Type: Interface	This optional interface enables the PEP to interoperate with the user-specified Security Information and Event Management (SIEM) system. IT aligns the IEF PEP with ZTA requirements. The requirements and functions provided by this interface are left to the user (/implementer/integrator) to define based on the capabilities and API of the specified SIEM system.
SituationalAwareness Type: Interface	The SSG may provide an interface that enables IEF components to interoperate with user-specified cyber and operational situational awareness (SA) systems to inform policy decisions.
SOAR Type: Interface	This optional interface enables the PEP to interoperate with the user-specified Security Orchestration, Automation, and Response (SOAR) system. It aligns the IEF PEP with ZTA requirements. The requirements and functions provided by this interface are left to the user (/implementer/integrator) to define based on the capabilities and API of the specified SOAR system.
ThreatIntelligence Type: Interface	The SSG may provide an interface that enables IEF components to interoperate with user-specified cyber and operational threat-risk intelligence systems to inform policy decisions.

7.6.2.2 Email Interfaces

As a specialization of the PEP, the Email-PEP provides two external interfaces:

- An interface to the email client; and
- An interface to the email server.

The Emails-PEP acts as a proxy service that authenticates each user and authorizes the receipt or release of:

- Each email (body and attachments) posted to the Email-Server from an Email-Client; and
- Each email (body and attachments) that the email client receives.

Clause 10.2 provides details on this interface.

`

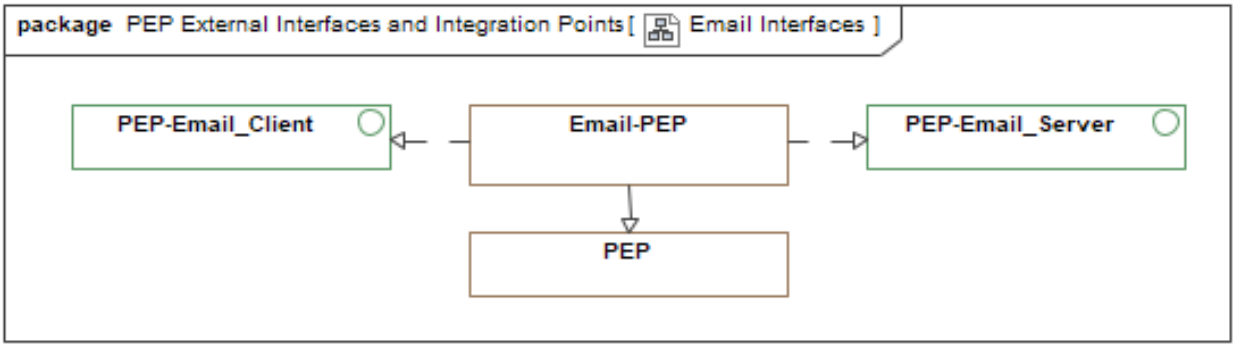


Figure 43 -Email Interfaces

The following table describes the elements illustrated in the previous figure - Email Interfaces.

Table 13 - Email Interfaces Elements	
Element Name	**Element Descriptions**
PEP-Email_Client Type: Interface	This interface enables the PEP to interoperate with the user-specified email client. The implementor or integrator tailors this interface to the API of the user-specified email client. Clause 10.1 provides details on this interface.
PEP-Email_Server Type: Interface	This interface enables the PEP to interoperate with the user-specified email server. The implementor or integrator tailors this interface to the API of the user-specified email server. Clause 10.1 provides details on this interface.

7.6.2.3 FileShare Interfaces

As a specialization of the PEP, the File-Share-PEP provides three external interfaces:

1. An interface to user applications requiring access to IEF-protected files;
2. An Interface to manage and administer files; and
3. An interface to the protected file server.

The FileShare-PEP is a proxy service that authenticates each user and authorizes the receipt or release of any file protected by an IEF implementation. Clause 10.2 provides details on this interface.

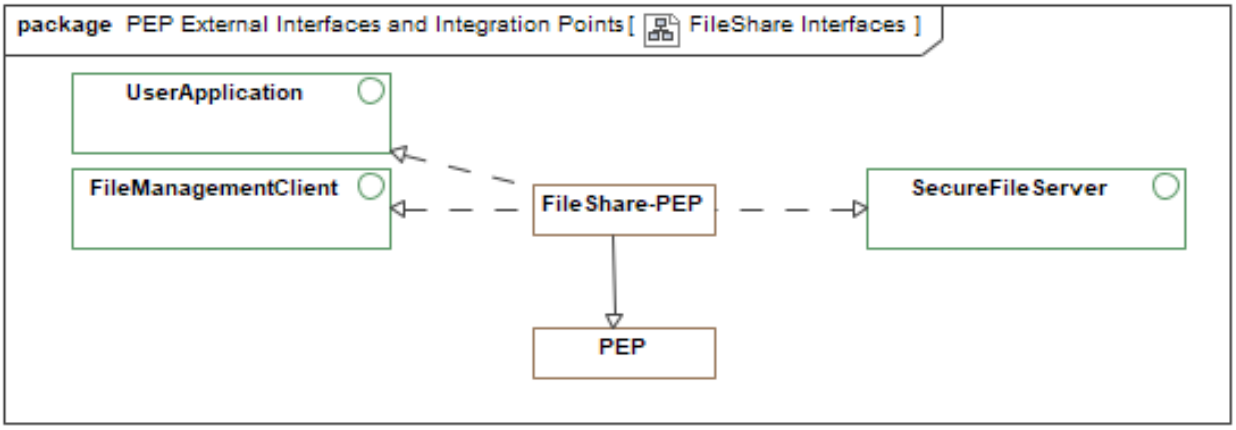


Figure 44 -FileShare Interfaces

The following table describes the elements illustrated in the previous figure - FileShare Interfaces.

Table 14 - FileShare Interfaces Elements	
Element Name	**Element Descriptions**
FileManagementClient Type: Interface	The file management client interface enables the PEP to interoperate with the user-specified file management client. The implementor or integrator tailors this interface to the API of the user-specified File Management client.
SecureFileServer Type: Interface	This interface enables the PEP to interoperate with the user-specified file server. The implementor or integrator tailors this interface to the API of the user-specified file server. Clause 10.2 provides details on this interface.
UserApplication Type: Interface	This user application interface enables the PEP to interoperate with the user-specified applications. The implementor or integrator tailors this interface to the API of the user-specified application. Clause 10.2 provides details on this interface.

7.6.2.4 Instant Messaging Interfaces

As a specialization of the PEP, the InstantMessaging-PEP or Chat-PEP provides two external interfaces:

1. An interface to the instant messaging or chat client and
2. An interface to the instant messaging or chat server.

The InstantMessaging-PEP is a proxy service that authenticates each user and authorizes the receipt or release of instant or chat messages. Clause 10.3 provides details on this PEP specialization.

`

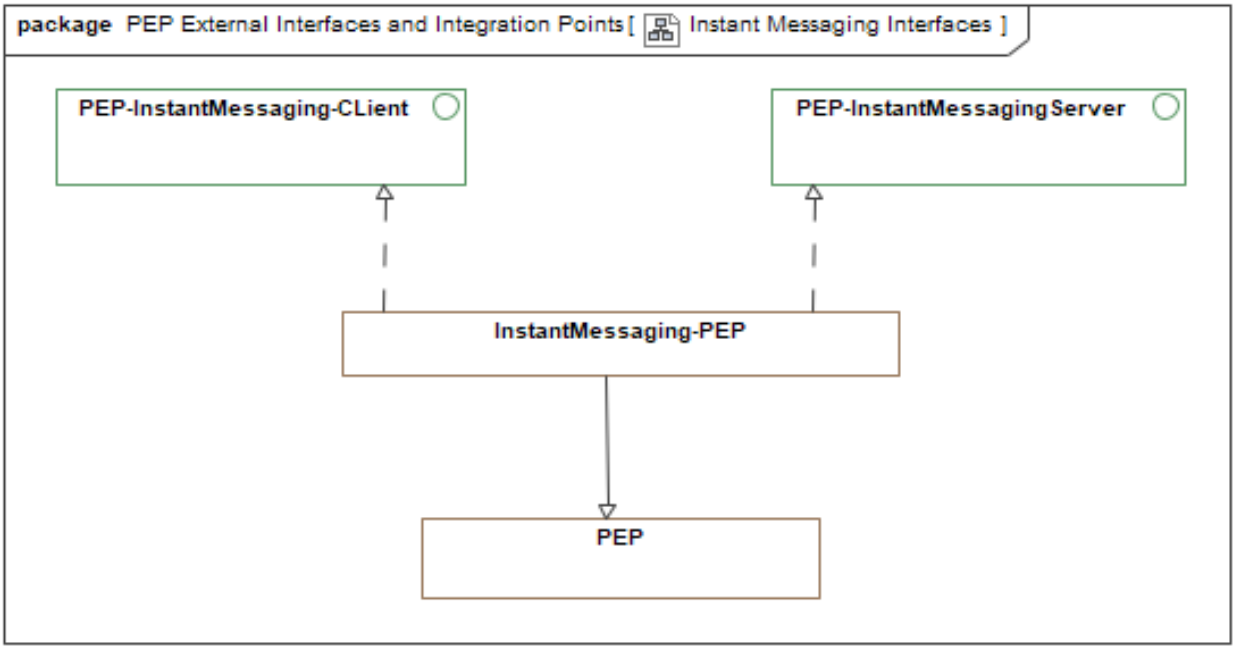


Figure 45 -Instant Messaging Interfaces

The following table describes the elements illustrated in the previous figure - Instant Messaging Interfaces.

Table 15 - Instant Messaging Interfaces Elements	
Element Name	**Element Descriptions**
PEP-InstantMessaging-CLient Type: Interface	This interface enables the PEP to interoperate with the user-specified instant messaging or chat client. Clause 10.3 provides details on this interface.
PEP-InstantMessagingServer Type: Interface	This interface enables the PEP to interoperate with the user-specified instant messaging or chat server. Clause 10.3 provides details on this interface.

7.6.2.5 Structured Messaging Interfaces

This interface specialization enables the StructuredMessaging-PEP to interoperate with User Applications in the environment.

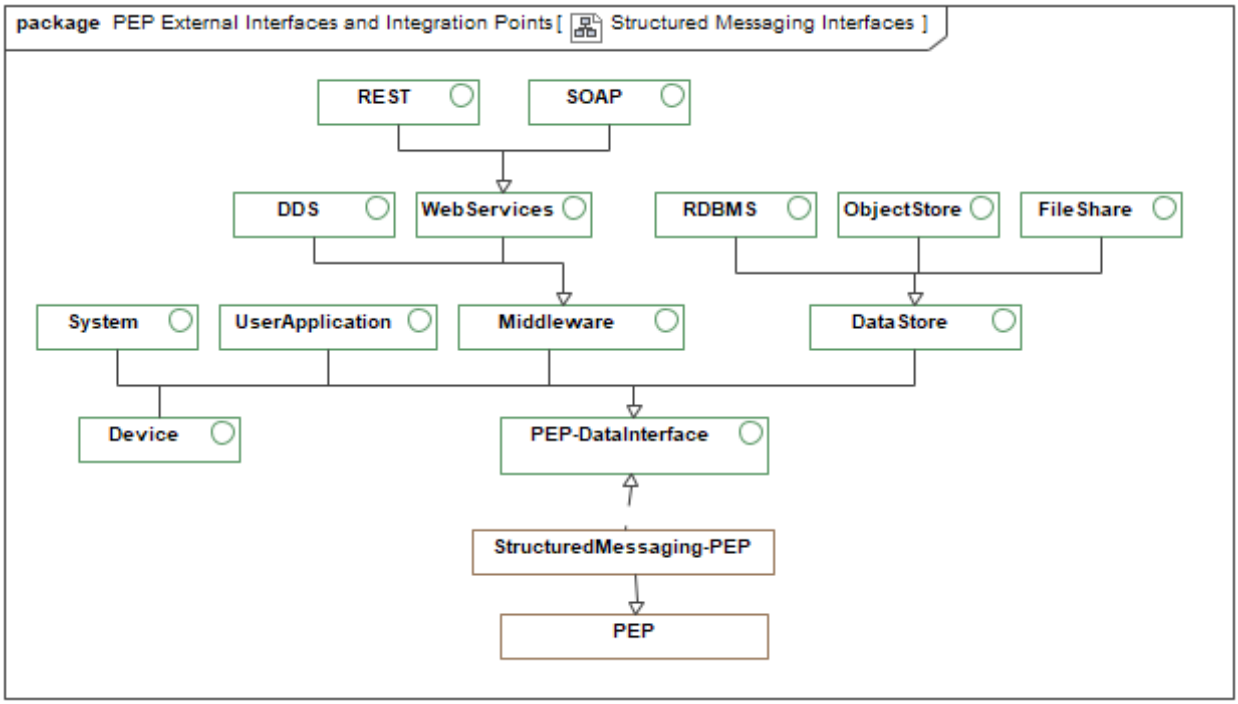


Figure 46 -Structured Messaging Interfaces

The following table describes the elements illustrated in the previous figure - Structured Messaging Interfaces.

Table 16 - Structured Messaging Interfaces Elements	
Element Name	**Element Descriptions**
DataStore Type: Interface	This data store interface specialization enables the StructuredMessaging-PEP to interoperate with data stores in the environment.
DDS Type: Interface	This DDS interface specialization enables the StructuredMessaging-PEP to interoperate using Data Distribution Services for real-time system (DDS*) protocols and infrastructure. * https://www.omg.org/spec/DDS
Device Type: Interface	This interface specialization enables the StructuredMessaging-PEP to interoperate with (IoT) devices in the environment.
FileShare Type: Interface	This file-share interface specialization enables the StructuredMessaging-PEP to interoperate with shared file stores in the users' environment.

`

Table 16 - Structured Messaging Interfaces Elements	
Element Name	**Element Descriptions**
Middleware Type: Interface	This middleware interface specialization enables the StructuredMessaging-PEP to interoperate with user applications, data systems, and devices using prescribed middleware technology.
ObjectStore Type: Interface	This object store interface specialization enables the StructuredMessaging-PEP to interoperate with Object-Orients Database Management Systems in the users' environment.

PEP-DataInterface
 	
Type: Interface
	The data interface enables the StructuredMessaging-PEP to interoperate with systems and applications in the users' environment, e.g.:
	• Systems,
	• User Application;
	• Middleware, including:
	  ○ DDS;
	  ○ Web Services or
	  ○ Enterprise Service Bus and
	• Data Stores, including:
	  ○ RDBMS;
	  ○ Object Store;and
	  ○ File Share.

RDBMS Type: Interface	This RDBMS interface specialization enables the StructuredMessaging-PEP to interoperate with Relational Database Management Systems (RDBMS) in the users' environment.
REST Type: Interface	This REST interface specialization enables the StructuredMessaging-PEP to interoperate with Representational State Transfer (REST) protocols and infrastructure.
SOAP Type: Interface	This SOAP interface specialization enables the StructuredMessaging-PEP to interoperate with Simple Object Access Protocol (SOAP) protocols and infrastructure.
System Type: Interface	This system interface specialization enables the StructuredMessaging-PEP to interoperate with information and data systems in the environment.

Table 16 - Structured Messaging Interfaces Elements	
Element Name	**Element Descriptions**
UserApplication Type: Interface	This user application interface enables the PEP to interoperate with the user-specified applications. The implementor or integrator tailors this interface to the API of the user-specified application. Clause 10.2 provides details on this interface.
WebServices Type: Interface	This web services interface specialization enables the StructuredMessaging-PEP to interoperate with web service protocols and infrastructure.

7.6.3 Other External Interfaces and Integration Points

The following clauses outline other external interfaces to user-specified infrastructure and applications. These interfaces enable the user to employ external services to deliver the functions of the IEF-defined components.

7.6.3.1 SSG External Interfaces

The following figure illustrates the SSG interfaces to the users' security infrastructure.

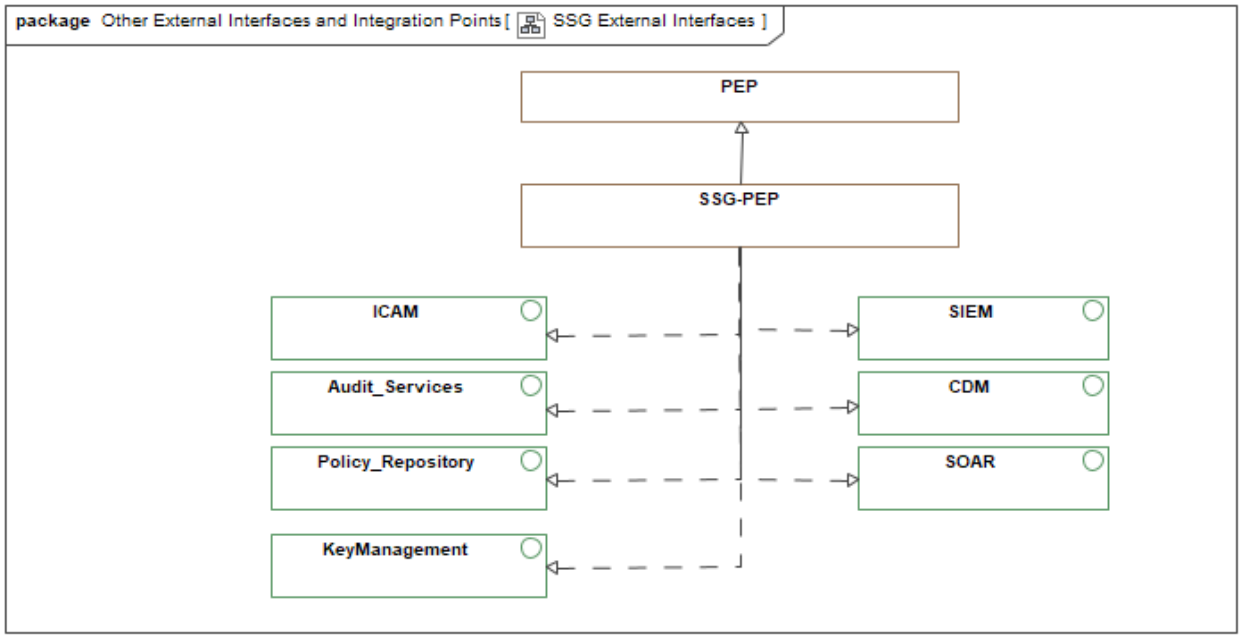


Figure 47 -SSG External Interfaces

The following table describes the elements illustrated in the previous figure - SSG External Interfaces.

`

Table 17 - SSG External Interfaces Elements	
Element Name	**Element Descriptions**
Audit_Services Type: Interface	This interface enables the IEF to integrate with and respond to the specified applications and services that require access information maintained as part of the Trusted Logging Services and the configurations and policy environments of the instantiated IEF components.
CDM Type: Interface	This optional interface enables the PEP to interoperate with the user-specified Continuous Diagnostic Monitoring (CDM) systems. It aligns the IEF PEP with ZTA requirements. The requirements and functions provided by this interface are left to the user (/implementer/integrator) to define based on the capabilities and API of the specified CDM system.
ICAM Type: Interface	This optional (see notes below) interface enables the PEP to interoperate with the user-specified Identity Credential and Access Management (ICAM) system. It also provides functions to authenticate users of IEF services and request and receive attributes (metadata) about a user (system/application/device) seeking to access IEF components, services, and data. The PEP may request user attributes, including: • Security Level; • Role; • Operational Associations; • Data attributes; • Equipment being used; and • Location. The requirements and functions provided by this interface are left to the user (/implementer/integrator) to define based on the capabilities and API of the specified ICAM system.
KeyManagement Type: Interface	The SSG (or PEP) may provide interfaces that enable IEF components to interoperate with user-specified policy Key Management Services (KMS), which manage (e.g., generate, escrow, and provision) the cryptographic keys.
Policy_Repository Type: Interface	The SSG may provide interface(s) that enable IEF components to interoperate with the user's policy development environment or policy management registry/repository to retrieve or publish IEF or DSS policies or IEF component configurations.
SIEM Type: Interface	This optional interface enables the PEP to interoperate with the user-specified Security Information and Event Management (SIEM) system. IT aligns the IEF PEP with ZTA requirements. The requirements and functions provided by this interface are left to the user (/implementer/integrator) to define based on the capabilities and API of the specified SIEM system.

Table 17 - SSG External Interfaces Elements	
Element Name	**Element Descriptions**
SOAR Type: Interface	This optional interface enables the PEP to interoperate with the user-specified Security Orchestration, Automation, and Response (SOAR) system. It aligns the IEF PEP with ZTA requirements. The requirements and functions provided by this interface are left to the user (/implementer/integrator) to define based on the capabilities and API of the specified SOAR system.

7.6.3.2 PDP External Interfaces

The following figure illustrates the PDP interfaces to the users' security infrastructure.

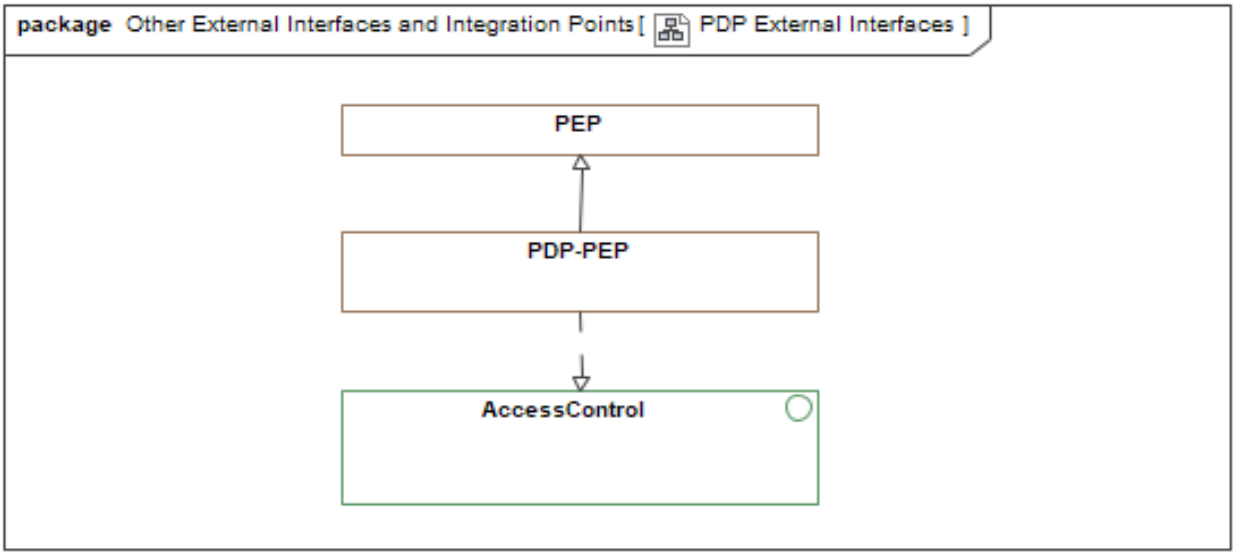


Figure 48 -PDP External Interfaces

The following table describes the elements illustrated in the previous figure - PDP External Interfaces.

Table 18 - PDP External Interfaces Elements	
Element Name	**Element Descriptions**
AccessControl Type: Interface	This PDP interface must provide features that enable the PEP to interoperate with user-specified Access Control and adjudication services, e.g.: • Attribute Based Access Control (ABAC); • Role Based Access Control (RBAC); • Policy Based-Access Control (PBAC) and • User-Based Access Control (UBAC).

`

Table 18 - PDP External Interfaces Elements	
Element Name	**Element Descriptions**
	The PEP prepares an adjudication request to the access control system (e.g., ABAC) and enforces its response. The requirements and functions provided by this interface are left to the user (/implementer/integrator) to define based on the capabilities and API of the specified Access Control (PDP) system.

7.6.3.3 CTS External Interfaces

The following figure illustrates the SSG interfaces to the users' security infrastructure.

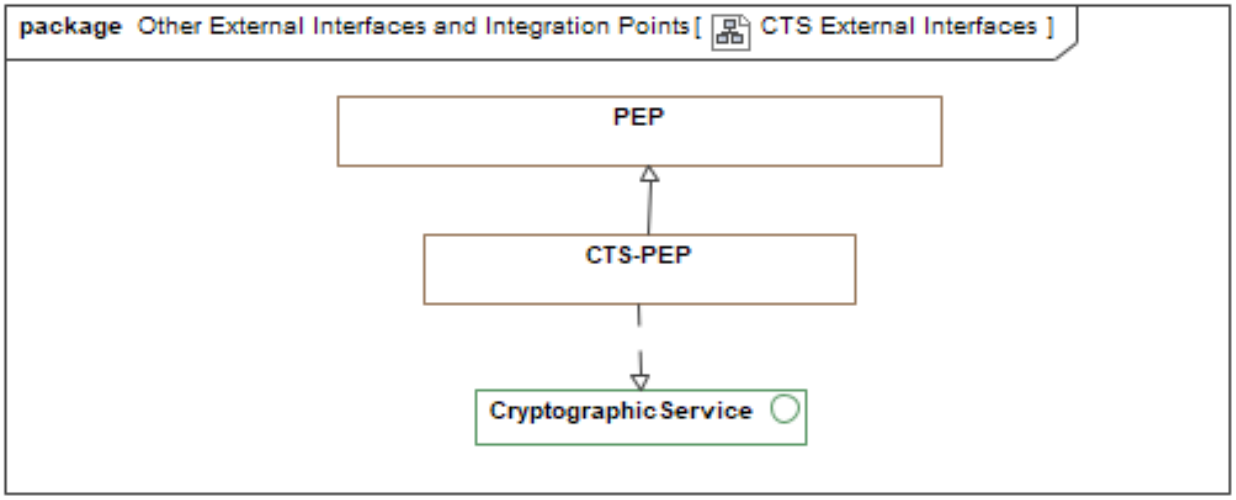


Figure 49 -CTS External Interfaces

The following table describes the elements illustrated in the previous figure - CTS External Interfaces.

Table 19 - CTS External Interfaces Elements	
Element Name	**Element Descriptions**
CryptographicService Type: Interface	The cryptographic service interface provides features that enable the PEP or CTS to interoperate with user-specified cryptographic services. The requirements and functions provided by this interface are left to the user (/implementer/integrator) to define based on the capabilities and API of the specified cryptographic system.

7.6.3.4 TLS External Interfaces

The following figure illustrates the TLS interfaces to the users' security infrastructure.

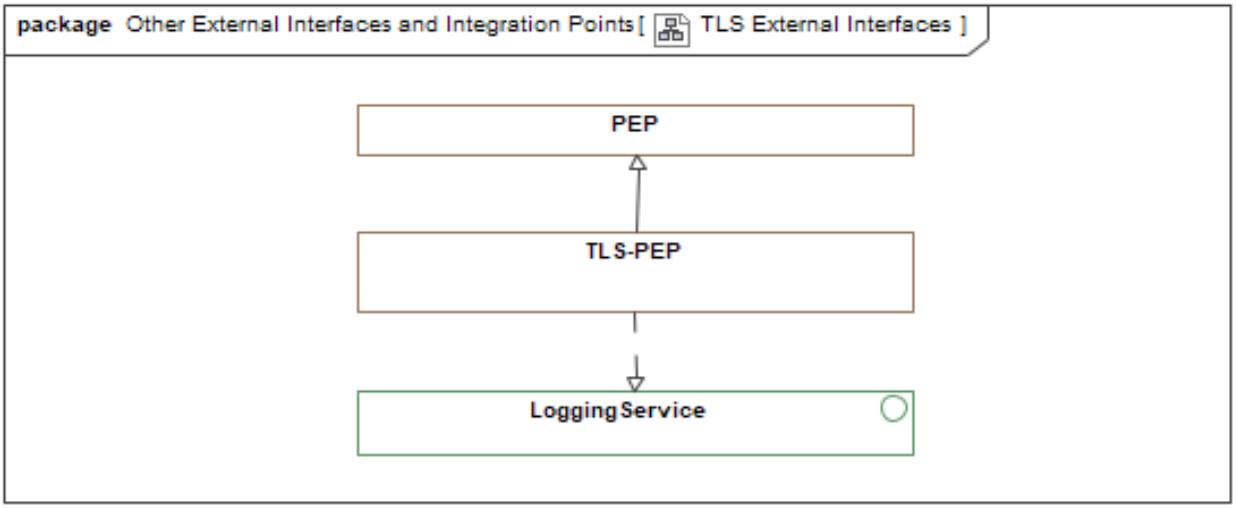


Figure 50 -TLS External Interfaces

The following table describes the elements illustrated in the previous figure - TLS External Interfaces.

Table 20 - TLS External Interfaces Elements	
Element Name	**Element Descriptions**
LoggingService Type: Interface	This TLS interface (see notes below) must provide features that enable the PEP or TLS to interoperate with the user-specified log management service (/system). The requirements and functions provided by this interface are left to the user (/implementer/integrator) to define based on the capabilities and API of the specified logging system.

7.6.3.5 PAP External Interface

The following figure illustrates the IEF interfaces to the users' administrative applications or systems.

`

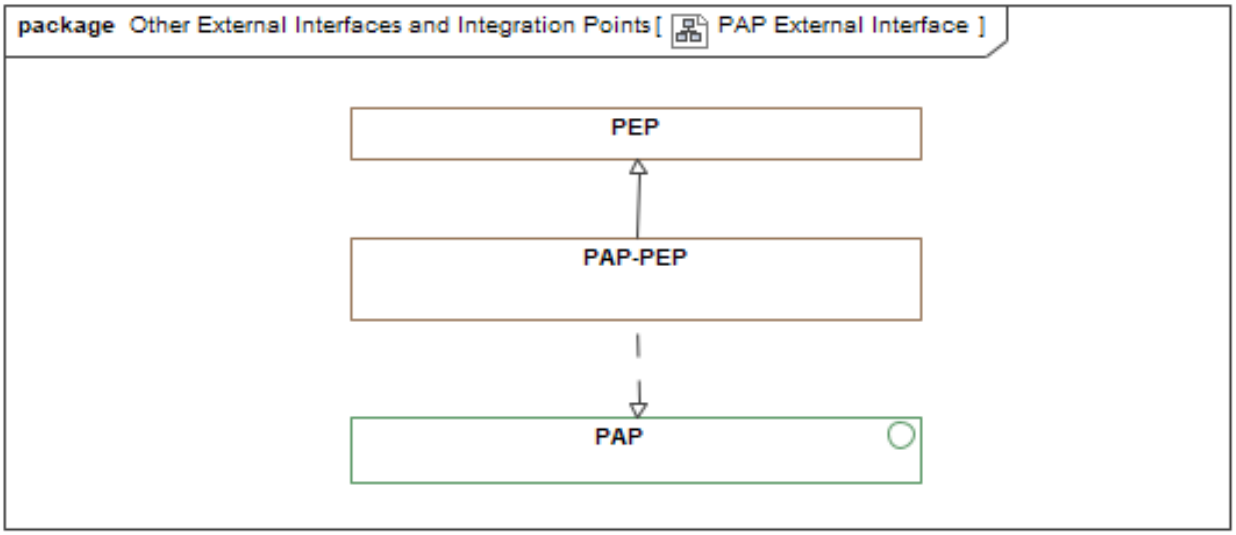


Figure 51 -PAP External Interface

The following table describes the elements illustrated in the previous figure - PAP External Interface.

Table 21 - PAP External Interface Elements	
Element Name	**Element Descriptions**
PAP Type: Interface	The PAP-Interface provides features that enable an external PAP application to send messages with an IEF configuration using user-specified middleware. The PAP and IEF components use the same message formats as used over the SMB, including: • PAP CommandMessage • PAP-CommandResponse; and • PAP-AlertWarning.

8 Policy Administration Point (PAP)

The Policy Administration Point (PAP) provides the user interface and services required to manage and administer IEF components and policy environments. Within the user's security policy limits, the PAP enables the authoring, deploying, and maintaining ISS policies and component configurations. The PAP enables an authorized user to manage and administer one or more IEF Components within its designated operating environment. The functions of the PAP must enable an authorized user to:

- Start-up in a default configuration, as specified in user-defined configuration files;
- Enable an authorized user to update its configuration based on changes in the operating context;
- Reestablish operation using the last known valid policy, configuration, and data states on resumption after a failure or error condition;
- Provide each IEF component with its distinct set of policies and configuration files (/operational parameters) and
- Provide a user interface that enables an authorized user to access PAP functions and services, including:
    - Disseminate ISS policies to specified IEF components;
    - Receive policies from user-specified policy management and development environment(s);
    - Validate and verify that received ISS policies are provisioned from an authorized source and conform to ISS specifications and configuration files;
    - Persist policies to the policy store specified for the PAP;
        - Retrieve policies from the policy store specified for the PAP;
    - Direct a specified IEF component to:
        - Provide its current operating configuration;
        - Provide its current policy configuration;
        - Receive ISS Policies;
        - Activate policies;
        - Deactivate policies;
        - Configure and tailor policies and
        - Save/Archive policies.
    - Direct a specified IEF component to change the value of one or more configuration properties;
    - Direct the specified IEF component to use its current operating configuration as default;
    - (Optional) Enable an authorized user to manage policy and configuration schedules, including:
        - Prepare a policy dissemination, activation, and deactivation schedule for each IEF component;
        - Load a policy schedule;
        - Activate policy schedules;
        - Deactivate policy schedules;
        - Modify policy schedules;
        - Prepare configuration dissemination, activation, and deactivation schedules for each IEF component;
        - Activate configuration schedule;

`

        - Deactivate configuration schedules and
        - Tailor configuration schedules;
    - Optional: Provide automated scheduling updates for each IEF component`s policy and configuration and
    - Optional: Provide automated scheduling updates for each IEF component`s policy and configuration to adapt IEF operation based on operational context, including:
        - Operational Phase;
        - Threat and Risk Level;
        - Operational Intent;
        - Roles & Responsibility; and
        - Operating Configuration.

8.1 PAP Operations

The PAP must provide features that intercept user (administrator) command requests, extract the command type and pertinent data elements, and interact with the PDP to adjudicate each user request. (See PDP-AuthorizationRequest message.)

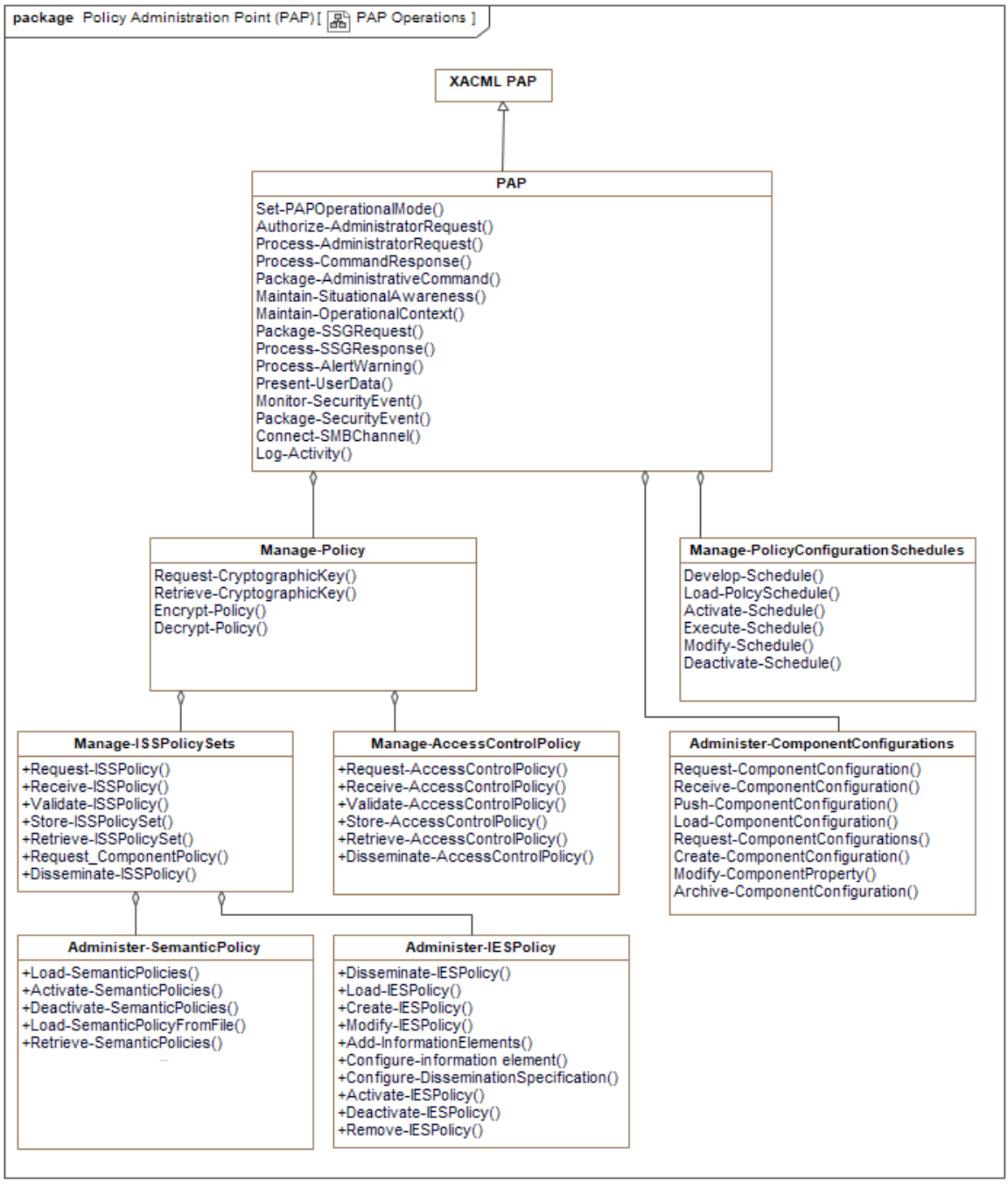


Figure 52 -PAP Operations

The following table identifies and describes the elements and operations illustrated in the previous figure - PAP Operations.

`

Table 22 - PAP Operations Elements	
Element Name	**Element and Operation Descriptions**
Administer-ComponentConfigurations	The PAP provides features that enable an authorized user (/administrator) to configure the operation of IEF components (e.g., PAP, PDP, PEPs, SSG, and SMB) in the operational environment. The user interface and command generation features for each component type form part of the PAP 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: SMB, PDPs, PEPs, PPSs, and SSG; • 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 primarily identifies interfaces and required functions to operate a policy-driven, data-centric ISS solution; the user and integrators determine whether these are manual or automated functions and interfaces. In each case, the PAP packages a PAP-Command containing the user's directive and issues the message to the IEF component over the SMB.
	Element Type: Class Owned Operations: Request-ComponentConfiguration: The PAP must provide features that enable an authorized user to request a component configuration for one or more IEF Components as a PAP Command Message over the SMB. Receive-ComponentConfiguration: The PAP must provide features that enable an authorized user to package and issue a PAP-Command message instructing 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. Push-ComponentConfiguration:

Table 22 - PAP Operations Elements	
Element Name	**Element and Operation Descriptions**
	The PAP must provide features that enable an authorized user to deploy a component configuration to an IEF component over the SMB. Load-ComponentConfiguration: The PAP must provide features that enable an authorized user to direct an IEF Component to load and execute a specified configuration using a PAP-Command message over the SMB. Request-ComponentConfigurations: The PAP must provide features that enable an authorized user to package and issue a PAP-Command message instructing an IEF component to package its operating configuration and store it in a specified location in the IEF-protected information store. Create-ComponentConfiguration: The PAP must provide features that enable an authorized user to create a component configuration for an IEF component. Modify-ComponentProperty: The PAP must provide features that enable an authorized user to edit component property and issue these changes to one or more components. Archive-ComponentConfiguration: The PAP must provide features that enable an authorized user to instruct an IEF component to assemble and package its current configuration and return the data set to the PAP (see PAP-CommandResponse message). The PAP then persists the component configuration as a file in a SecureAssetContainer to a protected information store.
Administer-IESPolicy	The PAP provides features enabling authorized users to administer Information Exchange Specification (IES) policies during operations.
	Element Type: Class Owned Operations: Disseminate-IESPolicy: The PAP must provide features that enable an authorized user to disseminate a set [1..*] of semantic policies to a specified PPS. Load-IESPolicy: The PAP must provide features that enable an authorized user to direct a PPS to load a specified set of semantic policies into operations. Create-IESPolicy: The PAP must provide features that enable an authorized user to create an Information Exchange Specification (IES) during operations.

`

Table 22 - PAP Operations Elements	
Element Name	**Element and Operation Descriptions**
	Modify-IESPolicy: The PAP must provide features that enable authorized users to configure or modify IES during operations. Add-InformationElements: The PAP must provide features that enable an authorized user to add information elements (Filtered Semantic Elements) to an IES. Configure-information element: The PAP must provide features that enable an authorized user to configure an information element (Filtered Semantic Element) using the patterns specified in the IES and Semantic policies loaded within the PPS. Configure-DisseminationSpecification: The PAP must provide features that enable an authorized user to configure the IES dissemination parameters using the patterns specified in the IES policies loaded within the PPS. Activate-IESPolicy: The PAP must provide features that enable an authorized user to activate IES policies. Deactivate-IESPolicy: The PAP must provide features that enable an authorized user to deactivate IES policies. Remove-IESPolicy: The PAP must provide features that enable an authorized user to remove (or delete) IES policies.
Administer-SemanticPolicy	The PAP provides features enabling authorized users to administer Information Exchange Specification (IES) policies during operations.
	Element Type: Class Owned Operations: Disseminate-SemanticPolicies: The PAP must enable an authorized user to disseminate semantic policies to a specified PPS. Load-SemanticPolicies: The PAP must provide features that enable an authorized user to direct a PPS to load a specified set of semantic policies disseminated by the PAP. Activate-SemanticPolicies:

Table 22 - PAP Operations Elements	
Element Name	**Element and Operation Descriptions**
	The PAP must provide features that enable an authorized user to direct a PPS to activate a specified set of semantic policies. Deactivate-SemanticPolicies: The PAP must provide features that enable an authorized user to direct a PPS to activate a specified set of semantic policies. Load-SemanticPolicyFromFile: The PAP must provide features that enable an authorized user to direct a PPS to load a specified set of semantic policies from a file. Retrieve-SemanticPolicies: The PAP must provide features that enable an authorized user to retrieve semantic policies from a file.
Manage-AccessControlPolicy	(Optional) The PAP's primary function is managing and administrating Access Control policies during operations. These features are only required if the PDP hosts access control adjudication and policy libraries. Many implementations will integrate the IEF with an enterprise access control system (e.g., ABAC).
	Element Type: Class Owned Operations: Request-AccessControlPolicy: The PAP may provide features that enable an authorized user to request an Access Control Policy set from a specified policy management or development environment through the SMB/SSG. Receive-AccessControlPolicy: The PAP may provide features that receive a requested access control policy from an authorized source using the SMB/SSG. Validate-AccessControlPolicy: The PAP may provide features that validate and verify the format, content, and source of an access control policy set for the PDP. These features ensure that the provisioned policy set is from an authorized source and conforms to policy language specifications (i.e., XACML). Access Control policies are typically XML documents that contain the rules and constraints enforced by the PDP to ensure that recipients only receive the data they need and are authorized to receive. Store-AccessControlPolicy: The PAP may provide features that enable an authorized user to persist a set of policies to a specified location. Retrieve-AccessControlPolicy: The PAP may provide features that enable an authorized user to retrieve an access control policy set from a specified location.

`

Table 22 - PAP Operations Elements	
Element Name	**Element and Operation Descriptions**
	Disseminate-AccessControlPolicy: The PAP may provide features that enable an authorized user to disseminate policy to the PDP.
Manage-ISSPolicySets	A primary function of the PAP is managing and administrating Information Sharing and Safeguarding (ISS) policies during operations.
	Element Type: Class Owned Operations: Request-ISSPolicy: The PAP must provide features that enable an authorized user to request an ISS Policy set from a user-specified Policy Management or Development Environment (e.g., Battle Lab) accessible through the SMB/SSG. Receive-ISSPolicy: The PAP must provide features that receive an ISS Policy Set from an authorized source through the SMB/SSG. Validate-ISSPolicy: The PAP must provide features that validate and verify an ISS policy Set's format, content, and source. These features ensure that the provisioned policy is from an authorized source and conforms to ISS policy language specifications (i.e., IEPPV). The PAP must also verify that the ISS Policies apply to the data domain it protects. Store-ISSPolicySet: The PAP must provide features that enable an authorized user to securely persist a set of ISS policies to a specified location. Retrieve-ISSPolicySet: The PAP must provide features that enable an authorized user to retrieve an ISS Policy set from a specified location. Request_ComponentPolicy: The PAP must provide features that enable authorized users to direct a PPS to package its ISS policies and return them to the PAP. Disseminate-ISSPolicy: The PAP must provide features that enable an authorized user to direct the PAP to disseminate ISS policy to the PPS.

Table 22 - PAP Operations Elements	
Element Name	**Element and Operation Descriptions**
Manage-Policy	The PAP must enable an authorized user to manage ISS and Access Control policy sets for PPS and PDP, respectively. This administration will enable authorized users to tailor policies to accommodate changes in the 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-CryptographicKey: The PAP must provide features that enable internal services to request a cryptographic key from an authorized Key Management System using the SSG. Alternately, the PAP may integrate SSG service interfaces and interoperate directly with user-specified key management services to request cryptographic keys. Retrieve-CryptographicKey: The PAP must provide features that enable internal services to receive cryptographic keys from an authorized Key Management System using the SSG. Alternately, the PAP may integrate SSG service interfaces and interoperate directly with user-specified key management services to retrieve cryptographic keys. Encrypt-Policy: The PAP must provide features that invoke the CTS to transform the policy or policy set into an unintelligible form using the cryptographic key provided by the Key Management System through the SMB/SSG. Alternately, the PAP may integrate SSG service interfaces and interoperate directly with user-specified cryptographic services to encrypt data objects. Decrypt-Policy: The PAP must provide features that invoke the CTS to transform the policy or policy set into its normative and intelligible form using the cryptographic key provided by the Key Management System through the SMB/SSG. Alternately, the PAP may integrate SSG service interfaces and interoperate directly with user-specified cryptographic services to decrypt data objects.
Manage-PolicyConfigurationSchedules	(Optional) PAP may enable an authorized user to plan, schedule, and automate policy dissemination and activation according to mission parameters.

`

Table 22 - PAP Operations Elements	
Element Name	**Element and Operation Descriptions**
	Element Type: Class Owned Operations: Develop-Schedule: The PAP may provide features that enable an authorized user to plan policy dissemination and activation schedules coupled with the IEF component configurations required to execute the scheduled policy deployment. Load-PolcySchedule: The PAP may provide features that enable an authorized user to Load a policy schedule from a specified file. Activate-Schedule: The PAP may provide features that enable an authorized user to activate a loaded schedule. Execute-Schedule: The PAP may provide features that automate policy dissemination and activation per each mission's requirements. Modify-Schedule: The PAP may provide features that enable an authorized user to modify a loaded schedule. Deactivate-Schedule: The PAP may provide features that enable an authorized user to deactivate a loaded schedule.
PAP	The Policy Administration Point (PAP) provides the functionality to manage and administer IEF components during operation. The IEF defines a general architecture for a PAP, identifying its core sub-components, functions, and interfaces (e.g., protocols and content). The PAP may be implemented as a component within a secure operating environment or as an external capability operating through a PEP. The PAP provides authorized users with an interface and the tools to manage and administer IEF components and policies. The PAP uses standards (Clause 16) messaging to communicate with IEF components through the SMB within its environment. Messages include: 1. PAP Command Message; 2. PAP Command Response Message; and 3. PAP AlertWarning Message.

Table 22 - PAP Operations Elements	
Element Name	**Element and Operation Descriptions**
	PAP functionality may be delivered as a core IEF Function or as part of the users' system administration capability. The latter must interface with a data messaging PEP connected to the SMB.
	Element Type: Class Owned Operations: Set-PAPOperationalMode: The PAP must provide features that enable an authorized user to select the mode of operation for the PAP. PAP Modes of operation may include: 1. Basic: Provides the user with the most basic functionality of the PAP and access to limited control of each IEF component and policy. 2. Advanced: Provides the user with the full functionality of the PAP and access to fine-grained control of each IEF component and policy. 3. Test: Provides the user with an extended set of features that enable debugging and testing of the PAP. Authorize-AdministratorRequest: The PAP must provide features that intercept user (administrator) command requests, extract the command type and pertinent data elements, and interact with the PDP to adjudicate each user request (See PDP-AuthorizationRequest message). Process-AdministratorRequest: The PAP must provide features that stage the processing of an authorized command request, including: 1. Manage PAP configuration and operation; 2. Manage and administer PDP and PPS policy; 3. Administer IEF component configurations; 4. Execute policy and configuration schedules and 5. Maintain operational awareness. Process-CommandResponse: The PAP must provide features that parse the PAP-CommandResponse message from an IEF component, extract the pertinent data, integrate the data elements into the PAP's situational awareness, and present the user interface response. Package-AdministrativeCommand: The PAP must provide features that gather the data elements needed for a command request to an IEF component and format the data into the appropriate message form. (See PAP-Command Response Message) Maintain-SituationalAwareness:

`

Table 22 - PAP Operations Elements	
Element Name	**Element and Operation Descriptions**
	The PAP must provide features that gather, store, and maintain data describing the current operating state of the IEF environment and 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: The PAP may provide features that collect and maintain information about the operational context for the IEF. The operational context may identify: 1. The operational phase; 2. The operational partners; 3. The partners' roles and responsibilities; 4. Context-specific information-sharing policies and 5. The user specified the quality of service. The PAP can influence the configuration of the IEF components and the PDP's policies by using operational context. It gathers relevant data through PAP communications with the user's Situational Awareness (SA) or incident management services, and the SSG provides communications between the PAP and user security services. Package-SSGRequest: The PAP must provide features that gather the data elements needed for an SSG request and format the data into the appropriate message form. (See PAP-Command Response Message) Process-SSGResponse: The PAP must provide features that parse the SSG-Response message containing situational awareness and context information, extract the pertinent data, and integrate the data elements into the PAP's situational awareness data store. (see SSG-Response message) Process-AlertWarning: The PAP must provide features that receive a PAP-AlertWarning message, extract its type and relevant data elements, integrate it into the PAP situational awareness information, and issue the data to the user. (see PAP-AlertWarning message) Present-UserData:

Table 22 - PAP Operations Elements	
Element Name	**Element and Operation Descriptions**
	The PAP is responsible for presenting user data. It gathers information from the execution of PAP features and the situational awareness data store, packages the data, and sends it to the user for presentation. Monitor-SecurityEvent: The PAP must provide features that monitor and report (e.g., present to the user) security events provided as PAP-AlertWarning messages from the IEF Components. Package-SecurityEvent: The PAP must provide 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 IM, Email, and Messaging PEPs. Connect-SMBChannel: The PAP must provide features that enable an authorized user to package and issue a PAP-Command message instructing an IEF component to use an SMB communication channel. Log-Activity: The PAP must provide features that log transactions to user-specified logging services through the TLS.
XACML PAP	The IEF-PEP derives from the core concepts in the XACML specification and Reference Architecture. It extends the XACML specification's focus on applications, networks, and devices into the data domain and Data-Centric Security (DCS).
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.

8.2 Administer IEF Component Configurations

The following figure further refines the administration and configuration management features provided by the PAP.

`

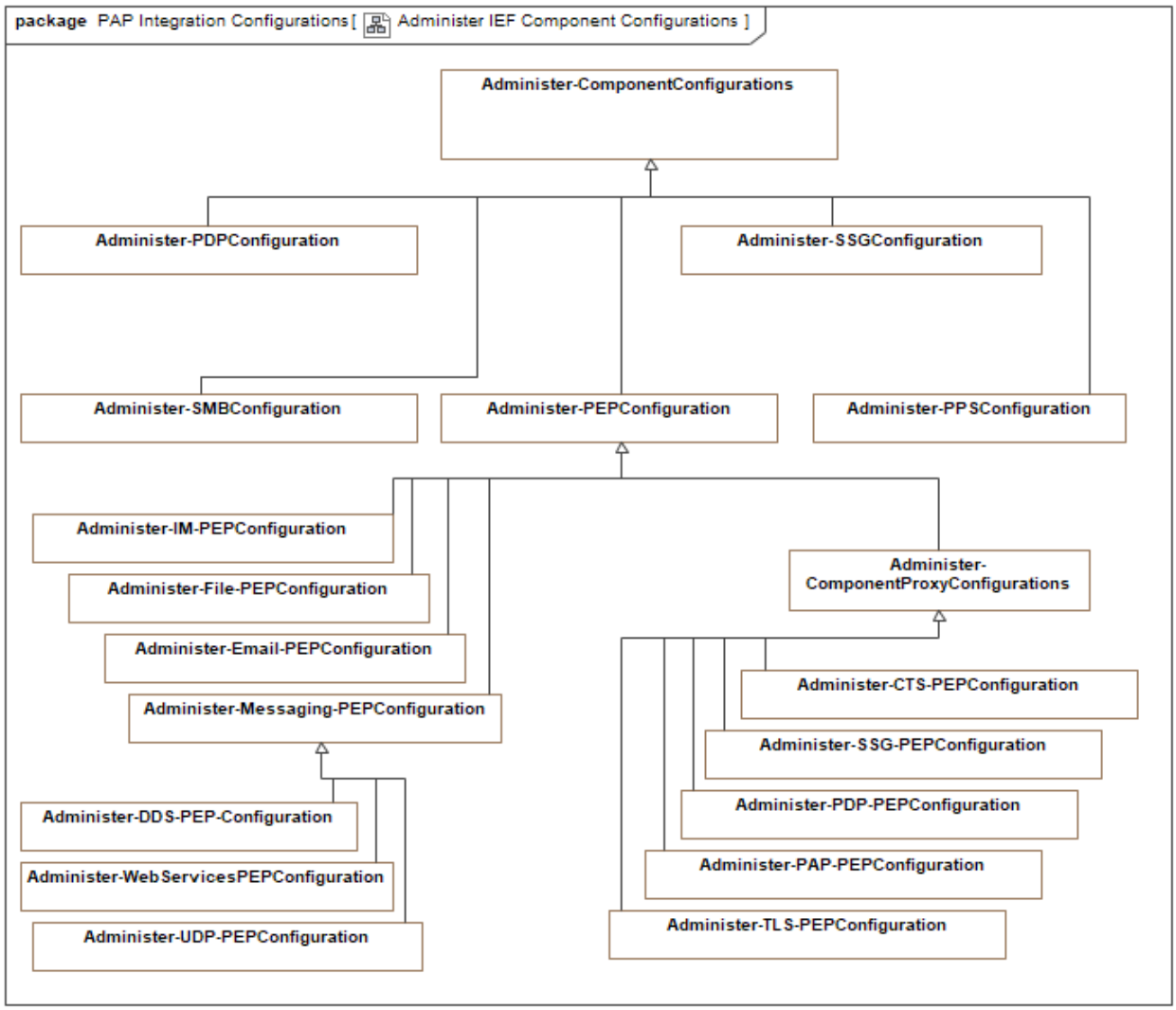


Figure 53 -Administer IEF Component Configurations

The following table identifies and describes the elements and operations illustrated in the previous figure - Administer IEF Component Configurations.

Table 23 - Administer IEF Component Configurations Elements	
Element Name	**Element and Operation Descriptions**
Administer-ComponentConfigurations	The PAP provides features that enable an authorized user (/administrator) to configure the operation of IEF components (e.g., PAP, PDP, PEPs, SSG, and SMB) in the operational environment. The user interface and command generation features for each component type form part of the PAP issued to the individual components where they are executed.

Table 23 - Administer IEF Component Configurations Elements	
Element Name	**Element and Operation Descriptions**
	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: SMB, PDPs, PEPs, PPSs, and SSG; • 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 primarily identifies interfaces and required functions to operate a policy-driven, data-centric ISS solution; the user and integrators determine whether these are manual or automated functions and interfaces. In each case, the PAP packages a PAP-Command containing the user's directive and issues the message to the IEF component over the SMB.
	Element Type: Class Owned Operations: Request-ComponentConfiguration: The PAP must provide features that enable an authorized user to request a component configuration for one or more IEF Components as a PAP Command Message over the SMB. Receive-ComponentConfiguration: The PAP must provide features that enable an authorized user to package and issue a PAP-Command message instructing 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. Push-ComponentConfiguration: The PAP must provide features that enable an authorized user to deploy a component configuration to an IEF component over the SMB. Load-ComponentConfiguration: The PAP must provide features that enable an authorized user to direct an IEF Component to load and execute a specified configuration using a PAP-Command message over the SMB. Request-ComponentConfigurations:

`

Table 23 - Administer IEF Component Configurations Elements	
Element Name	**Element and Operation Descriptions**
	The PAP must provide features that enable an authorized user to package and issue a PAP-Command message instructing an IEF component to package its operating configuration and store it in a specified location in the IEF-protected information store. Create-ComponentConfiguration: The PAP must provide features that enable an authorized user to create a component configuration for an IEF component. Modify-ComponentProperty: The PAP must provide features that enable an authorized user to edit component property and issue these changes to one or more components. Archive-ComponentConfiguration: The PAP must provide features that enable an authorized user to instruct an IEF component to assemble and package its current configuration and return the data set to the PAP (see PAP-CommandResponse message). The PAP then persists the component configuration as a file in a SecureAssetContainer to a protected information store.
Administer-ComponentProxyConfigurations	The PAP must provide features that enable an authorized user with the ability to administer specific IEF components that provide proxies to the user or infrastructure-delivered security services, e.g.: • SSG providing proxies to: ○ Identity Credential and Access Management (ICAM); ○ Security Information and Event Management; • PDP providing proxies to external access control adjudication, e.g.: ○ Attribute Based Access Control (ABAC); ○ Role Based Access Control (RBAC); ○ User Based Access Control (UBAC) or ○ Policy Based Access Control (PBAC); • TLS provides proxies for logging services; • CTS provides proxies to cryptographic services and • PAP provides proxies for Administration services and interfaces. Proxies implemented as PEPs can enforce Zero-Trust controls the user requires.

Table 23 - Administer IEF Component Configurations Elements	
Element Name	**Element and Operation Descriptions**
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
Administer-CTS-PEPConfiguration	The PAP must provide features that enable an authorized user to administer the CTS Services operating as a proxy to user-specified cryptographic services. This element allows the user to extend the IEF capability and exploit specific features of the cryptographic service provided by the user, implementor, or integrator.
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
Administer-DDS-PEP-Configuration	The PAP must provide features that enable an authorized user to administer a DDS interface (/proxy/PEP). The addition of PAP features will depend on the DDS implementation provided by the user, implementor, or integrator.
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
Administer-Email-PEPConfiguration	The PAP must provide features that enable an authorized user to administer an Email interface (/proxy/PEP). The addition of PAP features will depend on the email management clients and servers provided by the user, implementor, or integrator.
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
Administer-File-PEPConfiguration	The PAP must provide features that enable an authorized user to administer a File management interface (/proxy/PEP). The addition of PAP features will depend on the file management clients and servers provided by the user, implementor, or integrator.
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
Administer-IM-PEPConfiguration	The PAP must provide features that enable an authorized user to administer an instant messaging interface (proxy/PEP). The addition

Table 23 - Administer IEF Component Configurations Elements	
Element Name	**Element and Operation Descriptions**
	of PAP features will depend on the Instant Messaging or Chat clients and server provided by the user, implementor, or integrator.
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
Administer-Messaging-PEPConfiguration	The PAP must provide features that enable an authorized user to administer a middleware interface (/proxy/PEP). The addition of PAP features will depend on the middleware provided by the user, implementor, or integrator.
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
Administer-PAP-PEPConfiguration	The PAP must provide features that enable an authorized user to administer the PAP Services operating as a proxy to user-specified administration services. This element provides the ability to extend the IEF capability and exploit specific features of the security services provided by the user, implementor, or integrator.
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
Administer-PDP-PEPConfiguration	The PAP must provide features that enable an authorized user to administer a Policy Decision Point (PDP). In most instances, the PDP is a proxy for Access control services (e.g., ABAC, RBAC, PBAC, or UBAC).
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
Administer-PDPConfiguration	The PAP must provide features that enable an authorized user to administer a Policy Decision Point (PDP). In most instances, the PDP is a proxy for the Access control services (e.g., ABAC, RBAC, PBAC, or UBAC).

Table 23 - Administer IEF Component Configurations Elements	
Element Name	**Element and Operation Descriptions**
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
Administer-PEPConfiguration	The PAP must provide features that enable an authorized user to administer Policy Enforcement Points (PEP).
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
Administer-PPSConfiguration	The PAP must provide features that enable an authorized user to administer a Policy-based Packaging and Processing Service (PPS).
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
Administer-SMBConfiguration	The PAP must provide features that enable an authorized user to administer an IEF Secure Messaging Bus (SMB).
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
Administer-SSG-PEPConfiguration	The PAP must provide features that enable an authorized user to administer the SSG Services operating as a proxy to user-specified security (e.g., ICAM and key management) services. This element provides the ability to extend the IEF capability and exploit specific features of the security services provided by the user, implementor, or integrator.
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
Administer-SSGConfiguration	The PAP must provide features that enable an authorized user to administer an IEF Security Services Gateway.

`

Table 23 - Administer IEF Component Configurations Elements	
Element Name	**Element and Operation Descriptions**
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
Administer-TLS-PEPConfiguration	The PAP must provide features that enable an authorized user to administer the TLS Services, which operate as a proxy to user-specified logging services. This element provides the ability to extend the IEF capability and exploit specific features of the security services provided by the user, implementor, or integrator.
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
Administer-UDP-PEPConfiguration	The PAP must provide features that enable an authorized user to administer a UDP interface (proxy/PEP). The addition of PAP features will depend on the UDP implementation provided by the user, implementor, or integrator.
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
Administer-WebServicesPEPConfiguration	The PAP must provide features that enable an authorized user to administer a Web Services interface (/proxy/PEP). The addition of PAP features will depend on the implementation of Web Services by the user, implementor, or integrator.
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.

8.3 Administer Component Policy Operations

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

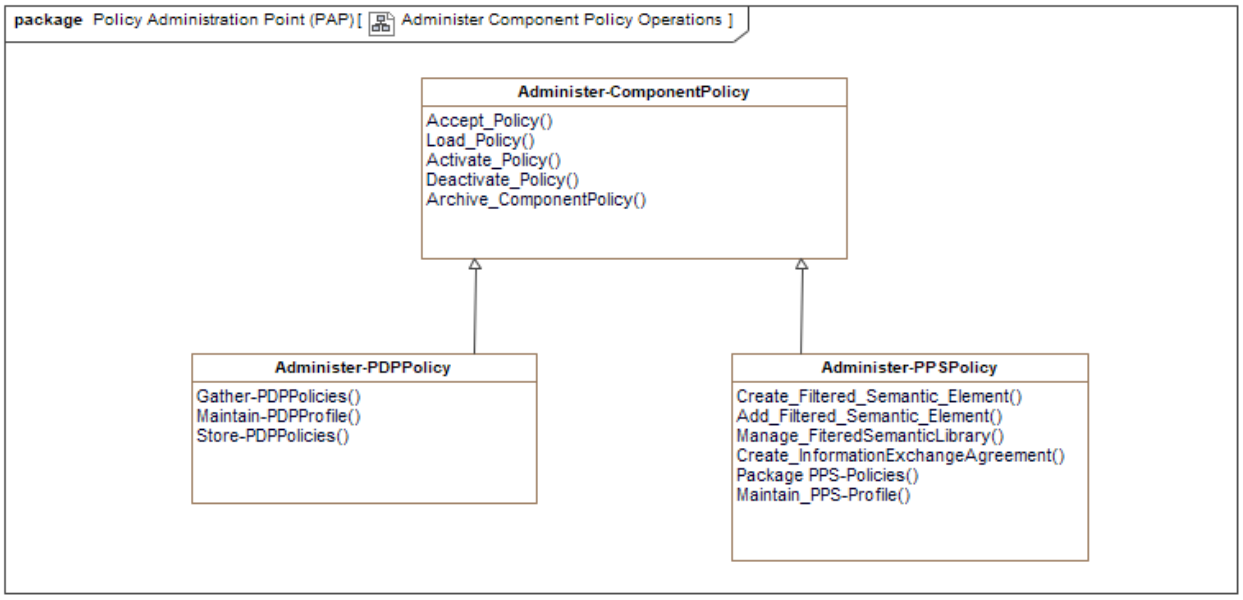


Figure 54 -Administer Component Policy Operations

The following table identifies and describes the elements and operations illustrated in the previous figure - Administer Component Policy Operations.

Table 24 - Administer Component Policy Operations Elements	
Element Name	**Element and Operation Descriptions**
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 must provide features that enable an authorized user to package and issue a PAP-Command message instructing the IEF component to receive a policy or policy set. The instructions are packaged and issued as a PAP-Command message. Load_Policy: The PAP must provide 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 must provide features that enable an authorized user to package and issue a PAP-Command message instructing a PDP or PPS to activate a policy in its environment.

`

Table 24 - Administer Component Policy Operations Elements	
Element Name	**Element and Operation Descriptions**
	Deactivate_Policy: The PAP must provide features that enable an authorized user to package and issue a PAP-Command message instructing a PDP or PPS to deactivate a policy in its policy environment. Archive_ComponentPolicy: The PAP must provide 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.
Administer-PDPPolicy	The PAP must enable an authorized user to administer the configurations and policies for one or more PDPs operating within the PAP-designated environment. The features also enable the user to manage the access and release of control policies employed on a remote PDP.
	Element Type: Class Owned Operations: Gather-PDPPolicies: The PAP must provide features that enable an authorized user to direct the PAP to gather PDP (access and release control) policies. Maintain-PDPProfile: The PAP must provide features that enable an authorized 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 (configuration) parameters, archived policy sets, and PDP status and events (alerts/warnings). Store-PDPPolicies: The PAP must provide features that enable an authorized user (/administrator) to write PDP policies to file in a location specified by the user.
Administer-PPSPolicy	The PAP must 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.

Table 24 - Administer Component Policy Operations Elements	
Element Name	**Element and Operation Descriptions**
	Element Type: Class Owned Operations: Create_Filtered_Semantic_Element: The PAP must provide 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 must enable an authorized user to add a configured FilteredSemanticElement to a specified InformationExchange_Specification (/contract). Manage_FiteredSemanticLibrary: The PAP must provide features that enable an authorized user to acquire, store, sort, and access pre-configured collections of FilteredSemanticElements. Create_InformationExchangeAgreement: The PAP must provide features that enable an authorized user to create a new InformationExchangeSpecification. As part of this process, the user specifies: 1. Session Information; 2. Channel Information; 3. Message and Network Protocols to be applied; 4. Security, Privacy, and sensitivity restrictions and 5. Information Elements to be enabled. Package PPS-Policies: The PAP must provide features that enable an authorized user to collect specified PPS (data packaging and processing) policies and package them for release or storage. Maintain_PPS-Profile: The PAP must provide features that enable an authorized user (/administrator) to administer profiles for each PPS in the environment. A PPS profile may include the 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).

`

8.4 PAP Integration Configurations

The following clauses illustrate several configuration patterns for integrating PAP functionality into an IEF Configuration. The PAP is a message-based interface to enable users, implementors, and integrators to deploy the PAP in several configurations, including:

1. An integrated component connected to the SMB;
2. As an external component connected via a PEP and
3. As administrative services within the user's infrastructure.

8.4.1 Administer Component Policy Operations

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

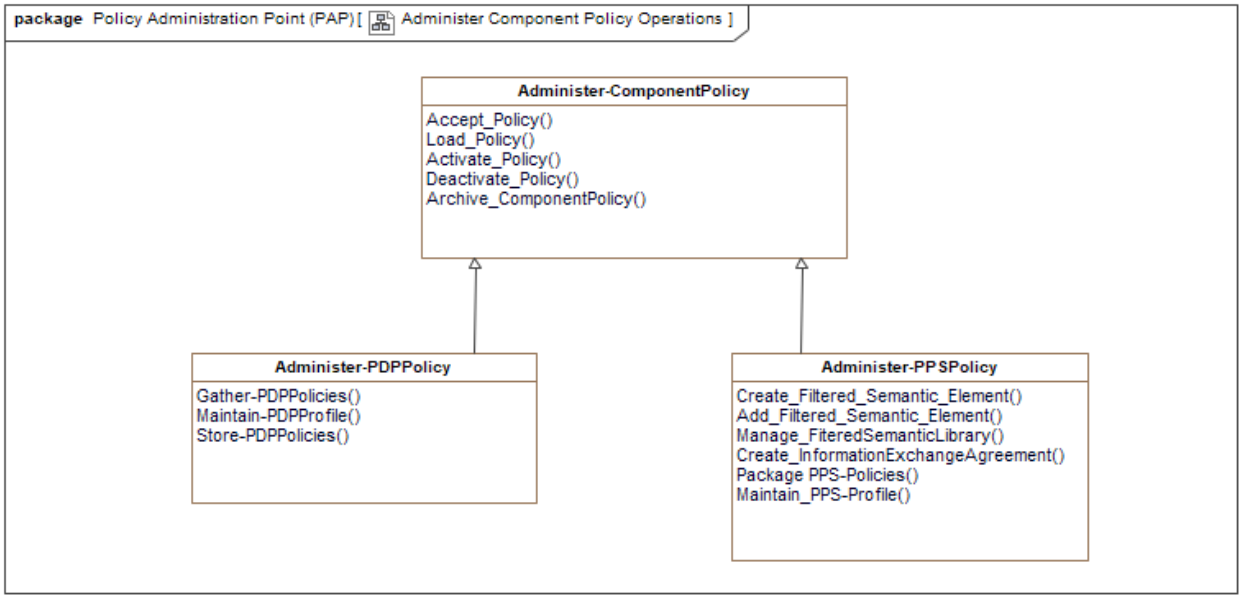


Figure 55 -Administer Component Policy Operations

8.4.2 Administer Component Policy Operations

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

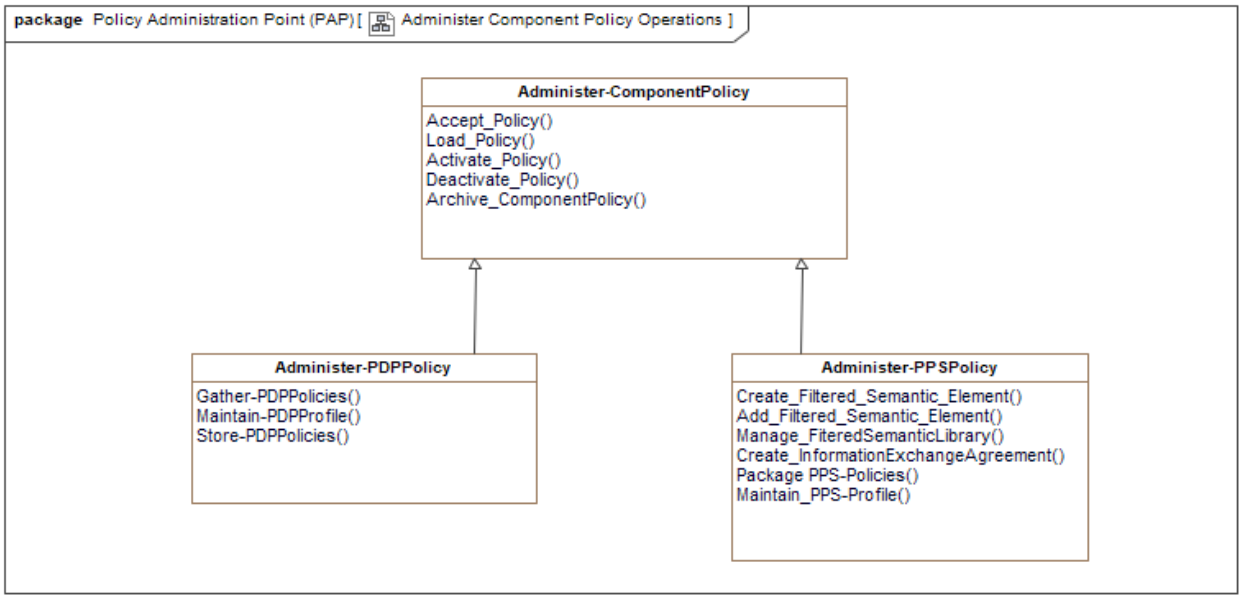


Figure 56 -Administer Component Policy Operations

`

9 Policy Decision Point (PDP)

The Policy Decision Point (PDP) adjudicates all data requests, receipts, and releases. The PDP adjudication and decisions incorporate the following data points:

- The attributes of the data user (e.g., attributes, location, and role);
- The attributes (/labels/metadata) bound to the data resources requested, received, or released;
- Access control policies tailored to the mission requirements to safeguard the specified data resource(s);
- (Optional) the security of the applications, systems, devices, and networks defined in authenticated software, hardware, and Network Bills of Material (BOM); and
- (Optional) the operational context.

The PEPs, PAP, and other integration points use the PDP to adjudicate access and release controls. The PEP features integrated into each outward-looking integration point formulate PDP queries and submit them to the PDP. Depending on the PDP response, the PEP features will authorize or deny the transaction. 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 authorized to request the specified message content using the communication channel?
- Is the data producer authorized to release the specified content?

The generalized decision process follows the following steps:

- The IEF service (PEP, PAP Interface, SSG, CTS, and TLS) intercepts each user information-sharing action;
- The IEF service extracts the metadata from the information elements in the exchange;
- The IEF service reformulates a decision request and sends it to the PDP via the SMB;
- The PDP receives the request and retrieves applicable security policy rules from the policy store;
- Interpreting the user's action in the context of the security policy, the PDP returns a policy decision (e.g., Permit or Deny);
- The PDP sends its decision to the requesting service;
- The requesting service enforces the PDP decision and
- The requesting service function logs the transaction to the TLS.

9.1 PDP 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 expresses the access control rules for the specific security domain and operation. The IEF PDP relies on the XACML as a policy expression for the PDP specification.

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

- The sensitivity (privacy, confidentiality, classification, or legal significance) of data elements being requested or accessed;
- The users' attributes to receive, access, release, or process the requested or accessed data elements;

- (Optional) The location (physical or network) of the recipient; and
- (Optional) The operational context in which the decision is made (e.g., phase, threat, operational roles (recipient and sender)).

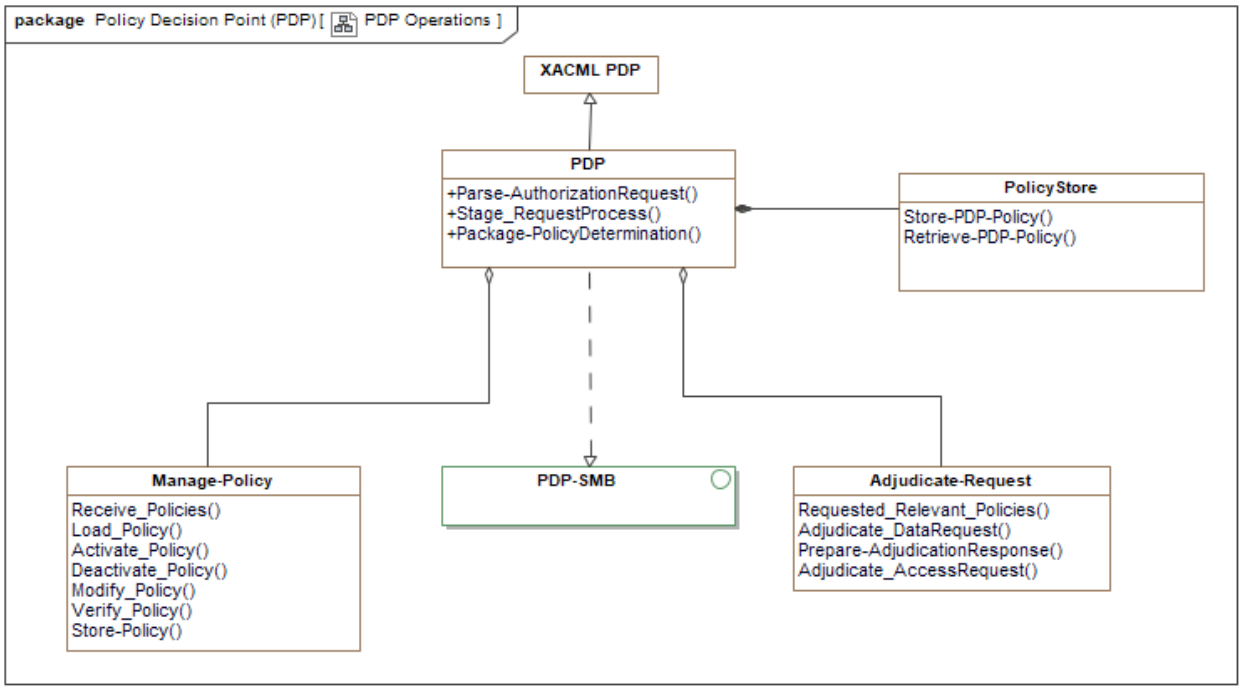


Figure 57 -PDP Operations

The following table identifies and describes the elements and operations illustrated in the previous figure - PDP Operations.

Table 25 - PDP Operations Elements	
Element Name	**Element and Operation Descriptions**
Adjudicate-Request	PDP features that combine to adjudicate a request by an IEF component to authorize: • Release InformationElements to a specified user; • Access to a specified information element by a specified user and • Execution of specified functions/operations by a specified user or IEF component.
	Element Type: Class Owned Operations: Requested_Relevant_Policies:

`

Table 25 - PDP Operations Elements	
Element Name	**Element and Operation Descriptions**
	The PDP must provide features that retrieve policies appropriate for adjudicating the specific request. The PDP retrieves relevant policies from its internal policy store. Adjudicate_DataRequest: The PDP must provide features that validate and verify that a specific user (e.g., provider and recipient) is authorized (has the attributes or policy rights) to release the specific information content to the specified recipients using the specified communication channels. PDP adjudication employs policies incorporating the users' explicit attributes, the markings (e.g., classifications and restrictions, releasability, and restrictions) bound to each information element, and PDP-relevant policies. If the user's authorization is verified, the PDP issues instructions to the PEP to permit the request. The PDP (e.g., data owner) specifies individual policies for files, email and attachments, instant messaging and attachments, structured messages and attachments. Prepare-AdjudicationResponse: The PDP must provide features that format the adjudication response for release. Adjudicate_AccessRequest: The PDP must provide features to validate and verify that each user is authorized to perform the specified request in conformance with the user-specified policy (e.g., ABAC Policy).
Manage-Policy	PDP features that execute policy management commands from an authorized PAP. PAP policy management commands include: 1. Accept_Policy; 2. Activate_Policy; 3. Deactivate_Policy; 4. Load_Policy; and 5. Modify_Policy.
	Element Type: Class Owned Operations: Receive_Policies: The PDP must provide features that respond to a PAP-Command message, directing it to receive a policy or set of policies contained in the message. Load_Policy:

Table 25 - PDP Operations Elements	
Element Name	**Element and Operation Descriptions**
	The PDP must provide features that respond to a PAP-Command message directing it to load a policy or set of policies from a secure data (/policy) store. Activate_Policy: The PDP must provide features that respond to a PAP-Command message directing it to activate (turn on) a policy or set of policies in its operational policy environment. When the command is activated, a policy must be instantiated in the operational policy store and an inactive state; otherwise, the PDP will issue a warning message. Deactivate_Policy: The PDP must provide features that respond to a PAP-Command message directing it to deactivate (turn off) a policy or set of policies in its operational policy environment. A policy must be instantiated in the operational policy store and in an inactive state when the command is deactivated; otherwise, the PDP issues a warning message. Modify_Policy: The PDP must provide features that respond to a PAP-Command message directing it to accept modifications to an existing policy in its environment. These modifications either alter or replace policies in the Policy Store. Verify_Policy: The PDP must provide features that validate and verify a set of policies, confirming that the policies are from a valid source and conform to the IEF Specifications. Store-Policy: The PDP must provide features that respond to a PAP-Command message, directing it to store its policy environment in a secure data (/policy) store.
PDP	The Policy Decision Point provides access control services (e.g., RBAC, ABAC, PBAC, or UBAC) to adjudicate data access to or release of resources to a specified user based on: 1. The sensitivity of the resource; 2. The clearances and attributes for each user; 3. (Optional) The operational context (e.g., location, device, mission, threat environment, and role) and 4. (Optional) Other security and data protection considerations of the user. The PDP may provide access control decision logic as native services within the PDP or as an interface to a user-specified decision system.

`

Table 25 - PDP Operations Elements	
Element Name	**Element and Operation Descriptions**
	Element Type: Class Owned Operations: Parse-AuthorizationRequest: The PDP must provide features that gather data from a PDP-AuthorizationRequest message and provide the data elements for the adjudication process. Stage_RequestProcess: The PDP must provide features that assess the request and stage the adjudication process of each authorization decision. The authorization request may contain multiple elements, e.g.: • Multiple information elements (e.g., a message containing a digest, packages, payloads attachments, or an email with attachments) with multiple recipients would require: ○ The Sender is to be authorized to release each information element and ○ Each Recipient is to be authorized to have access to each information element. • Multiple IEF resources (/components) are available to perform the required process. The Administrator must be authorized to access each resource and perform the requested function (/operation). • IEF component requesting resources through the SSG. Policies may change throughout 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: The PDP must provide features that gather the data elements comprising the PolicyDetermination and package it for release to the PEP or PAP to be enforced.
PolicyStore	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: The PDP must provide features that transfer a set of policies to its active and persistent policy store. Retrieve-PDP-Policy:

Table 25 - PDP Operations Elements	
Element Name	**Element and Operation Descriptions**
	The PDP must provide features that transfer a set of policies from the persistent policy store to active memory.
XACML PDP	The IEF-PDP derives from the core concepts in the XACML specification and Reference Architecture. It extends the XACML specification's focus on applications, networks, and devices into the data domain and Data-Centric Security (DCS).
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.

9.2 PDP Integration Configurations

The following clauses illustrate several configuration patterns for integrating PDP functionality into an IEF Configuration. The PDP is a message-based interface to enable users, implementors, and integrators to deploy the PDP in several configurations, including:

- An integrated component connected to the SMB;
- As an integration point to user-specified access control services (e.g., ABAC, RBAC, PBAC, and UBAC);
- As an external component connected directly to a PEP.

9.2.1 PDP Integrated Into IEF

This initial configuration provides the core PDP features and functions as an IEF element attached to the Secure Message Bus. The SMB enables the PAP and PEPs to employ the PDP to adjudicate access and release control decisions.

`

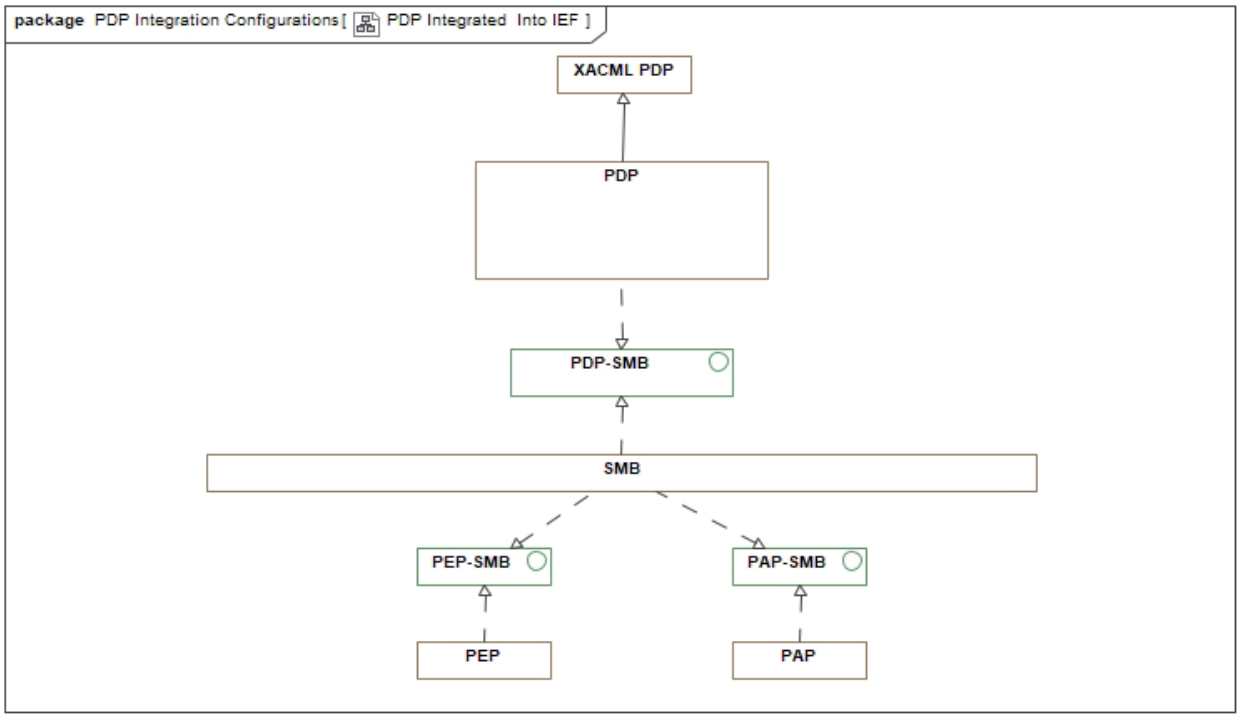


Figure 58 -PDP Integrated Into IEF

9.2.2 PDP Delivered as External Service

This initial configuration enables a user, implementor, or Integrator to interoperate with an access controls system in the users' environment (/infrastructure).

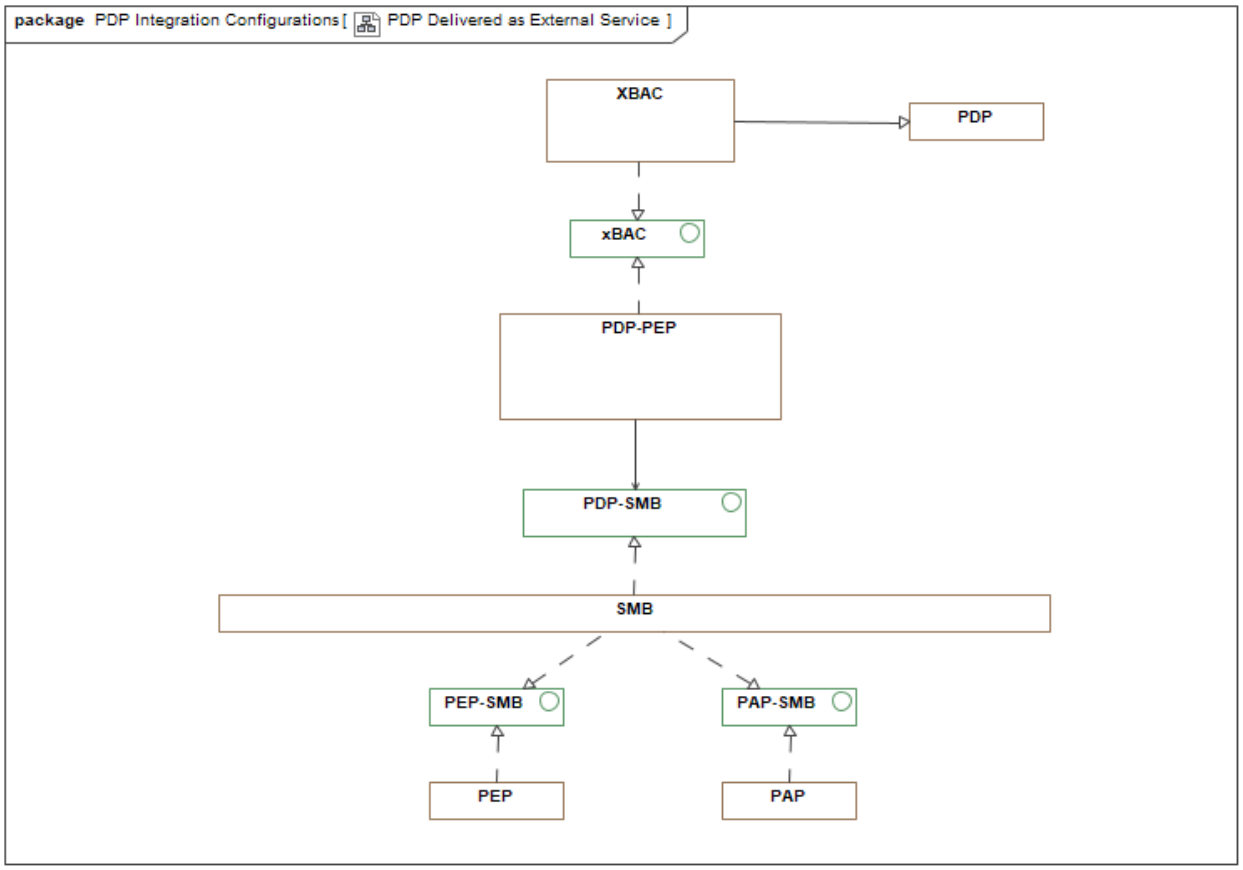


Figure 59 -PDP Delivered as External Service

`

10 Policy Enforcement Point (PEP)

The Policy Enforcement Point (PEP) intercepts each data or Information Element (e.g., message) transiting between a user (data producer or recipient) and the appropriate server (e.g., Email, Instant Messaging, File Share, and data middleware). The PEP ensures that the producer (/sender) is authorized to release and each consumer (/recipient) is authorized to access the data content. For the PEP to perform this function, each data exchange element must contain the security, confidentiality, privacy, and other user markings (/labels/metadata) required to adjudicate each exchange.

10.1 Core PEP Operations

The following figure identifies the Policy Enforcement Point (PEP) features that form part of each of the IEF-defined PEPs (e.g., Email-PEP, File-Share-PEP, Instant Messaging /chat room (IM-PEP) and DataMessaging-PEP)

The PEP offers multiple configurations to integrate IEF data sharing and safeguarding services into a Zero-Trust environment. (see IEF Component Specializations for PEP Options). However, it's important to note that the design and implementation of the variations of the PEP for integration into their desired infrastructure is the sole responsibility of the user or integrator, underscoring their crucial role in the process. The PEP may provide direct interfaces to the user (or integrator) provided security services, access the services through the Security Services Gateways (SSG), or a combination of both options. For this reason, the specification uses “may provide” versus “must provide” in each of the DirectConnect and SMBConnect options.

The IEF defines four (4) specializations for Data PEP, including:

- Email-PEP;
- File-PEP;
- IM-PEP and
- Messaging-PEP.

Each PEP specialization provides the features that address the specific data, protocol, and technology employed. The PEP also acts as the integration point between the user's security infrastructure and IEF domains.

At a high level, each PEP provides the following functions tailored to their data types and protocols:

- On the receipt of data from a user application or data system, the PEP performs the following:
    - Receive the exchange (e.g., data or information) element(s);
    - Extract the metadata bound to the exchange element(s);
    - Request the sender's attributes from the user-specified LDAP or ICAM system and
    - Request the recipient's (user/system) attributes from the user-specified LDAP or ICAM system and
    - Prepare and issue an adjudication request to the user-specified PDP or ABAC System;
    - Enforce the PDP or ABAC determination for the sending and receiving systems and users:
        - If permitted:
            - Decrypt the element content;
            - Extract the data elements: email, file, instant message, or data message, and
            - Marshal the data content to the appropriate data store(s);

        - if Denied: Blocks the receipt or release of data elements and issues an alert or warning to the user and the SIEM or continuous monitoring system or
        - If indeterminate: executes a User-defined action(s) and
    - Log the transaction to the logging service or TLS.
- On a data trigger (automated release of data), the PEP performs the following:
    - Gather the data elements d labels for each exchange element;
    - Gather attributes for the sender and each of the recipients from the user-specified LDAP or ICAM system;
    - Packages and issue an adjudication request to the user-specified PDP or ABAC system;
    - Orchestrate the enforcement of the PDP determination for each data or information element:
        - If permitted, releases the authorized elements to the requisite server, middleware, or client application;
        - If denied, block the unauthorized data elements and issue an alert or warning to the user and the SIEM or continuous monitoring system or
        - If indeterminate:, execute the user-defined action(s); and
    - Log the transaction to the logging service or TLS.
- On Data Request, the PEP performs the following:
    - Extract the request from the exchange element;
    - Authenticate the requester using their credentials or token and the LDAP or ICAM System;
    - Gather the attributes of the recipient from the LDAP or ICAM system;
    - Package up adjudication requests and issues the message(s) to the PDP;
    - Enforce the PDP determination for each received message:
        - Permit: Releases the part of the exchange to the requisite server, middleware, or client application:
            - Package the data request and pass it to the specific Data provider (e.g., email, instant messaging, or file server, of Packaging and Processing Service);
            - Receive the requested data from the data provider;
            - Gather attributes for the sender and each of the recipients;
            - Gather the data elements and labels for each data element;
            - Gather the data and labels for each data element;
            - Package up adjudication requests and issues the data element(s) to the PDP;
            - Enforce the PDP determination for each received data element:
                - If Permitted: Releases the part of the exchange to the requisite server, middleware, or client application;
                - If Denied: Block the release of the part of the exchange from the server, middleware, or client application or
                - If Indeterminate: User-defined action or
        - If Denied, Block the request and issue an Error or warning to the user and the SIEM or continuous monitoring system or

`

- If indeterminate, execute the user-defined action(s) and
- Log the transaction to the logging service or TLS.

Note: The user (/implementor) must determine which security service meets their specific needs.

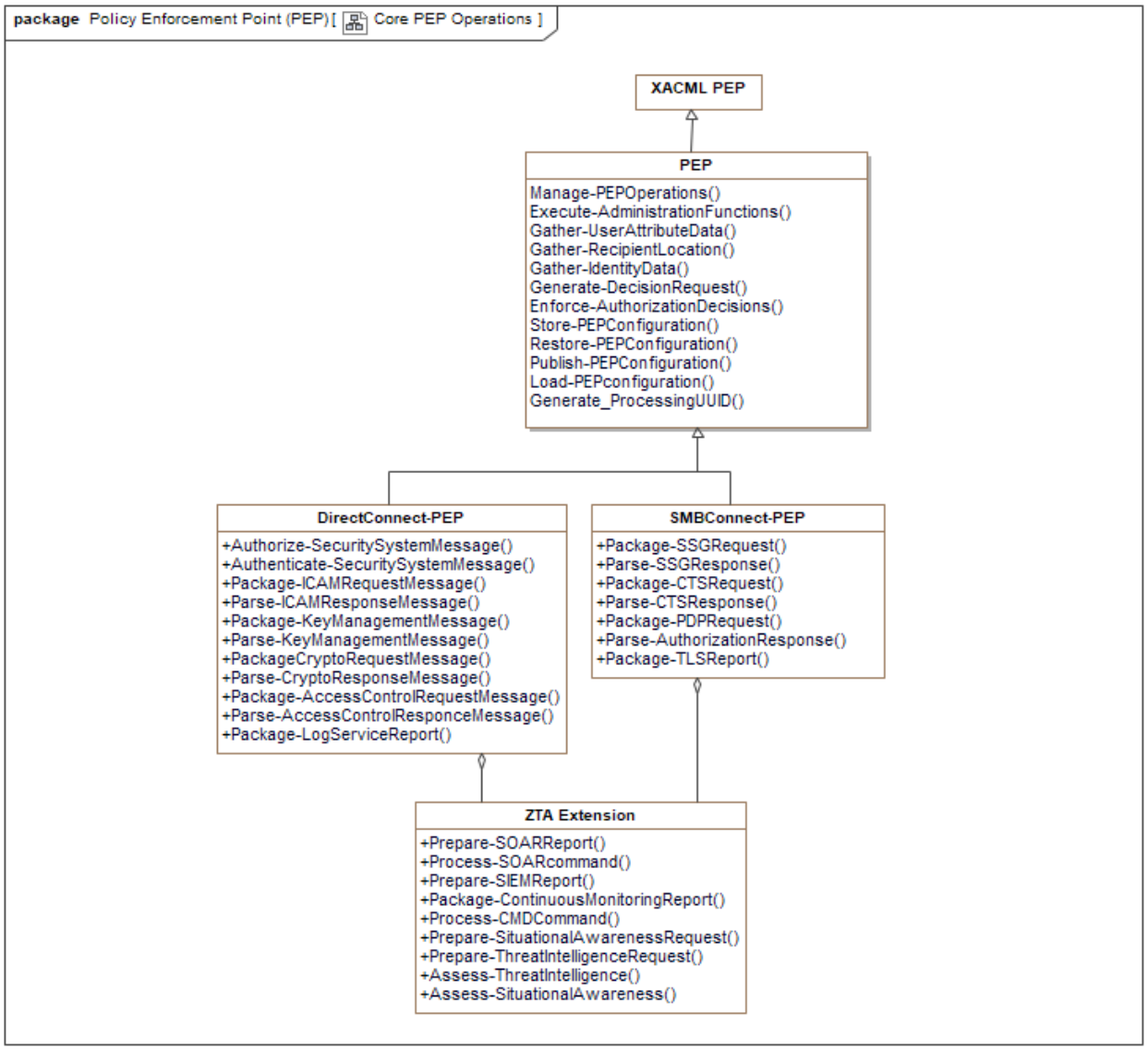


Figure 60 -Core PEP Operations

The following table identifies and describes the elements and operations illustrated in the previous figure - Core PEP Operations.

Table 26 - Core PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
DirectConnect-PEP	The PEP may allow direct access to user-provided security services. This configuration represents an alternative to those provided through SMB access to the SSG, PDP, CTS, and TLS.
	Element Type: Class Owned Operations: Authorize-SecuritySystemMessage: The PEP may provide features that authorize the receipt or release of messages from the user provided Security system, e.g.: 1. Identity, Credential, and Access Management (ICAM) system (e.g., Active Directory or LDAP); 2. Access Controls System (e.g., ABAC, RBAC, UBAC, or PBAC); 3. Cryptographic System;Key Management System; 4. Logging Service Report; 5. Continuous Monitoring Report and 6. Security Event and Information Management (SEIM) Report. Authenticate-SecuritySystemMessage: The PEP may provide features that authenticate messages from the user provided Security system, including: 1. Identity, Credential, and Access Management (ICAM) system (e.g., Active Directory or LDAP); 2. Access Controls System (e.g., ABAC, RBAC, UBAC, or PBAC); 3. Cryptographic System; 4. Key Management System; 5. Logging Service Report; 6. Continuous Monitoring Report and 7. SEIM Report. Package-ICAMRequestMessage: The PEP may provide features that gather ICAM properties (e.g., user token and request properties) and package a request message to the user-provided ICAM system. Parse-ICAMResponseMessage: The PEP may provide a feature that extracts data elements from an identity, credential, and ICAM response message. Package-KeyManagementMessage:

`

Table 26 - Core PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
	The PEP may provide features that gather key management properties and package a request message to the user-provided key management system. Parse-KeyManagementMessage: The PEP may provide features that extract data elements from a key management System response message. PackageCryptoRequestMessage: The PEP may provide features that gather the message, message metadata, and cryptographic keys and package a cryptographic service request message for the user-provided cryptographic System. Parse-CryptoResponseMessage: The PEP may provide features that extract data elements from a cryptographic system response message. Package-AccessControlRequestMessage: The PEP may provide features that gather the message, message metadata, and cryptographic keys and package a cryptographic service request message for the user-provided cryptographic System. Parse-AccessControlResponceMessage: The PEP may provide features that extract data elements from an access control (e.g., ABAC, RBAC, PBAC, or UBAC) system response message Package-LogServiceReport: The PEP may provide features that gather PEP transaction information and package a log report (/message). Inherited Operations: The above functions utilize the following inherited operations to deliver their features. Inherited functions include: Manage-PEPOperations inherited from PEP Execute-AdministrationFunctions inherited from PEP Gather-UserAttributeData inherited from PEP Gather-RecipientLocation inherited from PEP Gather-IdentityData inherited from PEP Generate-DecisionRequest inherited from PEP Enforce-AuthorizationDecisions inherited from PEP Store-PEPConfiguration inherited from PEP Restore-PEPConfiguration inherited from PEP

Table 26 - Core PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
	Publish-PEPConfiguration inherited from PEP Load-PEPconfiguration inherited from PEP Generate_ProcessingUUID inherited from PEP inherited from IEF_Component 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
PEP	The Policy Enforcement Point (PEP) intercepts all user and system requests to access or release data elements protected by IEF services. The PEP gathers information about the data elements, sender, recipients, and communication channel, packages an adjudication request to the PDP, and enforces the PDP access or release determination. In a zero-trust environment, the PEP interface authenticates and authorizes each transaction with an external cryptographic service. The PEP must provide one or both of the feature sets defined by the SMB or Direct Connect options.
	Element Type: Class Owned Operations: Manage-PEPOperations: The PEP must provide features that manage the execution of PEP functions in each potential configuration (e.g., Direct connect, SMB connection, or a combination of direct and SMB configurations). Execute-AdministrationFunctions: The PEP must provide features that execute AdministrativeCommands from an authorized Policy Administration Point. PEP administrative functions include: 1. Activate PEP features; 2. Deactivate PEP features;

`

Table 26 - Core PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
	3. Configure PEP Parameters; 4. Archive PEP Operational Environment; 5. Publish PEP Configuration; 6. Store PEP Configuration; and 7. Retrieve PEP Configuration; Gather-UserAttributeData: The PEP must provide features that gather the sender and receiver attributes to determine their authorization to receive data elements. Gather-RecipientLocation: (Optional) The PEP provides features that gather information about the recipients' location (physical or electronic). A recipient's location(s) (e.g., network location, physical location, and device) may impact the user's attributes and the information content they are authorized to receive. These features may only be available if the IEF can request the information from the user's situational awareness, incident management, or network management systems. Gather-IdentityData: The PEP must provide features that gather the identity information for the sender and recipients of the specified information element (s). These features allow users to request this information from the infrastructure services that provide identity management. All requests to the users' specified infrastructure are issued through the Security Services Gateway using an SSG-Request message. Generate-DecisionRequest: The PEP must provide features that generate a decision request to the PDP. Enforce-AuthorizationDecisions: The PEP Provides features that enforce PDP authorization decisions for each data receipt and release. Store-PEPConfiguration: The PEP Provides features that gather and store its configuration parameters in local storage. Restore-PEPConfiguration: The PEP Provides features that retrieve and load its configuration parameters from local storage. Publish-PEPConfiguration:

Table 26 - Core PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
	The PEP must provide features that gather and publish its configuration parameters to the PAP. Load-PEPconfiguration: The PEP must provide features that receive and load its configuration parameters from the PAP. Generate_ProcessingUUID: Upon receiving a message, the PEP must provide features that generate a universally unique identifier (UUID). The structure of the UUID must identify the node receiving the data. If the received message includes a UUID, the PEP must adopt the received UUID. Inherited Operations: The above functions utilize the following inherited operations to deliver their features. Inherited functions include: inherited from IEF_Component 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
SMBConnect-PEP	The SMB-PEP may allow access to all security services and data through the SMB and its access to the SSG, PDP, CTS, and TLS-enabled interfaces.
	Element Type: Class Owned Operations: Package-SSGRequest: The PEP may provide features that package Service Request Data as an SMB-Message directed to SSG to request information from the user's security infrastructure, e.g.: • To request Sender Identity Information from the User specified Identity Management Services;

`

Table 26 - Core PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
	• 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 • To request Cryptographic Keys from the user-specified Key Management system or service. Parse-SSGResponse: The PEP may provide features that parse an SSG-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; • Receiver Privileges and • Operational Context Data. Package-CTSRequest: The PEP may provide features that package the cryptographic keys and information elements as a CTS-Request for the CTS to encrypt or decrypt the information element. Parse-CTSResponse: The PEP may provide features that parse a CTS-Response message and extract the transformed information element. Package-PDPRequest: The PEP may provide the features to package a PDP-Request message, including: • Sender Attributes; • Recipient Attributes; • Resource Metadata (e.g., confidentiality labels); and • Environmental Data (optional): ○ Communication Channel Attributes; ○ Sender Device Attributes; ○ Recipient Device Attributes; ○ Sender Application Attributes; and

Table 26 - Core PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
	○ Recipient Application Attributes. Parse-AuthorizationResponse: The PEP may provide features that parse a PDP-Response message and extract the data necessary for other PEP features to enforce the PDP determinations. Package-TLSReport: The PEP may provide features that gather PEP transaction information and package a TLS-LogMessage. Inherited Operations: The above functions utilize the following inherited operations to deliver their features. Inherited functions include: Manage-PEPOperations inherited from PEP Execute-AdministrationFunctions inherited from PEP Gather-UserAttributeData inherited from PEP Gather-RecipientLocation inherited from PEP Gather-IdentityData inherited from PEP Generate-DecisionRequest inherited from PEP Enforce-AuthorizationDecisions inherited from PEP Store-PEPConfiguration inherited from PEP Restore-PEPConfiguration inherited from PEP Publish-PEPConfiguration inherited from PEP Load-PEPconfiguration inherited from PEP Generate_ProcessingUUID inherited from PEP inherited from IEF_Component 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 26 - Core PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
XACML PEP	The IEF-PEP derives from the core concepts in the XACML specification and Reference Architecture. It extends the XACML specification's focus on applications, networks, and devices into the data domain and Data-Centric Security (DCS).
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
ZTA Extension	The PEP may provide a ZTA extension that enables the PEP to interoperate within a Zero Trust Architecture and is required for logging, monitoring, and auditing. User-provided services in the security infrastructure. This configuration represents an alternate configuration to those provided through SMB access to the SSG, PDP, CTS, and TLS. The ZTA extensions seek to enable the PEP and other IEF components to react to changes in the current situation or operational context. These features communicate with the user's incident management, situational awareness, CDM, and SEIM systems to gather the information that may influence the access to or the release of information, including the users - e.g., role and responsibility, phase of the operation, operational and cyber threats, operational risk, command Intent, user location, communications links, and access device. When integrated into the Access Control Request Message to the PDP (or future AI capability), these contextual descriptions can significantly impact the policy handling related to sensitive information and the quality of service available to the recipients.
	Element Type: Class Owned Operations: Prepare-SOARReport: The PEP may provide features that gather PEP transaction data and package a Security Orchestration, Automation, and Response (SOAR)) reports (/message). Process-SOARcommand: The PEP may provide features that receive a command from a SOAR System and adjust PEP operations or policy to conform to the new requirements. Prepare-SIEMReport: The PEP may provide features that gather PEP transaction data and package a Security Information and Event Management (SIEM) report (/message).

Table 26 - Core PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
	Package-ContinuousMonitoringReport: The PEP provides features that gather PEP transaction information and package a Continuous Diagnostic and Mitigation (CDM) report (/message). Process-CMDCommand: The PEP may provide features that receive a command from a CDM System and adjust PEP operations to conform to the new requirements. Prepare-SituationalAwarenessRequest: The PEP may provide features that gather situational information from the user's environment and systems. Prepare-ThreatIntelligenceRequest: The PEP may provide features that gather threat intelligence from user-specified systems. Assess-ThreatIntelligence: The PEP may provide features that process and assess threat intelligence information and use that information to administer access and release control policy enforcement to mitigate the threat. Assess-SituationalAwareness: The PEP may provide features that process and assess situational information that is used to administer access and release control policy enforcement to address changes in the operational context.

10.2 Data PEPs

Each Data PEP specialization provides the features and functions needed to operate on the specific data, protocol, and technology employed. The PEP acts as the integration point between the user and IEF domains.

On the receipt of data, the PEP:

- Receives the exchange element (e.g., email, instant message, file or messaging)
- Extracts the metadata bound to the element;
- Enforces the PDP determination for the Message:
    - Permit: Enables the receipt of the exchange element;
    - Deny: Blocks the receipt of the exchange element and issues a warning to the Trusted Logging Service or
    - Indeterminate: User-defined action;
- Decrypt the exchange element;
- Extract the exchange element parts:

`

- Email: Email Body and each attachment;

  File: Each File;
- Instant (Chat) Message: Each Message and Each Attachment and
- Structured Message: The Digest, each Information Package, each Data Payload, and each Attachment;

- the receipt of the data content by the IEF node;
- Packages and issues adjudication requests to the Policy Decision Point;
- Enforces the PDP determination for each received data element:
  - Permit: Releases the part of the exchange to the requisite server, middleware, or Client;
  - Deny: Blocks the release of the part of the exchange from the server, middleware, or client application or
  - Indeterminate: User-defined action;
- Process the authorized content of the message per message type and
- Packages data content per user needs and attributes; Packages and issues adjudication requests to the Policy Decision Point for each recipient; Authorizes the release of data to each recipient; Logs its actions. The PEP may be able to redact information elements based on release instructions from the PDP.

10.2.1 Email PEP

An IEF Email PEP intercepts each email message as it transits between the client and 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 authorization to send the specific email body 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 attributes are again verified prior to any decryption of the email's content.

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

10.2.1.1 Email PEP Operations

The following figure identifies the 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:

1. Each participant is authorized to send or receive the information in the email;
2. Each information element (email body and attachments) is appropriately marked and
3. Each information element is appropriately protected.

The Email PEP is a proxy service that directs email traffic to the data protection and security services that enforce protection and security policy logic.

When a user sends a request to the Mail server to retrieve a new email, the PEP:

- Intercepts the request and verifies that the user is authorized to access the server;
- The request is routed to the server if the user is authorized to access the server. The PEP then intercepts each mail message for the user retrieved from the server. It verifies that the user is authorized to receive the message body and each attachment. 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
        - Routes the email to the user's email 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
        - Routes the email to the user's email client;
    - Reports the results of the transaction to the TLS and
    - If the policy requires, alert the administrator to each issue with the received email.

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

- Intercepts the mail message from the email 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
        - Routes it to the email 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, 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*];
- Reports the results of the transaction to the TLS and
- If the policy requires, alert the administrator to each issue with the sent email.

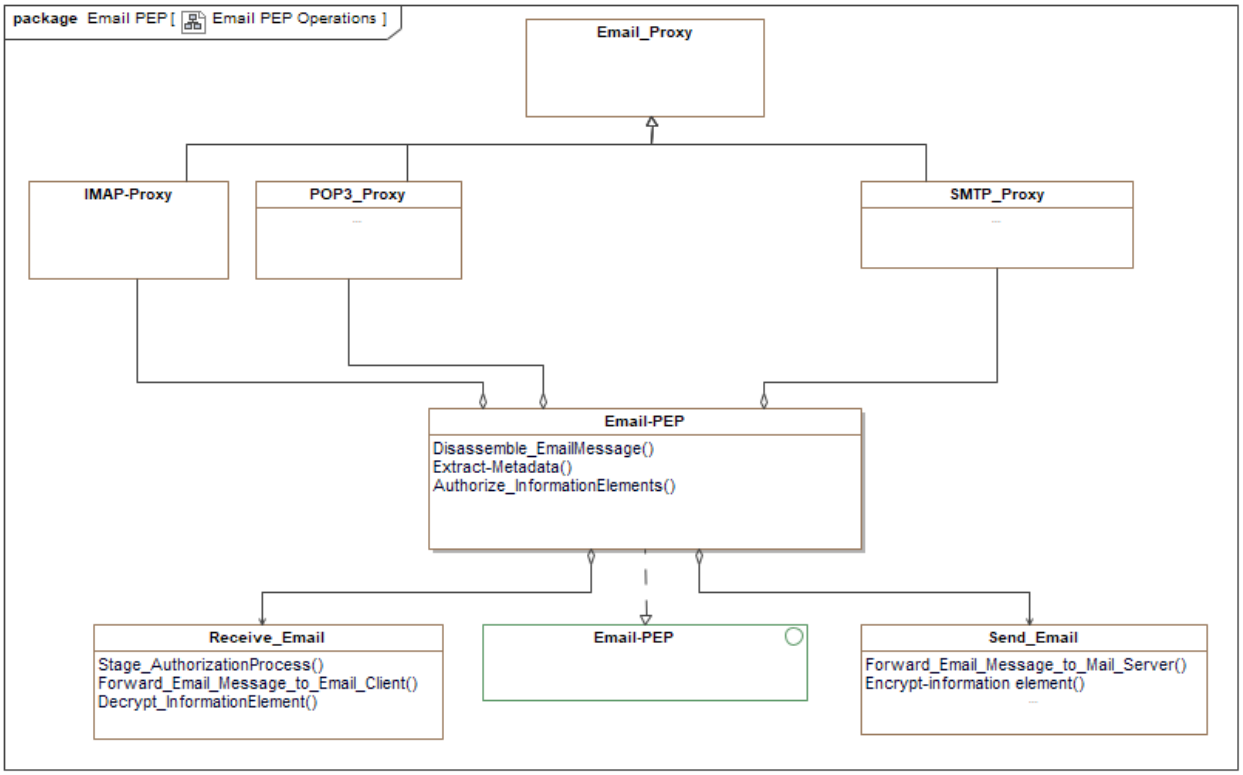


Figure 61 -Email PEP Operations

The following table identifies and describes the elements and operations illustrated in the previous figure - Email PEP Operations.

Table 27 - Email PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
Email-PEP	The Email-PEP operates as a proxy between the email client and the mail server.
	Element Type: Class Owned Operations: Disassemble_EmailMessage: The Email-PEP must provide features that disassemble a mail message and extract each of the enclosed information elements (body and attachment). Extract-Metadata: The Email-PEP must provide features that gather *Metadata* from the email header and each of the attachments contained in the Email message. The data collected includes:

Table 27 - Email PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
	• information element(s) metadata; and • User (Producer & recipient) Identifier (Email Address). Authorize_InformationElements: The Email-PEP must provide 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. Inherited Operations: The above functions utilize the following inherited operations to deliver their features. Inherited functions include: Manage-PEPOperations inherited from PEP Execute-AdministrationFunctions inherited from PEP Gather-UserAttributeData inherited from PEP Gather-RecipientLocation inherited from PEP Gather-IdentityData inherited from PEP Generate-DecisionRequest inherited from PEP Enforce-AuthorizationDecisions inherited from PEP Store-PEPConfiguration inherited from PEP Restore-PEPConfiguration inherited from PEP Publish-PEPConfiguration inherited from PEP Load-PEPconfiguration inherited from PEP Generate_ProcessingUUID inherited from PEP inherited from IEF_Component 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 27 - Email PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
	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 attachment) using email and 2. The recipients are authorized to receive the included content.
	Element Type: Class Owned Operations: InterceptEmailMessage: RouteEmailMassageToServer:
IMAP-Proxy	The IMAP proxy is an intermediary service connecting the mail client to the mail server. The e-mail client connects to the proxy server, which integrates IEF DCS services into the path. The IEF services ensure the emails sent to the server are authorized to the assigned recipients. The IMAP-Proxy routes all email messages to and then receives the authorized emails from the IMAP interface with the email-PEP.
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration. Inherited Operations: The above functions utilize the following inherited operations to deliver their features. Inherited functions include: InterceptEmailMessage inherited from Email_Proxy RouteEmailMassageToServer inherited from Email_Proxy
POP3_Proxy	The POP proxy is an intermediary service connecting the mail client to the mail server. The e-mail client connects to the proxy server, which integrates IEF DCS services into the path. The IEF services ensure the emails sent to the server are authorized to the assigned recipients. The POP-Proxy routes all email messages to and then

Table 27 - Email PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
	receives the authorized emails from the POP interface with the email-PEP.
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration. Inherited Operations: The above functions utilize the following inherited operations to deliver their features. Inherited functions include: InterceptEmailMessage inherited from Email_Proxy RouteEmailMassageToServer inherited from Email_Proxy
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 must provide features that queue each received mail message for authorization and decryption. As part of this process, the PEP performs the following: 1. Parse the mail message and extracts the enclosed information elements (body and Attachment(s)); 2. Verify the mail message is transmitted to the intended recipient - if not, alert the administrator and terminate the message processing. 3. Collect the metadata describing the sensitivity (e.g., Security Level and Caveats) from each of the enclosed information elements; 4. (optional) Request additional information (e.g., Operational Context and User Location); 5. Request the recipient's attributes and privileges from the users' security services and infrastructure (via the SSG); 6. Gathers metadata (e.g., security level, and caveats) from each of the information elements in the message; and 7. Package 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 recipient's attributes 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. Extract the PDP decisions and instructions from the PDP Response.

`

Table 27 - Email PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
	2. Verify that the recipient is authorized to receive the content. If not, execute 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. Repackage the mail message with only the elements that the PDP has authorized for receipt, 5. Issue the repackaged mail message to the User's Email Client Application. 6. Log the transaction to the TLS. This process is repeated for each received email. Forward_Email_Message_to_Email_Client: The Email-PEP must provide features that forward authorized mail messages to the user`s email client application. Decrypt_InformationElement: The PEP must provide features that stage the cryptographic transformation of each authorized information element (body and attachments). For each authorized information element enclosed in the mail Message, the PEP: 1. Requests the unique symmetric cryptographic key from the user`s Key Management services through a request to the SSG - 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 2. Receives the decrypted Information Element from the CTS-Response.
Send_Email	Email-PEP must provide 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 must provide features that queue the (sent) mail message for authorization and encryption. As part of this process, the PEP executes the following:

Table 27 - Email PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
	1. Parse the mail message and extract the enclosed information elements (body and Attachment(s)); 2. Verify that the mail message is transmitted to the intended recipient, if not, alert the administrator (see PAP-AlertWarning message) and terminate the message's processing. 3. Collect the metadata describing the sensitivity (e.g., Security Level and Caveats) from each of the enclosed information elements; 4. (optional) Request additional information (e.g., Operational Context and User Location); 5. Request each recipient's attributes and privileges from the user's security services and infrastructure (via the SSG); 6. Gather metadata (e.g., security level, and caveats) from each of the information elements in the message; and 7. Package a PDP-Request message and issue 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' attributes 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. Extract the PDP decisions and instructions from the PDP response. 2. Verify that each recipient is authorized to receive each of the information elements, if not, apply the PDP instructions*. 3. Stages** the decryption of the information elements authorized for release to the recipient. The PEP: ○ 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' attributes 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:

`

Table 27 - Email PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
	1. Extract the PDP decisions and instructions from the PDP response. 2. Verify that each recipient is authorized to receive each of the information elements, if not, apply the PDP 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 attributes, e.g.: 1. Terminate processing, return the email to the sender with an error message, and have the sender correct the issue. 2. Redact the offending information element, send the remaining information elements to the recipients, and send a warning message to the sender identifying the issue. 3. Send the email to only the recipients authorized to get all the information elements - 4. Send an error message to the sender identifying the issue. * User policy that governs PEP action. The PDP should provide these instructions 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 must provide features that forward authorized mail messages to the email server application. Encrypt-information element: The PEP must provide features that stage the cryptographic transformation of an email body and its attachments. For

Table 27 - Email PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
	each information element, the PEP enforces PDP instructions for encrypting information elements in a mail message. For each information element in the email message, the encrypted information element operations perform the following functions: 1. Requests a unique symmetric cryptographic key from the Key Generation Service (KGS); 2. Packages an SMB message containing the encryption key and the information element contained in the Email message; 3. Sends the message to the message specified CryptographicTransformationService (CTS); 4. Receives the encrypted Information Element; 5. Sends the information element data and the cryptographic key to the Key Escrow Service (KES); 6. Receives the Key Token from the KES and 7. 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: • 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 to the mail server for delivery.
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration. Inherited Operations: The above functions utilize the following inherited operations to deliver their features. Inherited functions include: InterceptEmailMessage inherited from Email_Proxy RouteEmailMassageToServer inherited from Email_Proxy

`

10.2.1.2 Email PEP Configurations

As illustrated in the following figure, the Email PEP forms a proxy between the user email clients and servers. It intercepts each email and channels it through the IEF-Services, which apply data protection to the content of the email and attachments.

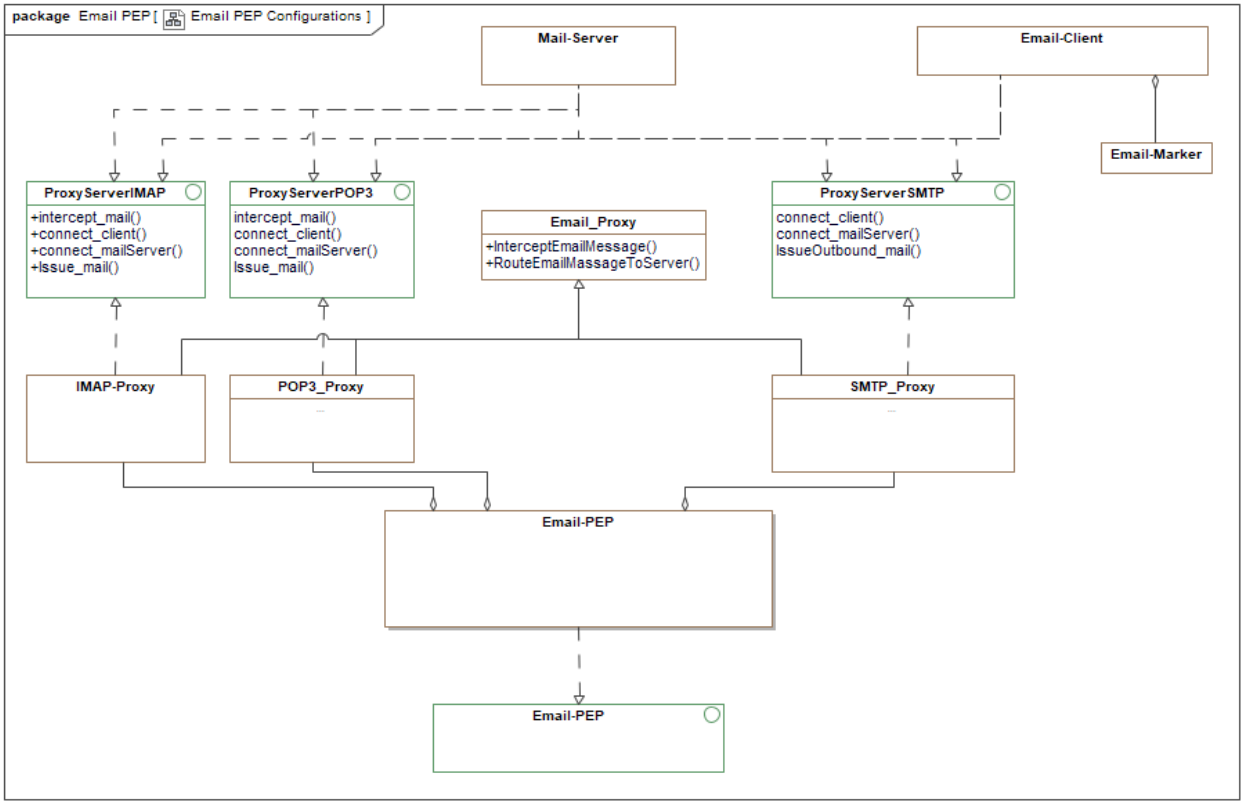


Figure 62 -Email PEP Configurations

The following table describes the elements illustrated in the previous figure - Email PEP Configurations.

Table 28 - Email PEP Configurations Elements	
Element Name	**Element Descriptions**
Email-Client Type: Class	The email client is a standard commercial off-the-shelf application selected by the user to access and generate 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 28 - Email PEP Configurations Elements	
Element Name	**Element Descriptions**
Email-Marker Type: Class	The Email Marker (or labeler) is a software service that integrates with off-the-shelf Email clients and obliges the authorized user to mark the email appropriately 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.
Email-PEP Type: Class	The Email-PEP operates as a proxy between the email client and the mail server.
Email_Proxy Type: Class	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 attachment) using email and 2. The recipients are authorized to receive the included content.
IMAP-Proxy Type: Class	The IMAP proxy is an intermediary service connecting the mail client to the mail server. The e-mail client connects to the proxy server, which integrates IEF DCS services into the path. The IEF services ensure the emails sent to the server are authorized to the assigned recipients. The IMAP-Proxy routes all email messages to and then receives the authorized emails from the IMAP interface with the email-PEP.
Mail-Server Type: Class	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 (e.g., 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.

`

Table 28 - Email PEP Configurations Elements	
Element Name	**Element Descriptions**
	The IEF services do not alter the standard e-mail headers or protocols. However, the content (body and attachments) of the e-mail message is encrypted. The IEF ensures that all messages are only provided to authorized recipients, appropriately marked, and encrypted, both at rest and in motion.
POP3_Proxy Type: Class	The POP proxy is an intermediary service connecting the mail client to the mail server. The e-mail client connects to the proxy server, which integrates IEF DCS services into the path. The IEF services ensure the emails sent to the server are authorized to the assigned recipients. The POP-Proxy routes all email messages to and then receives the authorized emails from the POP interface with the email-PEP.
SMTP_Proxy Type: Class	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: • 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 to the mail server for delivery.

10.2.2 File PEP

The File Sharing PEP protects individual files as they are stored on 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, caveat (warning orders), and supporting metadata. The markings are permanently bound to the files at rest, in transit, and in use, while the IEF protects them.

The IEF operates between the file management client and the protected file share. It 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.

10.2.2.1 File PEP Operations

The following figure identifies the core 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 all released files are bound with confidentiality marks (/labels), the key-token, and other supporting metadata.

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

As a simple process, the PEP:

1. Extracts the pertinent information (e.g., user Identification, request type, source, and target location) from the request message;
2. Gathers the user's attributes and privileges, role, responsibilities, location, and any other information required for the PDP to adjudicate the request;
3. Requests the file(s) and extracts its metadata (tags and labels) from the SAC header;
4. Packages a PDP request and issues it to the PDP to be adjudicated;
5. Receives the PDP's determination and enforces that decision;

If the user is authorized to access the file(s), the PEP:

1. Requests the cryptographic key(s) for the file(s);
2. Orchestrates the decryption of the file(s);
3. 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 SSG.

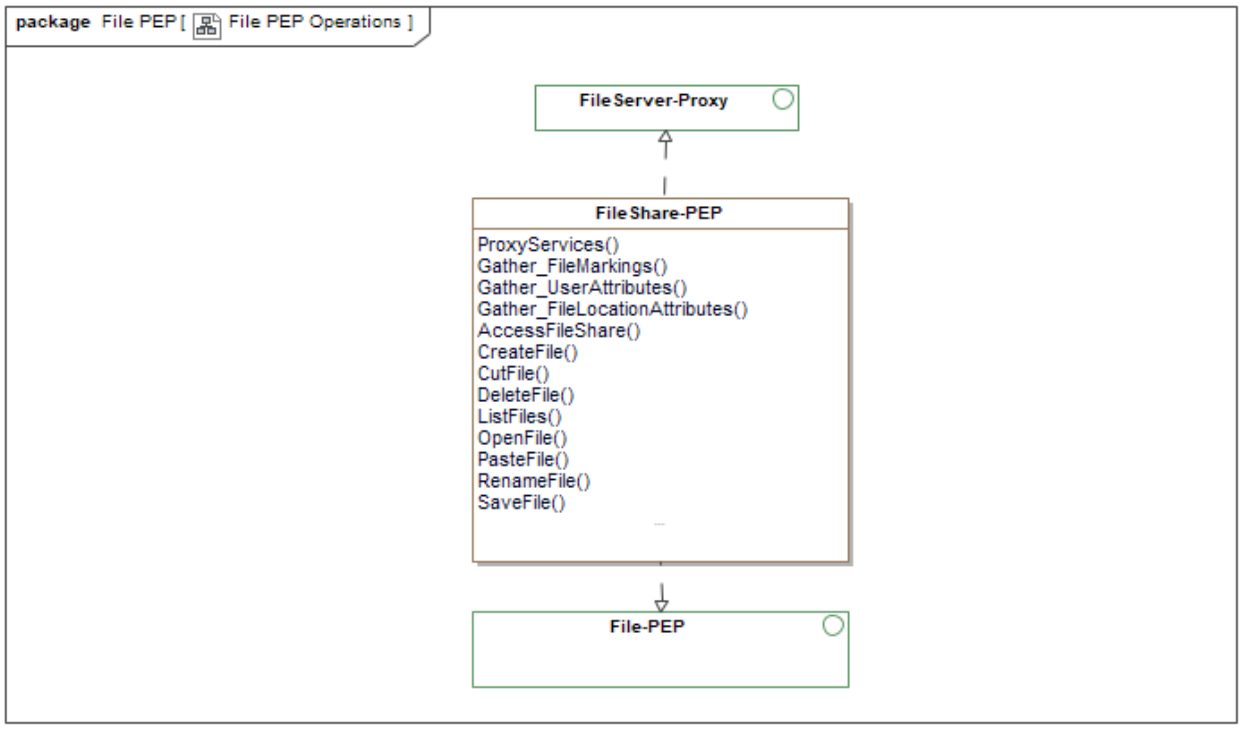


Figure 63 -File PEP Operations

`

The following table identifies and describes the elements and operations illustrated in the previous figure - File PEP Operations.

Table 29 - File PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
FileShare-PEP	The File-PEP operates between user applications and the file server. The PEP: 1. Intercepts each file-based action (e.g., open, save, save-as, move, and delete); 2. Authenticates the user; 3. Extract the metadata for each involved file; 4. Gathers user attributes; 5. Composes a PDP adjudication request; 6. Issues the adjudication request to the PDP; and 7. Enforce the PDP receipt and release determinations. (Optional) The File-PEP may redact or interact with the PPS to tailor the content to the recipients' attributes.
	Element Type: Class Owned Operations: ProxyServices: The File-PEP must provide proxy services between data providers and consumers and the protected file share. Gather_FileMarkings: The File-PEP must provide features to extract file metadata (/labels) from the metadata binding (e.g., SAC, AdatP-4778, ACP-240, or IC-TDF). Gather_UserAttributes: The File-PEP must provide features to retrieve the user's identity from the requesting application. The PEP uses this data to authenticate the user and authorize the user's rights to access requested files from the file store or post a file to the file store. Gather_FileLocationAttributes: The File-PEP must provide features to extract file location information from the application request and request location access restrictions from the user's directory services (e.g., LDAP or ICAM) via the SSG. AccessFileShare: The File-PEP must provide features that intercept a user application's request to access a specified file/share

Table 29 - File PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
	(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., files and folders) that the user is authorized to access. The PEP performs the following: • Intercept the request; • Extract the user's identity (/toke) from the request; • Package and issue a request for the user's attributes (e.g., ICAM or LDAP request), as well as those from the specified file share; • Package and issue a PDP or ABAC request; • If authorized, provide the file name and attributes to the user application and • Log the Transaction to the TLS. **Note**: Any request for additional services may be made through the SMB or to the user-specified services. CopyFile: The File-PEP must provide 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 performs the following: • Intercept the request; • Extract the user's identity (/token) from the request; • Request the user attributes from the user-specified LDAP or ICAM system; • Extract the markings (/labels) from the SAC envelope header; • Package and issue an adjudication request to the ICAM or LDAP system; • If SAC is not present: ○ Request that the User provide sensitivity marking and other supporting metadata needed to create a SAC and ○ Generate a request for encryption of the file and creation of its SAC; • Package and issue an adjudication request to the PDP or ABAC services; • If the operation is authorized, the PEP determines if the security policies necessitate that the duplicate file(s) require a new cryptographic key.; • If a key is involved, request a key and issue the packages to the CTS for processing;

`

Table 29 - File PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
	• Hold the encrypted file in a temporary storage location until the paste operation is requested and • Log the transaction to the TLS. Notes: 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). Any request for additional services may be made through the SMB or to the user-specified services. CreateFile: The File-PEP must provide features that intercept each request to create a file (of type) at a specified location in the IEF-protected file share. The PEP performs the following: • Intercept the request; • Extract the user's identity from the request; • Create the information element (file); • Request the user attributes from the user-specified LDAP or ICAM system; • Request the sensitivity level and restriction marking to be placed on the file. • Package and issue an adjudication request to the PDP or ABAC Service; • If the operation is authorized, package a 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; • Log the transaction to the TLS. **Note**: Any request for additional services may be made through the SMB or the user-specified services. CutFile: The File-PEP must provide features that intercept each request to cut a file or files from one location to another (move the file to the specified location); the PEP performs the following functions PEP: • Intercepts the request; • Extract the user's identity from the request; • Extract the markings contained in the SAC Envelope Header;

Table 29 - File PEP Operations Elements	
Element Name	**Element and Operation Descriptions**

	• Package and issue an SSG-Request for required user attributes;
	• Request the user's attributes from the LDAP or ICAM system;
	• If the markings (/labels) are not present:
	  ○ request that the User provide sensitivity marking and other supporting metadata; and
	  ○ Generates a CTS-Request for encryption of the file and creation of its SAC;
	• Package and issue an adjudications request to the PDP or ABAC to access and remove the source copy;
	• If the operations are authorized, determine if the security policies necessitate that the duplicate file(s) require a new cryptographic key.
	• If a new cryptographic key (new SAC) is required, acquire a key and send the package to the CTS for processing;
	• Hold the encrypted file and metadata in a temporary store until the paste operation is requested and
	• Log the transaction to the TLS.
	Note: Any request for additional services may be made through the SMB or the user-specified services.
	DeleteFile:
	The File-PEP must provide 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 performs the following:
	• Intercept the request;
	• Extract the user's identity from the request;
	• Request the user attributes from user-specified LDAP or ICAM system;
	• Prepare and issue a request for user attributes to the Device and folder.
	• If the SAC is present, extract the file markings from the envelop header.
	• Package and issue an adjudication request to the PDP or ABAC.
	• If authorized, direct the file-share to remove the files.
	• Log the transaction to the TLS.
	Note: Any request for additional services may be made through the SMB or the user-specified services.
	ListFiles:

`

Table 29 - File PEP Operations Elements	
Element Name	**Element and Operation Descriptions**

	The File-PEP must provide features that intercept each request to list the contents of a device/folder. When a user requests a folder listing, the PEP performs the following:
	• Intercept the request;
	• Extract the user's identity from the request;
	• Request the user's attributes from the LDAP or ICAM system;
	• Request the list of file names and attributes from the secure file share;
	• Package and issue an adjudication request to the PDP or ABAC services to authorize access to the files;
	• For each file:
	  ○ Verify the user is authorized to see the file; (PDP-Response); and
	  ○ If not authorized, redact the file names and attributes from the dataset;
	• Send the redacted dataset to the File Manager Client and
	• Package and send a transaction report to the TLS.
	Note: Any request for additional services may be made through the SMB or the user-specified services.
	OpenFile:
	The File-PEP must provide features that intercept a user application's request to open a specified file (device: folder/file). On receipt, the PEP performs the following:
	• Intercept the request;
	• Extract the user's identity from the request;
	• Request the user's attributes from the LDAP or ICAM system;
	• Package and issue an SSG-Request for additional resource authorization requirements and restrictions that may apply;
	• Request the SAC from the files-share;
	• Extract the file metadata from the Envelop header;
	• Package and issue a PDP-Request to authorize access to the files;
	• If access is authorized (PDP-Response):
	  ○ Package and issue a CTS-Request containing the SAC and

Table 29 - File PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
	○ Send the decrypted file (CTS-Response) to the User Application and • Package and send a transaction report to the TLS. **Note**: Any request for additional services may be made through the SMB or the user-specified services. PasteFile: The File-PEP must provide 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 performs the following: • Intercept the request; • Extract the user's identity from the request; • Request the user's attributes from the LDAP or ICAM system; • Package and issue a request for additional resource authorization requirements and restrictions that may apply to the target Device:/Folder; • Package and issue a PDP request to authorize the operation of the files; • If the operation is authorized, copy and store the file in the specified location (device:/folder); • If the operation that placed the SAC in the temporary store was the "Cut" operation, delete the file from the original location and • Package and send a transaction report to the TLS. **Note**: Any request for additional services may be made through the SMB or the user-specified services. RenameFile: The File-PEP must provide features that intercept each request to rename a file. When a user requests that a file be renamed, the PEP: • Intercepts the request; • Extracts the user's identity from the request; • Request the user's attributes from the LDAP or ICAM system; • Packages and issues an SSG-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, package a CTS-Request to create a new SAC with the new file name;

`

Table 29 - File PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
	• 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. **Note**: Any request for additional services may be made through the SMB or the user-specified services. SaveFile: The File-PEP must provide 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 user's identity from the request; • Request the user's attributes from the LDAP or ICAM system; • Packages and issues an SSG-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 of the files; • If the operation is authorized, copy and store the file to the specified location (device:/folder); and • If the operation that placed the SAC in the temporary store was the "Cut" operation, delete the file from the original location. • Packages and sends a transaction report to the TLS. **Note**: Any request for additional services may be made through the SMB or the user-specified services. Inherited Operations: The above functions utilize the following inherited operations to deliver their features. Inherited functions include: Manage-PEPOperations inherited from PEP Execute-AdministrationFunctions inherited from PEP Gather-UserAttributeData inherited from PEP Gather-RecipientLocation inherited from PEP Gather-IdentityData inherited from PEP Generate-DecisionRequest inherited from PEP Enforce-AuthorizationDecisions inherited from PEP Store-PEPConfiguration inherited from PEP

Table 29 - File PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
	Restore-PEPConfiguration inherited from PEP Publish-PEPConfiguration inherited from PEP Load-PEPconfiguration inherited from PEP Generate_ProcessingUUID inherited from PEP inherited from IEF_Component 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

10.2.2.2 File PEP Configurations

As illustrated in the following figure, the File PEP forms a proxy between user applications, file managers, and the file server. It intercepts each request and releases and channels it through the IEF-Services, which apply data protection to the files' content.

`

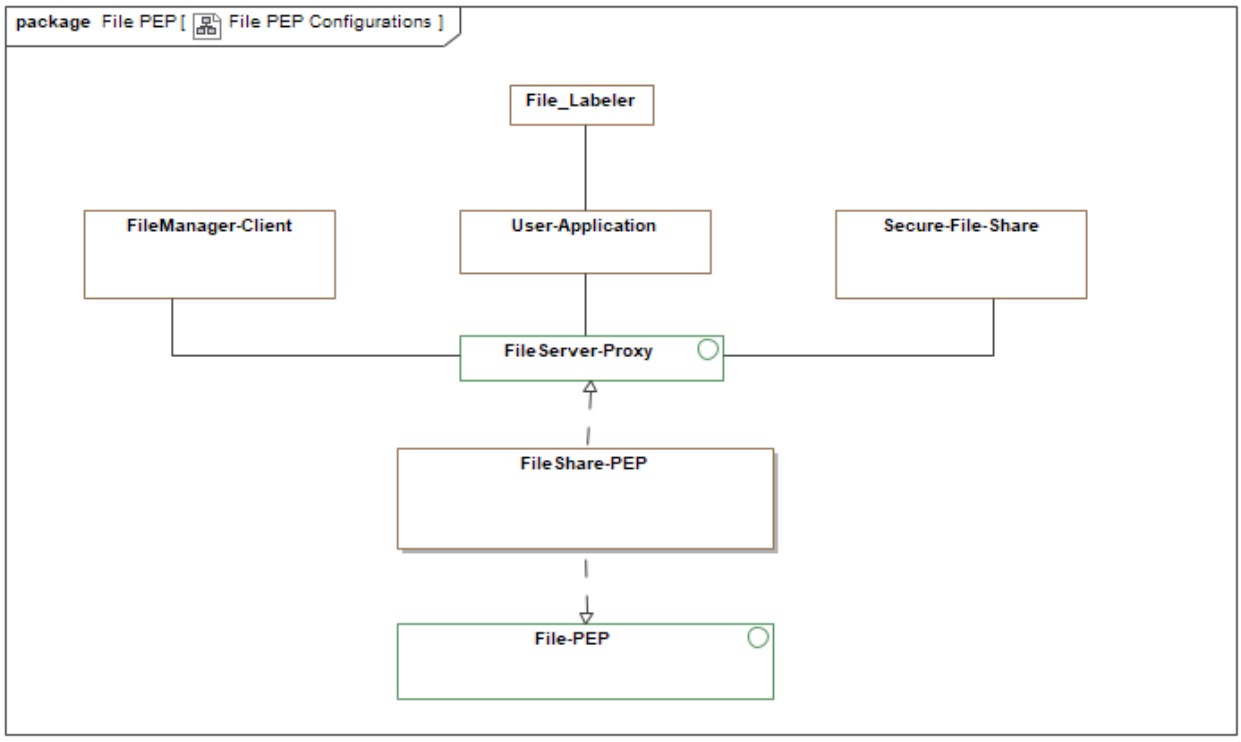


Figure 64 -File PEP Configurations

The following table describes the elements illustrated in the previous figure - File PEP Configurations.

Table 30 - File PEP Configurations Elements	
Element Name	**Element Descriptions**
File_Labeler Type: Class	The File Marker is a software service that integrates with off-the-shelf user applications and obliges (/prompts) the authorized user to mark each file appropriately with the required markings. The IEF requires that all elements of the email be appropriately marked (Tagged or labeled) in order to apply access and release policies.
FileManager-Client Type: Class	An off-the-shelf file manager or file browser provides an application and interface for managing 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 permissions.
FileShare-PEP Type: Class	The File-PEP operates between user applications and the file server. The PEP: 1. Intercepts each file-based action (e.g., open, save, save-as, move, and delete); 2. Authenticates the user;

Table 30 - File PEP Configurations Elements	
Element Name	**Element Descriptions**
	3. Extract the metadata for each involved file; 4. Gathers user attributes; 5. Composes a PDP adjudication request; 6. Issues the adjudication request to the PDP; and 7. Enforce the PDP receipt and release determinations. (Optional) The File-PEP may redact or interact with the PPS to tailor the content to the recipients' attributes.
Secure-File-Share Type: Class	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.
User-Application Type: Class	Any user application that: • Access files stored by the IEF-protected File share; or • Access data sharable using structured messages.

10.2.3 Instant Messaging (Chat) 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 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.

10.2.3.1 Instant Messaging PEP Operations

The following figure identifies the 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 information. Elements at the sensitivity and protection levels are assigned a dedicated chat or chat room. The IM-PEP is implemented as a proxy architecture where the proxy redirects IM Traffic to the IEF services, which enforces the relevant policy and protection logic.

`

The IM-PEP features intercept each transaction between the IM client and server and ensure that the sender and receivers are authorized to participate in the exchange. The IM-PEP enforces 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 cryptographic key, which the IM-PEP applies to each message as it 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 protected with their unique cryptographic key.

Before access to a chat room or message is granted, users must have the policy right to access it, given the chat room’s sensitivity markings. Once in the chat room, all messages are delivered to the recipient, but specially marked messages are delivered only if the recipient has the policy right to see data with the associated security attributes.

For each exchange between the chat client and server, the Instant Messaging PEP performs the following:

1. Intercept the message;
2. Extract the metadata on the message;
3. Retrieve the user attributes from the user-specified LDAP or ICAM system;
4. Package and issue an adjudication request to the PDP or ABAC system;
5. Enforce the PDP or ICAM results (Permit, Deny, or indeterminate) and
6. Log the transaction to the TLS.

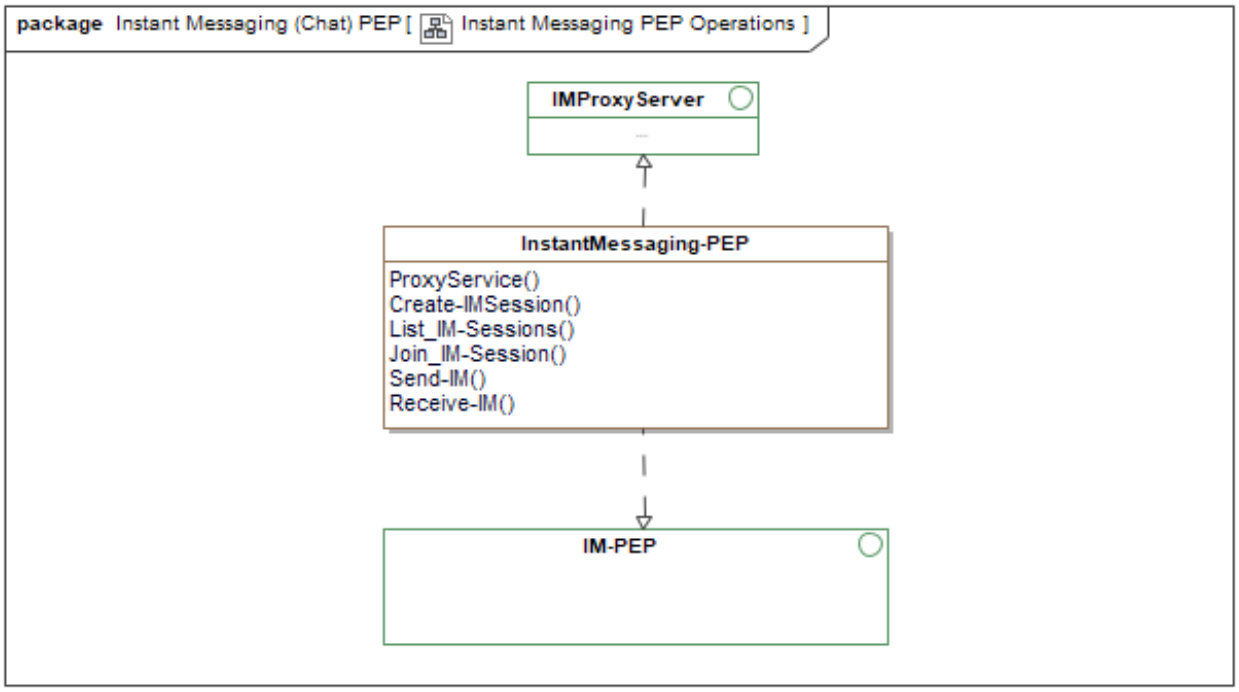


Figure 65 -Instant Messaging PEP Operations

The following table identifies and describes the elements and operations illustrated in the previous figure - Instant Messaging PEP Operations.

Table 31 - Instant Messaging PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
InstantMessaging-PEP	The IM or chat PEP operates between user applications and the IM server. The PEP performs the following: 1. Intercept each message; 2. Authenticate the sender and recipients; 3. Extract the metadata from the message of request; 4. Compose and issue an adjudication request to the PDP or ABAC system; 5. Enforce the PDP receipt and release determination for each message or request and 6. Log the transaction to the TLS. The InstantMessaging-PEP specializes in protecting chat or IM Messages. Clause 10.3 provides the details for this interface.
	Element Type: Class Owned Operations: ProxyService: Proxy services that intercept instant/text messages to and from a protected IM service. 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 performs the following: • Intercept the user request to create or establish a chat room; • Parse and extract the pertinent information from the request message; • Requests the User's attributes from the LDAP or ICAM system; • 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; ○ Request the chat room from the IM Server; and ○ Send notifications to the requested participants and

`

Table 31 - Instant Messaging PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
	○ Notify the user that the chat room has been created and • Package and send 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. As part of this process, the IM PEP performs the following: • Intercept the user request to list available community/public chat rooms; • Parse and extract the pertinent information from the request message; • Request the list from the IM Server; • Request the User's attributes from the LDAP or ICAM system; • Request release authorizations from the PDP or ABAC system; • Redact the names and identifiers of the chat rooms that the user is not authorized; • Package and send the redacted list to the user and • Package and send 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 performs the following: • 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 User's attributes from the LDAP or ICAM system; • Request Authorization determination from the PDP that the user is authorized to join the chat room; • If authorized, add the user to the distribution list for the chat room and • Packages and sends a transaction report to the TLS. Send-IM:

Table 31 - Instant Messaging PEP Operations Elements	
Element Name	**Element and Operation Descriptions**

	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 performs the following:
	• Intercept the message;
	• Parse and extract the pertinent information from the sent message;
	• Request the User's attributes from the LDAP or ICAM system;
	• Request authorizations from the PEP that the user is authorized to send information of that sensitivity level to the specified chat room;
	• If authorized:
	  ○ Request the cryptographic key for the chat room;
	  ○ Send Encryption Request (Package SAC) with the cryptographic key and information element to the CTS;
	  ○ Receive and process the CTS-Response message;
	  ○ Package and send the IM Message (SAC) to the IM server and
	• Package and send a transaction report to the TLS.
	
	Receive-IM:
	The IM-PEP provides features to intercept an instant 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 performs the following:
	• Intercept the message;
	• Parse and extract the pertinent information from the received message;
	• Requests the User's attributes from the LDAP or ICAM system;
	• If packaged as an SAC, processes the SAC to extract sensitivity markings of a special message and the information element Token;
	• Request 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 the SAC;

`

Table 31 - Instant Messaging PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
	○ Send decryption Request with the information element and cryptographic key to the CTS; ○ Receive and process the CTS response; ○ Package and send the IM message (Decrypted) to the IM-Client and • Package and send a transaction report to the TLS. Inherited Operations: The above functions utilize the following inherited operations to deliver their features. Inherited functions include: Manage-PEPOperations inherited from PEP Execute-AdministrationFunctions inherited from PEP Gather-UserAttributeData inherited from PEP Gather-RecipientLocation inherited from PEP Gather-IdentityData inherited from PEP Generate-DecisionRequest inherited from PEP Enforce-AuthorizationDecisions inherited from PEP Store-PEPConfiguration inherited from PEP Restore-PEPConfiguration inherited from PEP Publish-PEPConfiguration inherited from PEP Load-PEPconfiguration inherited from PEP Generate_ProcessingUUID inherited from PEP inherited from IEF_Component 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

Table 31 - Instant Messaging PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
	Archive-Properties inherited from IEF_Component

10.2.3.2 Instant Messaging PEP Configurations

As illustrated in the following figure, the PEP forms a proxy between the IM Client and Server. It intercepts each instant message and channels it through the IEF services, which apply data protection to the message's content.

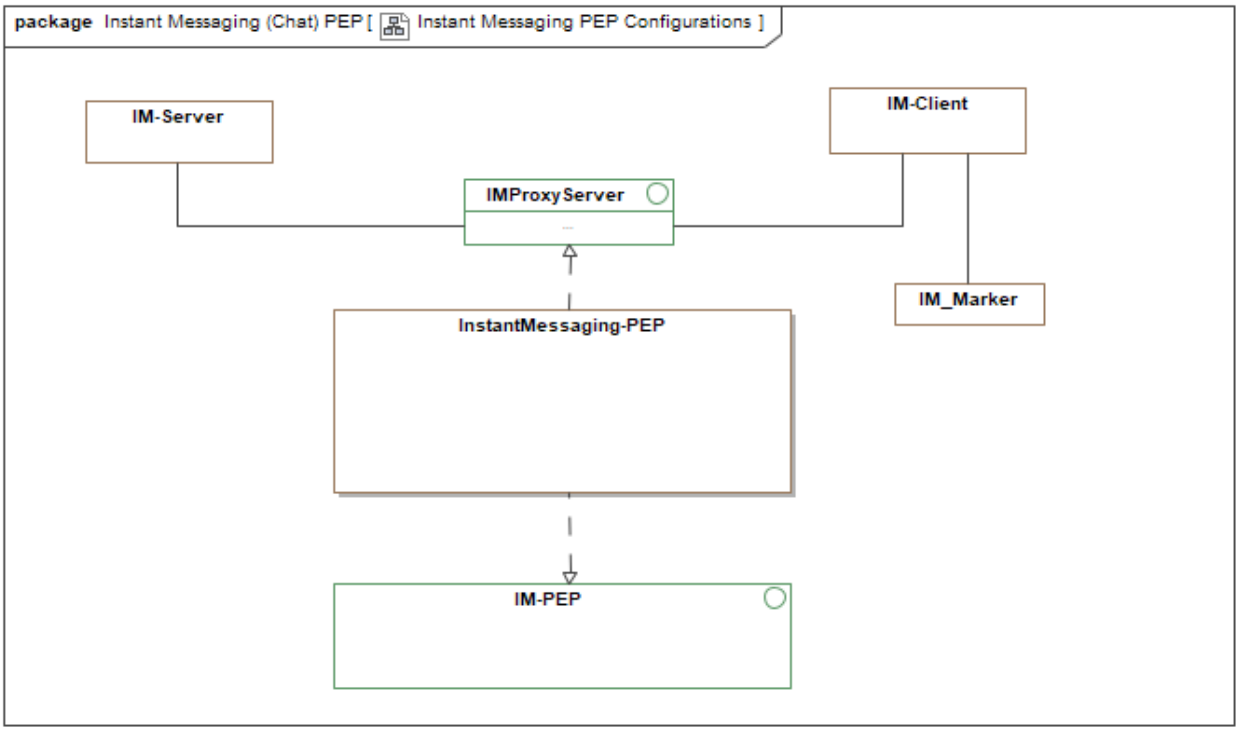


Figure 66 -Instant Messaging PEP Configurations

The following table describes the elements illustrated in the previous figure - Instant Messaging PEP Configurations.

`

Table 32 - Instant Messaging PEP Configurations Elements	
Element Name	**Element Descriptions**
IM-Client Type: Class	A user-specified off-the-shelf application that connects users to online chat rooms offers near real-time text transmission over the Internet or Intranet.
IM-Server Type: Class	A user-specified 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.
IM_Marker Type: Class	The IM-Marker is a software application (or Plugin) that integrates with off-the-shelf IM clients and obliges and prompts the authorized user to mark text messages appropriately. The IEF requires that text messages and emails be appropriately marked (Tagged or labeled) in order to apply access and release policies.
InstantMessaging-PEP Type: Class	The IM or chat PEP operates between user applications and the IM server. The PEP performs the following: 1. Intercept each message; 2. Authenticate the sender and recipients; 3. Extract the metadata from the message of request; 4. Compose and issue an adjudication request to the PDP or ABAC system; 5. Enforce the PDP receipt and release determination for each message or request and 6. Log the transaction to the TLS. The InstantMessaging-PEP specializes in protecting chat or IM Messages. Clause 10.3 provides the details for this interface.

10.2.4 Structured Messaging PEP

The structured messaging (SM) PEP acts as a proxy to a user-specified data server. The SM-PEP intercepts each message (data request, data, and administration) transiting to and from the server and directs it to the requisite IEF services. The SM PEP provides the following user operations:

1. Authenticate the sender or recipient of the message;
2. Extract the message metadata;
3. Gather user attributes from the use-specified LDAP or ICAM system;
4. Package and issue an adjudication request to the user-specified PDP or ABAC system;
5. Enforce the PDP or ABAC Response:

1. If permitted, release the received message to the PPS for processing or release the message to the user-specified middleware;
2. If denied, block the receipt or release of the message and
3. If indeterminant, perform user-specified actions and

6. Log the transaction to the TLS.

In most configurations, the SM PEP acts as the API between the middleware and the packaging and Processing services (PPS) that provide Data-Centric Security (DCS) or data sharing and safeguarding (DSS) services for the data store. See Clause 11 for more information on the PPS.

10.2.4.1 Structured Messaging PEP Operations

The following figure depicts a structured messaging PEP configuration in which the user's information services are situated outside the IEF environment. It identifies the capabilities comprising the structured messaging PEP, which do not functionally change between configurations beyond the addition of the proxies.

This reference architecture offers two configurations for the PEP to access security services:

1. The PEP can use the SMB to access the CTS, SSG, and TLS to provide access to security services (e.g., ICAM, access control [e.g., PDP], Cryptographic, key management and logging; or
2. As illustrated in this clause, the PEP can provide dedicated interfaces to each necessary service., including:

- Identity, Credential, and Access Management (ICAM) interface;
- Logging Interface;
- Access Control Interface;
- Cryptographic Services Interface; and
- Key Management Services Interface.

The optional interfaces enable the PEP to interact directly with the user's security infrastructure. Clause 10.4 provides details on this PEP specialization.

`

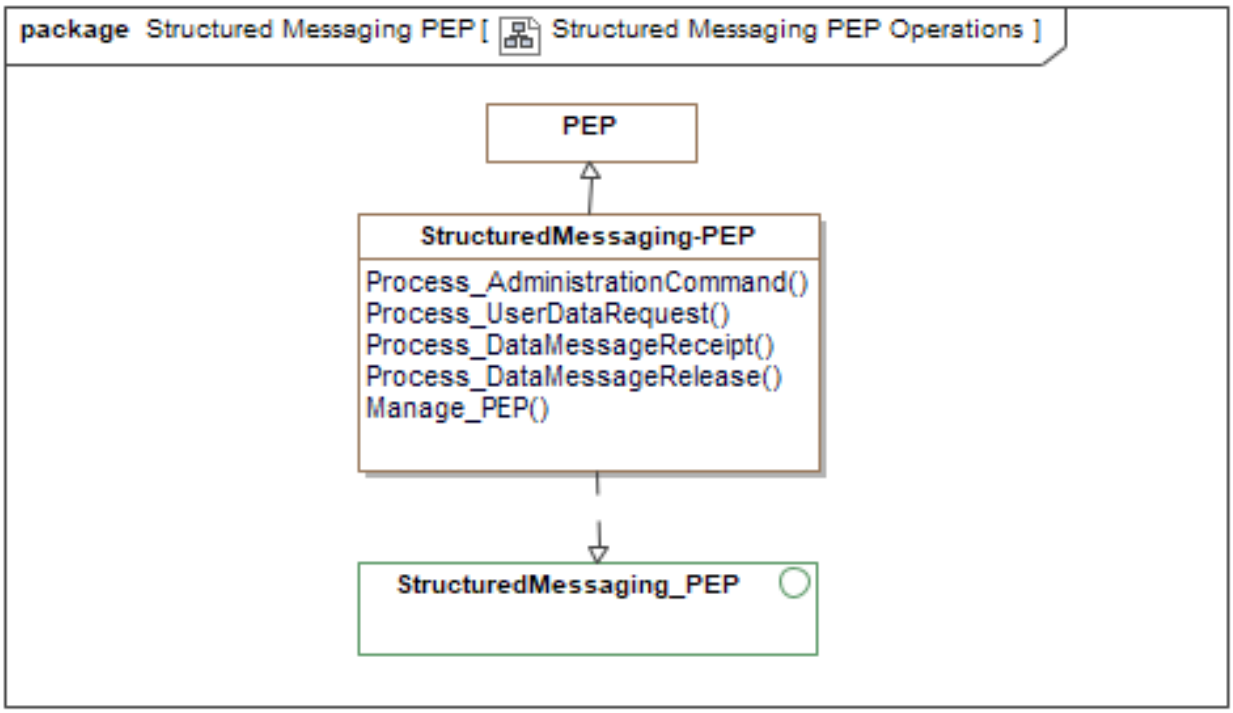


Figure 67 -Structured Messaging PEP Operations

The following table identifies and describes the elements and operations illustrated in the previous figure - Structured Messaging PEP Operations.

Table 33 - Structured Messaging PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
PEP	The Policy Enforcement Point (PEP) intercepts all user and system requests to access or release data elements protected by IEF services. The PEP gathers information about the data elements, sender, recipients, and communication channel, packages an adjudication request to the PDP, and enforces the PDP access or release determination. In a zero-trust environment, the PEP interface authenticates and authorizes each transaction with an external cryptographic service. The PEP must provide one or both of the feature sets defined by the SMB or Direct Connect options.
	Element Type: Class Owned Operations: Manage-PEPOperations: The PEP must provide features that manage the execution of PEP functions in each potential configuration (e.g., Direct

Table 33 - Structured Messaging PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
	connect, SMB connection, or a combination of direct and SMB configurations). Execute-AdministrationFunctions: The PEP must provide features that execute AdministrativeCommands from an authorized Policy Administration Point. PEP administrative functions include: 1. Activate PEP features; 2. Deactivate PEP features; 3. Configure PEP Parameters; 4. Archive PEP Operational Environment; 5. Publish PEP Configuration; 6. Store PEP Configuration; and 7. Retrieve PEP Configuration; Gather-UserAttributeData: The PEP must provide features that gather the sender and receiver attributes to determine their authorization to receive data elements. Gather-RecipientLocation: (Optional) The PEP provides features that gather information about the recipients' location (physical or electronic). A recipient's location(s) (e.g., network location, physical location, and device) may impact the user's attributes and the information content they are authorized to receive. These features may only be available if the IEF can request the information from the user's situational awareness, incident management, or network management systems. Gather-IdentityData: The PEP must provide features that gather the identity information for the sender and recipients of the specified information element (s). These features allow users to request this information from the infrastructure services that provide identity management. All requests to the users' specified infrastructure are issued through the Security Services Gateway using an SSG-Request message. Generate-DecisionRequest: The PEP must provide features that generate a decision request to the PDP. Enforce-AuthorizationDecisions: The PEP Provides features that enforce PDP authorization decisions for each data receipt and release. Store-PEPConfiguration:

`

Table 33 - Structured Messaging PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
	The PEP Provides features that gather and store its configuration parameters in local storage. Restore-PEPConfiguration: The PEP Provides features that retrieve and load its configuration parameters from local storage. Publish-PEPConfiguration: The PEP must provide features that gather and publish its configuration parameters to the PAP. Load-PEPconfiguration: The PEP must provide features that receive and load its configuration parameters from the PAP. Generate_ProcessingUUID: Upon receiving a message, the PEP must provide features that generate a universally unique identifier (UUID). The structure of the UUID must identify the node receiving the data. If the received message includes a UUID, the PEP must adopt the received UUID. Inherited Operations: The above functions utilize the following inherited operations to deliver their features. Inherited functions include: inherited from IEF_Component 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
StructuredMessaging-PEP	As a specialization of the PEP, the StructuredMessaging-PEP provides external interfaces between: 1. System or user application and a protected data store or

Table 33 - Structured Messaging PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
	2. systems and applications. The StructuredMessaging-PEP acts as a proxy service that authenticates each user and authorizes the receipt or release of each data message. Clause 10.4 provides details on this interface.
	Element Type: Class Owned Operations: Process_AdministrationCommand: The StructuredMessaging-PEP must provide features that orchestrate the authorization of rejection of an administration command to the PEP or other IEF component in the environment. An administration command must include: • User's Identity or token; • Reference to PAP being used and • Release instructions (e.g., communication channel and messaging protocol) for the administration response. On receipt of an administrative command from a PAP, the PEP performs the following: • Extract and parse the message header and metadata; • Gather the user's Attributes; • Gather the source of the message; • Gather the access confidentiality attributes for the data node; • (optionally) Gathers the location and other SA information. • package and issue an adjudication request to the PDP or ABAC System; • Enforces the PDP or ABAC response: ○ If permitted, package and issue the command(s) to the specified component for processing; ○ If denied, block the request and ○ if indeterminant, enforce user-defined policy and • Log the transaction to the TLS. Process_UserDataRequest: The StructureMessaging-PEP must provide 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 user's

`

Table 33 - Structured Messaging PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
	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). The PEP then performs the following: • Gather the users' attributes; • Gather the access confidentiality attributes for the data node; • (optionally) Gathers the location and other SA information; • package and issue an adjudication request to the PDP or ABAC System; • Enforces the PDP or ABAC response: ○ If permitted, he PEP packages and issues a request to the PPS for processing; ○ If denied, Blocks the request; ○ if indeterminant, enforce user-defined policy and • Log the transaction to the TLS. Process_DataMessageReceipt: The Messaging-PEP must provide 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: • Extract and parse the message header and metadata; • Identify the message type. • Gather the rules governing the structure and content of the message type. • De-construct the message to collect the embedded information elements. • Extract the metadata about each information element, which may include: ○ Message metadata - data elements describing the content of the information element (/Message);

Table 33 - Structured Messaging PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
	○ Data Owner Metadata - data elements identifying the owner/steward of the content of the information element (/Message); ○ Privacy Metadata - data elements describing the private content and release restriction associated with the information element (/Message); ○ Security (Confidentiality) Metadata - data elements describing the classified content and release restriction associated with the information element (/Message) and ○ HandlingInstructions - data elements describing any specialized data/information access, processing, or storage instructions for the information element (/Message); • Request the recipient's attributes from the user-specified AD or ICAM system; • (Optional) Request 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, decrypt the Payload (information element(s)) provided in the message: • Packages and issues the Metadata and decrypted information element to the PPS for processing and • Logs the transaction to the TLS. Process_DataMessageRelease: The Messaging-PEP must provide features that stage the processing of an information element(s) from a PPS for release to the user-specified middleware. The SMB messaging protocol is stripped by the interface prior to the commencement of processing. The Messaging-PEP then performs the following: • Extract 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.

`

Table 33 - Structured Messaging PEP Operations Elements	
Element Name	**Element and Operation Descriptions**

	• Request the recipient(s) attributes from the LDAP or ICAM System;
	• (Optional) Requests the operational context for the exchange from the user's SA or incident management system;
	• Package and issue an authorization request to the PDP;
	• If authorized, encrypt the authorized information element(s):
	  ○ Requests Cryptographic Keys and tokens from the user-specified Key Management Service(s) (SSG-Request);
	  ○ Packages a CTS-Request for the encryption of each authorized information element and
	  ○ Packages and issues the Message to the middleware for dissemination and
	• Logs the transaction to the TLS.
	Manage_PEP:
	The Messaging-PEP must provide 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:
	The above functions utilize the following inherited operations to deliver their features. Inherited functions include:
	Manage-PEPOperations inherited from PEP
	Execute-AdministrationFunctions inherited from PEP
	Gather-UserAttributeData inherited from PEP
	Gather-RecipientLocation inherited from PEP
	Gather-IdentityData inherited from PEP
	Generate-DecisionRequest inherited from PEP
	Enforce-AuthorizationDecisions inherited from PEP
	Store-PEPConfiguration inherited from PEP
	Restore-PEPConfiguration inherited from PEP
	Publish-PEPConfiguration inherited from PEP
	Load-PEPconfiguration inherited from PEP
	Generate_ProcessingUUID inherited from PEP

Table 33 - Structured Messaging PEP Operations Elements	
Element Name	**Element and Operation Descriptions**
	inherited from IEF_Component 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

10.2.4.2 Structured Messaging PEP Operations

The following figure depicts a structured messaging PEP configuration in which the user's information services are situated outside the IEF environment. It identifies the capabilities comprising the structured messaging PEP, which do not functionally change between configurations beyond the addition of the proxies.

This reference architecture offers two configurations for the PEP to access security services:

1. The PEP can use the SMB to access the CTS, SSG, and TLS to provide access to security services (e.g., ICAM, access control [e.g., PDP], Cryptographic, key management and logging; or
2. As illustrated in this clause, the PEP can provide dedicated interfaces to each necessary service., including:

- Identity, Credential, and Access Management (ICAM) interface;
- Logging Interface;
- Access Control Interface;
- Cryptographic Services Interface; and
- Key Management Services Interface.

The optional interfaces enable the PEP to interact directly with the user's security infrastructure. Clause 10.4 provides details on this PEP specialization.

`

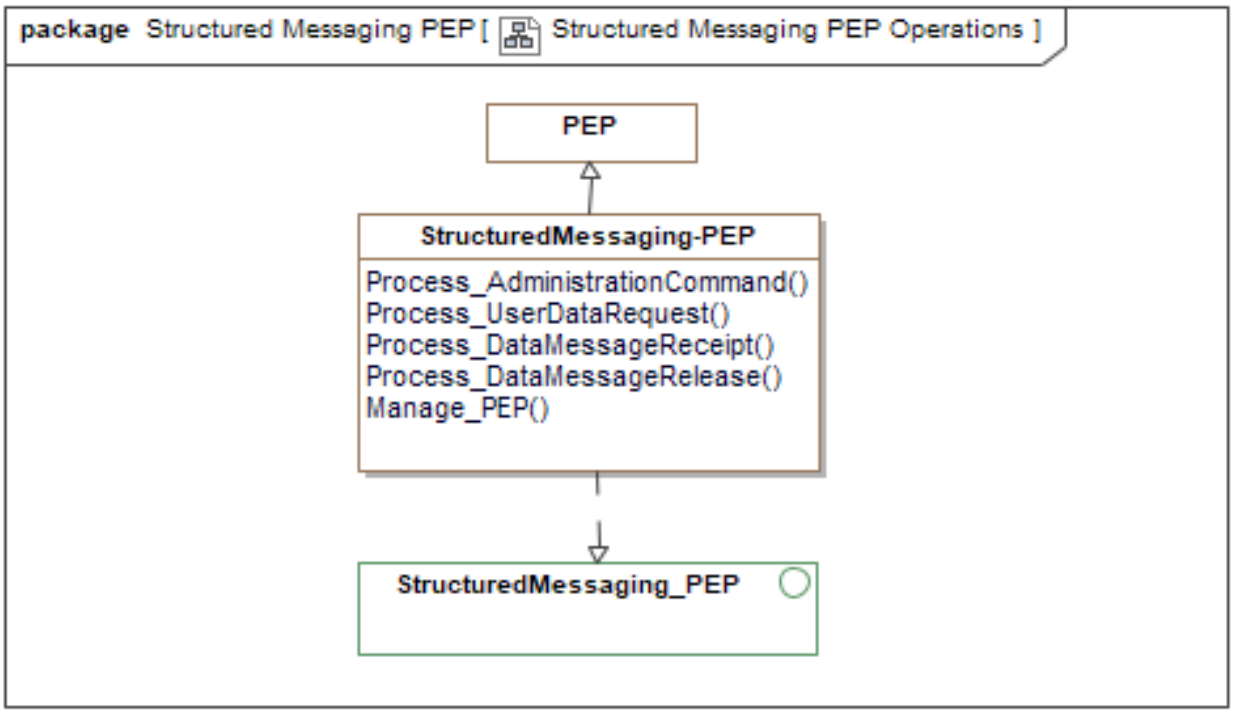


Figure 68 -Structured Messaging PEP Operations

The following table describes the elements illustrated in the previous figure - Structured Messaging PEP Operations.

Table 34 - Structured Messaging PEP Operations Elements	
Element Name	**Element Descriptions**
PEP Type: Class	The Policy Enforcement Point (PEP) intercepts all user and system requests to access or release data elements protected by IEF services. The PEP gathers information about the data elements, sender, recipients, and communication channel, packages an adjudication request to the PDP, and enforces the PDP access or release determination. In a zero-trust environment, the PEP interface authenticates and authorizes each transaction with an external cryptographic service. The PEP must provide one or both of the feature sets defined by the SMB or Direct Connect options.
StructuredMessaging-PEP Type: Class	As a specialization of the PEP, the StructuredMessaging-PEP provides external interfaces between: 1. System or user application and a protected data store or 2. systems and applications.

Table 34 - Structured Messaging PEP Operations Elements	
Element Name	**Element Descriptions**
	The StructuredMessaging-PEP acts as a proxy service that authenticates each user and authorizes the receipt or release of each data message. Clause 10.4 provides details on this interface.

10.2.4.3 Structured Messaging Proxy Configuration

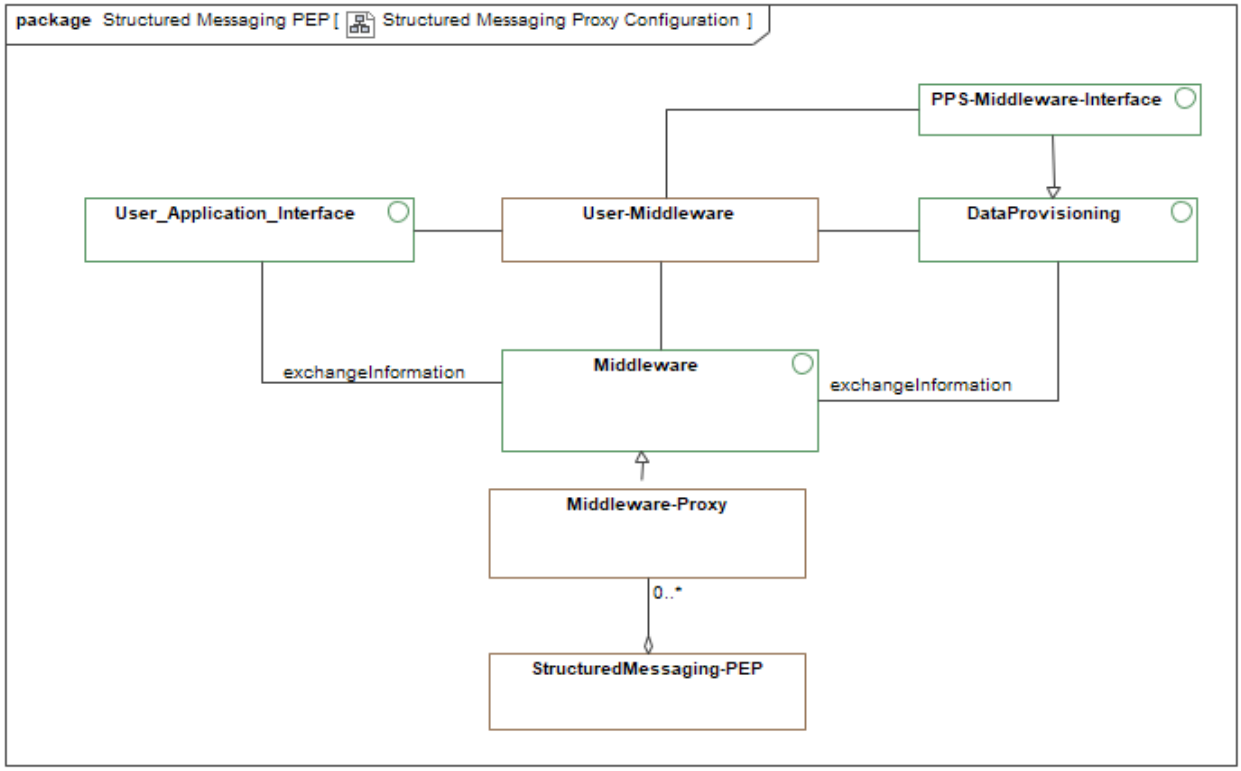


Figure 69 -Structured Messaging Proxy Configuration

The following table describes the elements illustrated in the previous figure - Structured Messaging Proxy Configuration.

Table 35 - Structured Messaging Proxy Configuration Elements	
Element Name	**Element Descriptions**
Middleware-Proxy Type: Class	A proxy service that intercepts messages to and from a protected structured data source.

`

Table 35 - Structured Messaging Proxy Configuration Elements	
Element Name	**Element Descriptions**
StructuredMessaging-PEP Type: Class	As a specialization of the PEP, the StructuredMessaging-PEP provides external interfaces between: 1. System or user application and a protected data store or 2. systems and applications. The StructuredMessaging-PEP acts as a proxy service that authenticates each user and authorizes the receipt or release of each data message. Clause 10.4 provides details on this interface.
User-Middleware Type: Class	A user-specified data exchange system enables software systems, applications, and systems to share information and data elements (e.g., DDS, Enterprise Services Bus, or Web Services).

10.3 PEP General Configurations

The following clauses provide general configurations for IEF PEPs.

10.3.1 SMB Integration to User Infrastructure

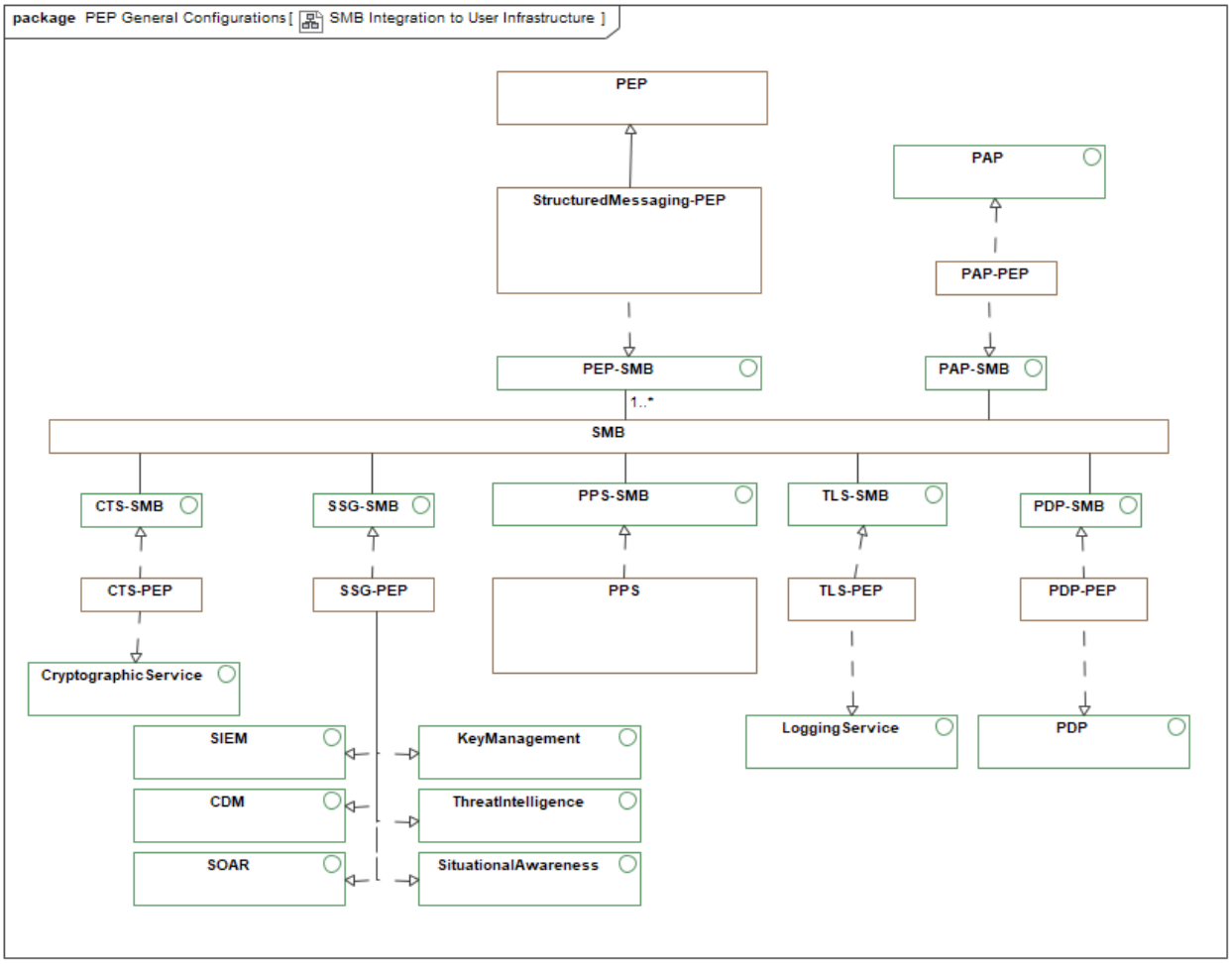


Figure 70 -SMB Integration to User Infrastructure

The following table describes the elements illustrated in the previous figure - SMB Integration to User Infrastructure.

Table 36 - SMB Integration to User Infrastructure Elements	
Element Name	**Element Descriptions**
CTS-PEP Type: Class	This CTS-PEP connects the IEF Services to user-specified cryptographic services through a ZTA-enabled interface. The CTS operates in the same fashion as the SSG, and it is tailored to cryptographic operations.
PAP-PEP Type: Class	The PAP-PEP must provide features that integrate a user-specified administration system (or Policy Administration Point) with the users' security infrastructure. This administration system would provide the features described in Clause 8.
PDP-PEP Type: Class	This PEP (/proxy) connects the IEF Services to access control adjudication services provided by the user or the local infrastructure (e.g., Cloud Service Provider (CSP)). The PDP-PEP employs external services (e.g., RBAC, ABAC, PBAC, or

`

Table 36 - SMB Integration to User Infrastructure Elements	
Element Name	**Element Descriptions**
	UBAC) to adjudicate access to or release of resources to a specified user. (Clause 9)
PEP Type: Class	The Policy Enforcement Point (PEP) intercepts all user and system requests to access or release data elements protected by IEF services. The PEP gathers information about the data elements, sender, recipients, and communication channel, packages an adjudication request to the PDP, and enforces the PDP access or release determination. In a zero-trust environment, the PEP interface authenticates and authorizes each transaction with an external cryptographic service. The PEP must provide one or both of the feature sets defined by the SMB or Direct Connect options.
PPS Type: Class	The Policy-based Packaging and Processing Service (PPS) transitions structured Information Elements (e.g., NIEM, EDXL, and HL7) between data stores and information exchange services following local information sharing, safeguarding, and DCS policies. The IEF PPS enforces policies that conform to the specification of the Information Exchange Packaging Policy Vocabulary (IEPPV). The PPS allows users to selectively package (aggregate, transform, mark, filter, structure, and format) information elements for publication to authorized recipients. It also allows processing (parsing, transforming, mapping, and Marshalling) structured data and integrating the data elements into user-specified data stores.
SMB Type: Class	The Secure Message Bus (SMB) is the user, implementor, or integrator selected data exchange technology (e.g., DDS + Security features) used to exchange operational and administrative data between IEF Components. The SMB is isolated from the exchange technologies that integrate an IEF configuration into the users' environment. In the IEF, data protection is provided by a set of interconnected services that interact through the exchange of messages on two separate and isolated messaging infrastructures: • A security messaging infrastructure that carries policy data, security attributes, and cryptographic information; and • A messaging infrastructure that carries tamper-resistant operational logging information. The IEF Secure Messaging Bus (SMB) is a software layer between the operating system and the IEF components. The SMB isolates the IEF component communications from the user's applications and infrastructure. The inter-component

Table 36 - SMB Integration to User Infrastructure Elements	
Element Name	**Element Descriptions**
	(/service) messaging is isolated from the standard operational communications to mitigate penetration and tampering using defense-in-depth strategies. The IEF internal information exchange protocol conforms to industry-accepted, open standards that are based on XML(e.g., extensible Messaging and Presence Protocol (XMPP)). The messaging services/infrastructure) provisions information (messages) between IEF services /components). Although the specific protocol, format, and content of the messages will depend on the nature of the service being accessed, all messages are delivered through the same communications mechanism. The ability to provide robust, secure, and trusted delivery of security messages between IEF services forms the critical core of the IEF architecture.
SSG-PEP Type: Class	The SSG-PEP is a hybrid between the SSG and a PEP, providing ZT security elements to the interfaces between the IEF and the user's security infrastructure. It connects the IEF Services to Security services provided by the user or the local infrastructure (e.g., Cloud Service Provider (CSP)). It enables the IEF components to interoperate with user-specified security services. (e.g., Identity, credential, access management services, access control services, and essential management services) and infrastructure using a standardized SMB Interface. (Clause 12)
StructuredMessaging-PEP Type: Class	As a specialization of the PEP, the StructuredMessaging-PEP provides external interfaces between: 1. System or user application and a protected data store or 2. systems and applications. The StructuredMessaging-PEP acts as a proxy service that authenticates each user and authorizes the receipt or release of each data message. Clause 10.4 provides details on this interface.
TLS-PEP Type: Class	This PEP (/proxy) connects the IEF Services to logging services provided by the user or the local infrastructure (e.g., Cloud Service Provider (CSP)). The TLS-PEP is the integration point between IEF components and external logging services. (Clause 15)

10.3.2 Direct Integration to User Infrastructure

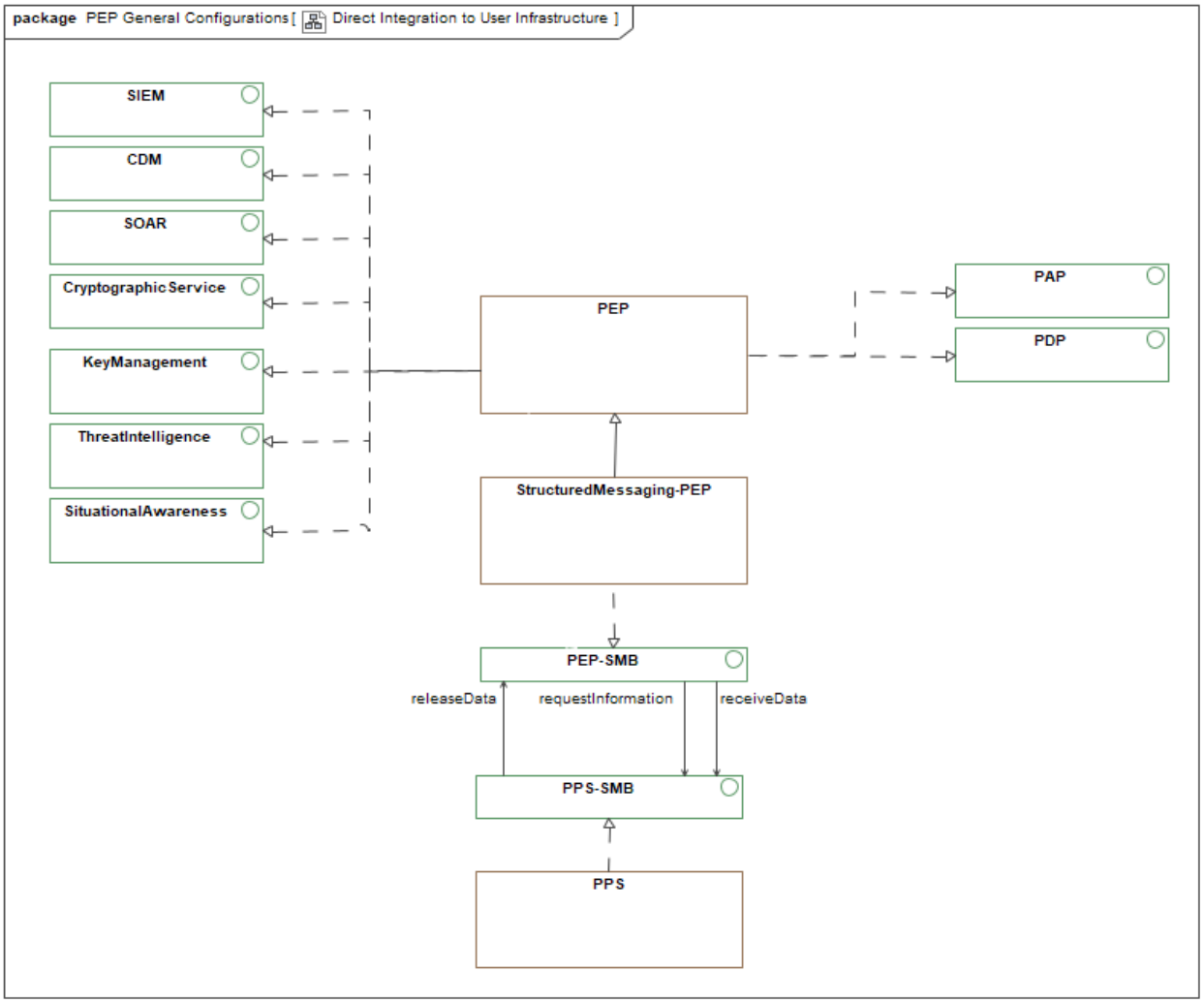


Figure 71 -Direct Integration to User Infrastructure

The following table describes the elements illustrated in the previous figure - Direct Integration to User Infrastructure.

Table 37 - Direct Integration to User Infrastructure Elements	
Element Name	**Element Descriptions**
PEP Type: Class	The Policy Enforcement Point (PEP) intercepts all user and system requests to access or release data elements protected by IEF services. The PEP gathers information about the data elements, sender, recipients, and communication channel, packages an adjudication request to the PDP, and enforces the PDP access or release determination. In a zero-trust environment, the PEP interface authenticates and authorizes each transaction with an external cryptographic service.

Table 37 - Direct Integration to User Infrastructure Elements	
Element Name	**Element Descriptions**
	The PEP must provide one or both of the feature sets defined by the SMB or Direct Connect options.
PPS Type: Class	The Policy-based Packaging and Processing Service (PPS) transitions structured Information Elements (e.g., NIEM, EDXL, and HL7) between data stores and information exchange services following local information sharing, safeguarding, and DCS policies. The IEF PPS enforces policies that conform to the specification of the Information Exchange Packaging Policy Vocabulary (IEPPV). The PPS allows users to selectively package (aggregate, transform, mark, filter, structure, and format) information elements for publication to authorized recipients. It also allows processing (parsing, transforming, mapping, and Marshalling) structured data and integrating the data elements into user-specified data stores.
StructuredMessaging-PEP Type: Class	As a specialization of the PEP, the StructuredMessaging-PEP provides external interfaces between: 1. System or user application and a protected data store or 2. systems and applications. The StructuredMessaging-PEP acts as a proxy service that authenticates each user and authorizes the receipt or release of each data message. Clause 10.4 provides details on this interface.

`

11 Policy-based Packaging and Processing Services (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., STANAGS (e.g., 5527, and 4559), MIM, 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 PPS Components

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

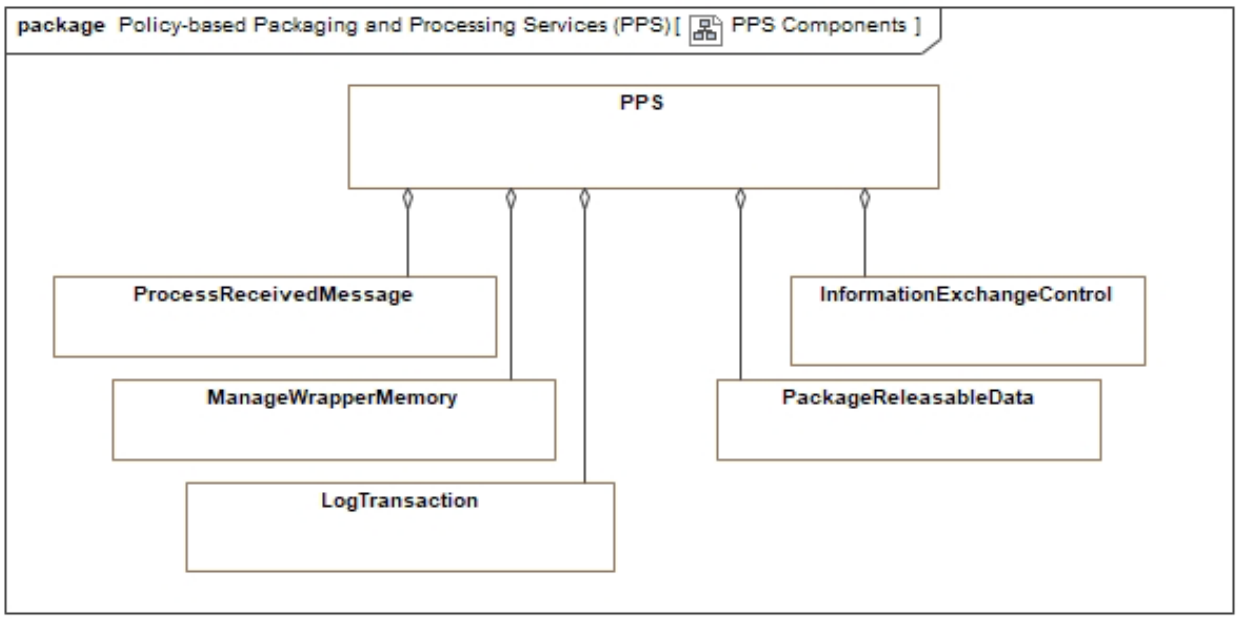


Figure 72 -PPS Components

The following table describes the elements illustrated in the previous figure - PPS Components.

Table 38 - PPS Components Elements	
Element Name	**Element Descriptions**
InformationExchangeControl Type: Class	The Information Exchange Controls enforce the Information Exchange Specifications (IES) policies (or exchange policies) expressed using the IEPPV. The IES expresses policies (rules and constraints) governing the release of information elements (/messages) permitted for release on each IES (e.g., Community of Interest).
LogTransaction Type: Class	The Log Transaction feature compiles fragments of a PPS transaction log history (from message receipt to the release required data element) and prepares the transaction log for release to the TLS.
ManageWrapperMemory Type: Class	Manage Wrapper Memory provides PPS features that persist data elements in memory and to the user data store. The features also maintain the consistency and integrity of PPS memory.
PackageReleasableData Type: Class	Packaging of releasable Data includes aggregating, transforming, labeling (/marking), and redacting data elements per semantic policy. These features ensure that the released data content conforms to the recipients' needs and authorizations. The features transform the data elements from the semantics of the user data store to those of the exchange semantic (/canonical model).
PPS Type: Class	The Policy-based Packaging and Processing Service (PPS) transitions structured Information Elements (e.g., NIEM, EDXL, and HL7) between data stores and information exchange services following local information sharing, safeguarding, and DCS policies. The IEF PPS enforces policies that conform to the specification of the Information Exchange Packaging Policy Vocabulary (IEPPV). The PPS allows users to selectively package (aggregate, transform, mark, filter, structure, and format) information elements for publication to authorized recipients. It also allows processing (parsing, transforming, mapping, and Marshalling) structured data and integrating the data elements into user-specified data stores.
ProcessReceivedMessage Type: Class	Processing received data includes parsing, mapping, transformation, and marshaling data elements and attributes to the user data store. The features transform the data elements from the exchange semantic (canonical exchange model for the received message) to the storage semantic of the data store.

11.1.1 PPS Operations

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

`

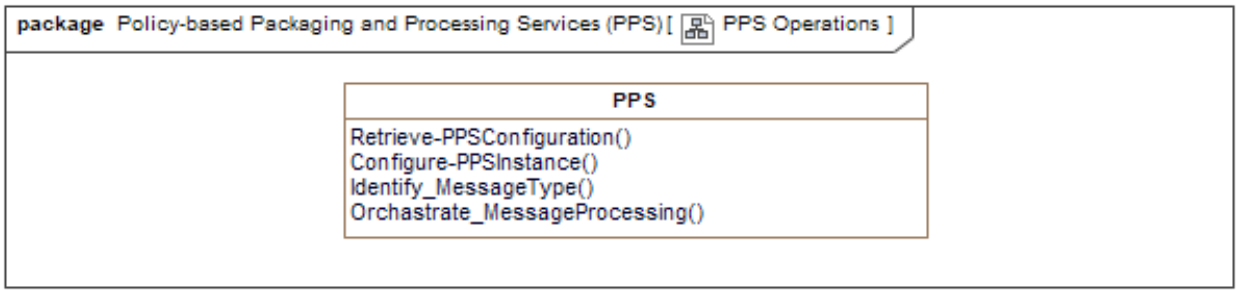


Figure 73 -PPS Operations

The following table identifies and describes the elements and operations illustrated in the previous figure - PPS Operations.

Table 39 - PPS Operations Elements	
Element Name	**Element and Operation Descriptions**
PPS	The Policy-based Packaging and Processing Service (PPS) transitions structured Information Elements (e.g., NIEM, EDXL, and HL7) between data stores and information exchange services following local information sharing, safeguarding, and DCS policies. The IEF PPS enforces policies that conform to the specification of the Information Exchange Packaging Policy Vocabulary (IEPPV). The PPS allows users to selectively package (aggregate, transform, mark, filter, structure, and format) information elements for publication to authorized recipients. It also allows processing (parsing, transforming, mapping, and Marshalling) structured data and integrating the data elements into user-specified data stores.
	Element Type: Class Owned Operations: Retrieve-PPSConfiguration: The PPS must provide features that retrieve and process a PPS configuration file. Configure-PPSInstance: The PPS must provide features that configure the PPS per the content of the configuration file. Identify_MessageType: The PPS must provide features that decompose a received message and identify its type, which may include: • A Data Message containing data from a sensor, system, or application; • A Data Request Message containing the parameters of a data request and the attributes of the requester or

Table 39 - PPS Operations Elements	
Element Name	**Element and Operation Descriptions**
	• An Administration Command Message from a PAP (see PAP Command Message). Orchastrate_MessageProcessing: The PPS must provide features that stage or orchestrate its activities to meet the processing requirements for the message type received. Inherited Operations: The above functions utilize the following inherited operations to deliver their features. Inherited functions include: inherited from IEF_Component 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

11.1.2 ProcessReceivedMessage

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

`

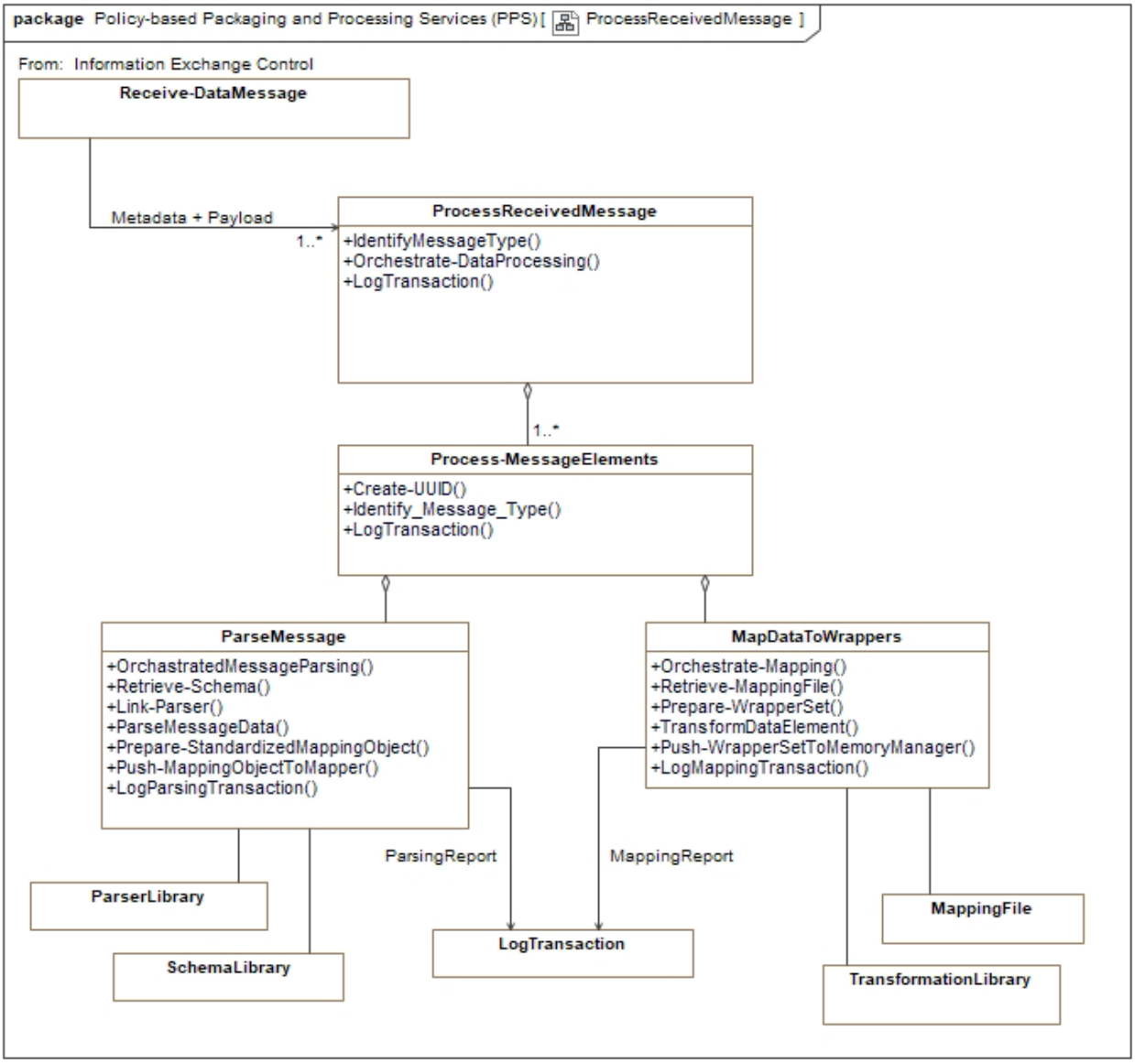


Figure 74 -ProcessReceivedMessage

The following table identifies and describes the elements and operations illustrated in the previous figure - ProcessReceivedMessage.

Table 40 - ProcessReceivedMessage Elements	
Element Name	**Element and Operation Descriptions**
LogTransaction	The Log Transaction feature compiles fragments of a PPS transaction log history (from message receipt to the release required data element) and prepares the transaction log for release to the TLS.

Table 40 - ProcessReceivedMessage Elements	
Element Name	**Element and Operation Descriptions**
	Element Type: Class Owned Operations: Receive-LogElement: The PPS must provide features that receive a transaction log fragment from an IEF component. Integrate-LogElement: The PPS must provide features that integrate a log fragment into the transaction log structure, which tracks PEP and PPS activity from the receipt of a data message until the transaction is concluded through the execution of all watchpoint data releases. Format-LogReport: The PPS must provide features that transform the transactional log object into the log report format for release to the TLS and user-specified logging system. Issue-LogReport: The PPS must provide features that push the log report to the TLS for release.
MapDataToWrappers	The Map Data to Wrapper(s) features generate wrapper definitions for data and metadata elements and attributes in the received message. They map, transform, and marshall elements and attributes for inclusion in PPS memory and routing to the user's data store.
	Element Type: Class Owned Operations: Orchestrate-Mapping: The PPS must provide features that orchestrate the wrapper mapping and generation process. Retrieve-MappingFile: The PPS must provide features that map received data and metadata elements to Wrapper Elements defined in the semantic policy using a user-controlled mapping file. Prepare-WrapperSet: The PPS must provide features that prepare the set of trappers generated from the received message to the form required by the memory manager. TransformDataElement: The PPS must provide features that transform data to conform to storage model semantics. Push-WrapperSetToMemoryManager:

`

Table 40 - ProcessReceivedMessage Elements	
Element Name	**Element and Operation Descriptions**
	The PPS must provide features that push the generated wrappers to the memory manager to be generated in memory and persisted to the user's data store. LogMappingTransaction: The PPS must provide features that prepare and push mapping log elements for integration into the transactional log report.
MappingFile	The mapping file provides the rules and constraints for aligning received data elements and attributes to the user-specified data storage structure.
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
ParseMessage	The PPS provides features that decompose the information element (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 in the information.
	Element Type: Class Owned Operations: OrchastratedMessageParsing: The PPS must provide features that orchestrate the parsing of the data and metadata in the received message. Retrieve-Schema: The PPS must provide features that retrieve the schema file for the specific message being processed. This file is retrieved from the IEF schema library. Link-Parser: The PPS must provide features that engage the parser operations needed to parse the message type. The parser operations are provided in a dynamically linked library for the specific implementation language. (e.g., ParserLibrary.Jar) ParseMessageData: The PPS must provide features that decompose the message data and metadata elements (Payload) attributes and interrelationships, populate required to map the message

Table 40 - ProcessReceivedMessage Elements	
Element Name	**Element and Operation Descriptions**
	content to the underlying data store element (/Wrapper Element) * Semantic (data) patterns are defined by Semantic, Transactional and Wrapper Elements in PPS policy conforming to the IEPPV. Prepare-StandardizedMappingObject: Take the extracted data and metadata and integrate the elements into a standard mapping object to be processed. Push-MappingObjectToMapper: Push the mapping object to the mapper process to map and transform the objects to Wrapper definitions in the semantic policy. LogParsingTransaction: The PPS must provide features that prepare and push parsing. Log elements for integration into the transactional log report
ParserLibrary	The set of methods used to parse the message syntaxes (e.g., XML, JSON, BSON, or user-specified) used in the domain. The library represents a dynamically linked library or methods or operations in the implementation language. (e.g., ParserLIB.JAR)
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
Process-MessageElements	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: Create-UUID: Generate a Universally Unique Identifier (UUID) for the incoming message. If the message has a UUID embedded in the header, this UUID should be used to maintain the provenance of the content. Identify_Message_Type: The PPS provides features that identify the type of information message being received from the PEP. LogTransaction: The PPS must provide features that prepare and push processing activity log elements for integration into the transactional log report

`

Table 40 - ProcessReceivedMessage Elements	
Element Name	**Element and Operation Descriptions**
ProcessReceivedMessage	Processing received data includes parsing, mapping, transformation, and marshaling data elements and attributes to the user data store. The features transform the data elements from the exchange semantic (canonical exchange model for the received message) to the storage semantic of the data store.
	Element Type: Class Owned Operations: IdentifyMessageType: The PPS must provide features that identify the type of message received from the PEP. Orchestrate-DataProcessing: The PPS must provide features that orchestrate the processing of all elements (e.g., metadata (/manifest) and payloads in the message. LogTransaction: The PPS must provide features that prepare and push processing activity log elements for integration into the transactional log report.
Receive-DataMessage	When the PPS receives an administrative message from an authorized PAP, it stages the processing of the embedded administrative commands.
	Element Type: Class Owned Operations: Separate-DataComponents: The PPS must provide features that separate a message into its parts for processing. Orchestrate-DataMessageProcessing: The PPS must provide features that stage (orchestrate) the parts for processing (e.g., parsing, mapping, transformation, and marshaling). ReleaseMessagePartsForProcessing: The PPS must provide features that release parts of the message to the parsing and mapping components for processing. LogTransaction: The PPS must provide features that prepare and push processing activity log elements for integration into the transactional log report.
SchemaLibrary	The set of canonical data schemas used to exchange data within the domain of interest.

Table 40 - ProcessReceivedMessage Elements	
Element Name	**Element and Operation Descriptions**
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
TransformationLibrary	The set of methods used to transform or filter data during processing and packaging data elements. The library represents a dynamically linked library of methods or operations in the implementation language. (e.g., TransformationLIB.JAR) This transformation library may also be used to dynamically link to formatting and binding services needed to finalize messages for release.
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.

11.1.3 ManageWrapperMemory

The PPS provides features for managing wrapper objects in memory, limiting the amount of memory used, and maintaining the integrity of the managed data sets.

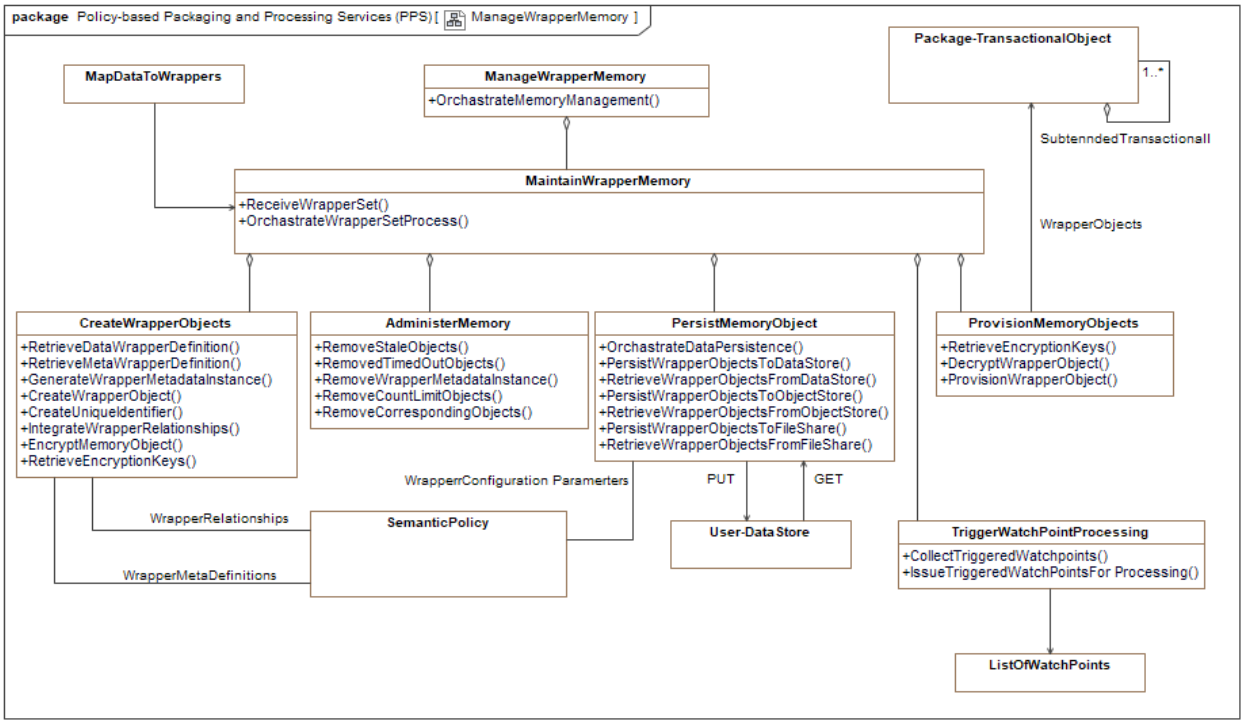


Figure 75 -ManageWrapperMemory

`

The following table identifies and describes the elements and operations illustrated in the previous figure - ManageWrapperMemory.

Table 41 - ManageWrapperMemory Elements	
Element Name	**Element and Operation Descriptions**
AdministerMemory	The administering memory features automate the modification and removal of wrapper objects in the PPS memory per semantic policy, specifically the meta-definition and configuration parameters assigned to the wrapper object. These features seek to control the amount of active memory used by the PPS.
	Element Type: Class Owned Operations: RemoveStaleObjects: The PPS must provide features that remove stale wrapper objects from memory per semantic policy and wrapper configuration. RemovedTimedOutObjects: The PPS must provide features that remove timed-out wrapper objects from memory. RemoveWrapperMetadataInstance: The PPS must provide features that remove wrapper metadata from memory when the wrapper object is removed. RemoveCountLimitObjects: The PPS must provide features that remove wrapper objects that exceed the memory count permitted by the wrapper element configuration parameter. RemoveCorrespondingObjects: The PPS must provide features that remove wrapper objects that correspond to those removed by other administrative functions.
CreateWrapperObjects	The create wrapper object features stage the creation of memory-based data objects conforming to the semantic policy and the data extracted from the received message.
	Element Type: Class Owned Operations: RetrieveDataWrapperDefinition: The PPS must provide features that retrieve a wrapper definition from the set of definitions issued by the parsing and mapping services. RetrieveMetaWrapperDefinition:

Table 41 - ManageWrapperMemory Elements	
Element Name	**Element and Operation Descriptions**
	The PPS must provide features that retrieve the definition of the wrapper element from the semantic policies. GenerateWrapperMetadataInstance: The PPS must provide features that instantiate a meta-wrapper definition instance in memory and link it to the wrapper object. The meta-wrapper definition maintains the specific configuration of the wrapper object. CreateWrapperObject: The PPS must provide features that instantiate a memory-based data object data object based on its definition in the semantic policies. The PPS then populates the newly created object with the data included in the data wrapper definition. CreateUniqueIdentifier: The PPS must provide features that generate a universally unique identifier (UUID) and integrate it into the wrapper object. IntegrateWrapperRelationships: The PPS must provide features that align related wrapper objects per semantic policy and the data retrieved in the message. EncryptMemoryObject: The PPS may provide features that encrypt wrapper objects in memory. If specified in the wrapper meta-wrapper definition, wrapper objects are encrypted. RetrieveEncryptionKeys: The PPS must provide features that generate or retrieve the encryption keys (e.g., symmetric or PKI).
ListOfWatchPoints	A list of semantic elements corresponding to active information exchange specifications that are triggered by data received in a message.
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
MaintainWrapperMemory	The PPS maintains an object store in memory. This enables it to operate with or without a persistent data store (e.g., RDBMS, Object Store, or file share).
	Element Type: Class Owned Operations: ReceiveWrapperSet:

`

Table 41 - ManageWrapperMemory Elements	
Element Name	**Element and Operation Descriptions**
	The PPS provides features that retrieve sets of wrapper definitions extracted from a received message. OrchastrateWrapperSetProcess: The PPS must provide features that stage the processing of all wrapper definitions retrieved from parsed and mapped messages.
ManageWrapperMemory	Manage Wrapper Memory provides PPS features that persist data elements in memory and to the user data store. The features also maintain the consistency and integrity of PPS memory.
	Element Type: Class Owned Operations: OrchastrateMemoryManagement: The PPS must provide features that maintain the consistency and integrity of wrapper memory.
MapDataToWrappers	The Map Data to Wrapper(s) features generate wrapper definitions for data and metadata elements and attributes in the received message. They map, transform, and marshall elements and attributes for inclusion in PPS memory and routing to the user's data store.
	Element Type: Class Owned Operations: Orchestrate-Mapping: The PPS must provide features that orchestrate the wrapper mapping and generation process. Retrieve-MappingFile: The PPS must provide features that map received data and metadata elements to Wrapper Elements defined in the semantic policy using a user-controlled mapping file. Prepare-WrapperSet: The PPS must provide features that prepare the set of trappers generated from the received message to the form required by the memory manager. TransformDataElement: The PPS must provide features that transform data to conform to storage model semantics. Push-WrapperSetToMemoryManager: The PPS must provide features that push the generated wrappers to the memory manager to be generated in memory and persisted to the user’s data store. LogMappingTransaction:

Table 41 - ManageWrapperMemory Elements	
Element Name	**Element and Operation Descriptions**
	The PPS must provide features that prepare and push mapping log elements for integration into the transactional log report.
Package-TransactionalObject	The PPS provides features that aggregate, transform, label (/mark), and redact data objects.
	Element Type: Class Owned Operations: Package-TransactionalObject: The PPS must provide features that prepare the aggregated, transformed, and labeled objects into a self-contained package for inclusion within another transactional or semantic element. Orchestrate-SubtendedTransactionalProcessing: The PPS must provide features that stage the aggregation, transformation, labeling, and redaction of subtended transactional elements. Retrieve-SubtendedObjects: The PPS must provide features that retrieve subtended objects from the wrapper or semantic memory for inclusion within the transactional pattern. Enforce-ContextFilters: The PPS must provide features that filter transactional objects based on their data or metadata content. These filters redact data objects that do not meet specific user requirements. Enforce-ConfigurationParameters: The PPS must provide features that apply or enforce specific configuration parameters found in the Transactional meta-definition or the IES specification. Executed-DataTransformations: The PPS must provide features that link the transactional packaging to specific data transformations of filters that tailor the data to the exchange semantics.
PersistMemoryObject	The PPS provides the ability to persist data to persistent stores (e.g., RDBMS, ObjectStores, or file shares) per semantic and change (/sharing) policy.
	Element Type: Class Owned Operations: OrchastrateDataPersistence: The PPS must provide policies that stage the storage of data objects to a persistent data store per policy.

`

Table 41 - ManageWrapperMemory Elements	
Element Name	**Element and Operation Descriptions**
	PersistWrapperObjectsToDataStore: The PPS must provide features that persist (/store) a data object in an RDBMS table if specified in semantic or exchange (/sharing) policy. RetrieveWrapperObjectsFromDataStore: The PPS must provide features that retrieve wrapper objects from an RDBMS. PersistWrapperObjectsToObjectStore: The PPS must provide features that persist (/store) a data object in an object store if specified in semantic or exchange (/sharing) policy. RetrieveWrapperObjectsFromObjectStore: The PPS must provide features that retrieve wrapper objects from an Object store. PersistWrapperObjectsToFileShare: The PPS must provide features that persist (/store) a data object in a file share if specified in semantic or exchange (/sharing) policy. RetrieveWrapperObjectsFromFileShare: The PPS must provide features that retrieve wrapper objects from a file share.
ProvisionMemoryObjects	The PPS provides features that extract wrapper objects from memory and provide them to the packaging services. Wrapper objects are provided based on UUIDs within the object and the semantic policies.
	Element Type: Class Owned Operations: RetrieveEncryptionKeys: The PPS must provide features that generate or retrieve the encryption keys (e.g., symmetric or PKI). DecryptWrapperObject: The PPS may provide features that decrypt wrapper objects from memory. ProvisionWrapperObject: The PPS must provide features that provision wrapper objects per UUIDs and Semantic policy.
SemanticPolicy	The Semantic Policy represents a serialized version of the IEPPV policy model for the specific domain's data exchange and storage models. The serialized model is ingested at runtime and used to

Table 41 - ManageWrapperMemory Elements	
Element Name	**Element and Operation Descriptions**
	inform PPS processing, memory management, and packaging operations.
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
TriggerWatchPointProcessing	The PPS collects watchpoints triggered during the wrapper creation process.
	Element Type: Class Owned Operations: CollectTriggeredWatchpoints: The PPS must provide features that collect the watchpoints triggered during the creation of wrapper objects found in a received message. IssueTriggeredWatchPointsFor Processing: The PPS must provide features that issue the collected set of watchpoints to the information exchange control for processing.
User-DataStore	The PPS operates with a user-specified data store (e.g., RDBMS, Object Store, or file share) to persist data objects.
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.

11.1.4 PackageReleasableData

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

`

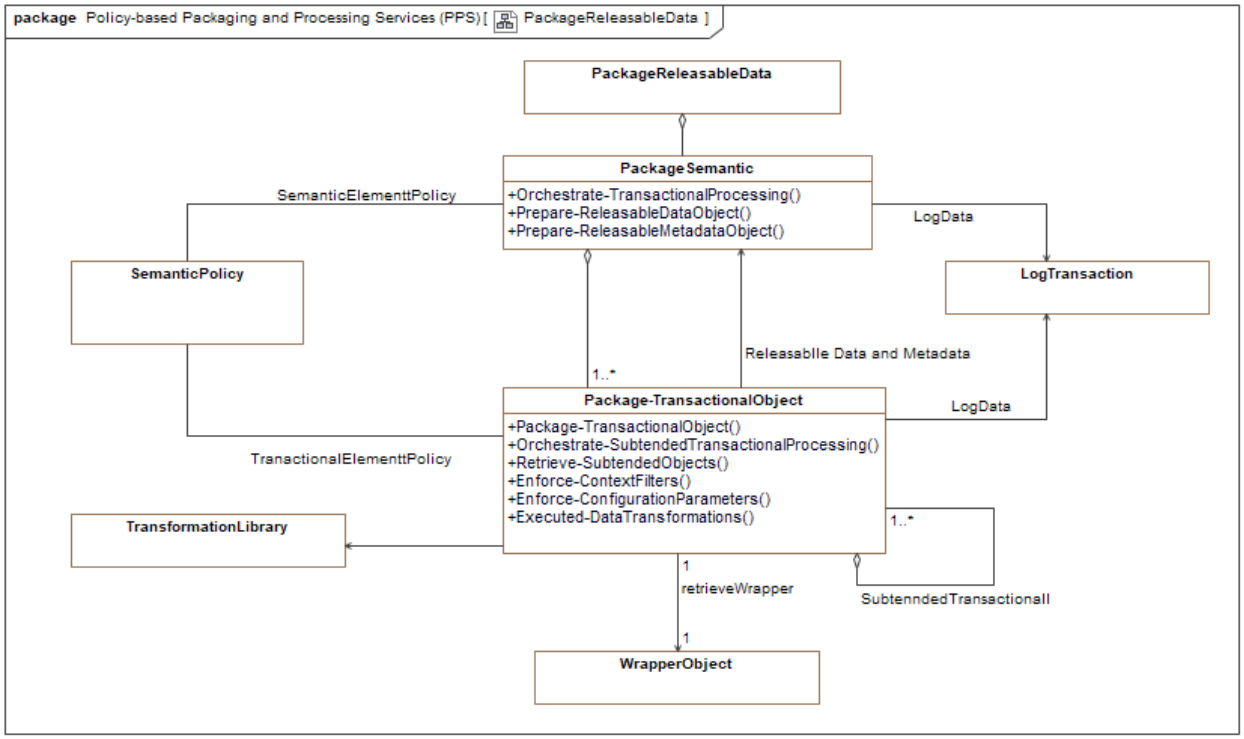


Figure 76 -PackageReleasableData

The following table identifies and describes the elements and operations illustrated in the previous figure - PackageReleasableData.

Table 42 - PackageReleasableData Elements	
Element Name	**Element and Operation Descriptions**
LogTransaction	The Log Transaction feature compiles fragments of a PPS transaction log history (from message receipt to the release required data element) and prepares the transaction log for release to the TLS.
	Element Type: Class Owned Operations: Receive-LogElement: The PPS must provide features that receive a transaction log fragment from an IEF component. Integrate-LogElement: The PPS must provide features that integrate a log fragment into the transaction log structure, which tracks PEP and PPS activity from the receipt of a data message until the transaction is concluded through the execution of all watchpoint data releases. Format-LogReport:

Table 42 - PackageReleasableData Elements	
Element Name	**Element and Operation Descriptions**
	The PPS must provide features that transform the transactional log object into the log report format for release to the TLS and user-specified logging system. Issue-LogReport: The PPS must provide features that push the log report to the TLS for release.
Package-TransactionalObject	The PPS provides features that aggregate, transform, label (/mark), and redact data objects.
	Element Type: Class Owned Operations: Package-TransactionalObject: The PPS must provide features that prepare the aggregated, transformed, and labeled objects into a self-contained package for inclusion within another transactional or semantic element. Orchestrate-SubtendedTransactionalProcessing: The PPS must provide features that stage the aggregation, transformation, labeling, and redaction of subtended transactional elements. Retrieve-SubtendedObjects: The PPS must provide features that retrieve subtended objects from the wrapper or semantic memory for inclusion within the transactional pattern. Enforce-ContextFilters: The PPS must provide features that filter transactional objects based on their data or metadata content. These filters redact data objects that do not meet specific user requirements. Enforce-ConfigurationParameters: The PPS must provide features that apply or enforce specific configuration parameters found in the Transactional meta-definition or the IES specification. Executed-DataTransformations: The PPS must provide features that link the transactional packaging to specific data transformations of filters that tailor the data to the exchange semantics.
PackageReleasableData	Packaging of releasable Data includes aggregating, transforming, labeling (/marking), and redacting data elements per semantic policy. These features ensure that the released data content conforms to the recipients' needs and authorizations. The features transform the data elements from the semantics of the user data store to those of the exchange semantic (/canonical model).

`

Table 42 - PackageReleasableData Elements	
Element Name	**Element and Operation Descriptions**
	Element Type: Class Owned Operations: OrchastrateDataPackaging: The PPS must provide features that stage the release of data to authorized recipients.
PackageSemantic	The PPS prepares the data and metadata for release verification and routing.
	Element Type: Class Owned Operations: Orchestrate-TransactionalProcessing: The PPS must provide features that stage the packaging of transactional objects (/elements) per semantic policy. Prepare-ReleasableDataObject: The PPS may provide features that prepare the data objects for release to the exchange services. These preparations are specific to the user's implementation of the specification. Prepare-ReleasableMetadataObject: The PPS may provide features that prepare the metadata objects for release to the exchange services. These preparations are specific to the user's implementation of the specification.
SemanticPolicy	The Semantic Policy represents a serialized version of the IEPPV policy model for the specific domain's data exchange and storage models. The serialized model is ingested at runtime and used to inform PPS processing, memory management, and packaging operations.
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
TransformationLibrary	The set of methods used to transform or filter data during processing and packaging data elements. The library represents a dynamically linked library of methods or operations in the implementation language. (e.g., TransformationLIB.JAR) This transformation library may also be used to dynamically link to formatting and binding services needed to finalize messages for release.

Table 42 - PackageReleasableData Elements	
Element Name	**Element and Operation Descriptions**
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
WrapperObject	The PPS retrieves one wrapper object from memory corresponding to the transactional build's UUID. The wrapper also holds the identifiers for subtended transactions in the build pattern. The semantic policy maintains the descriptions of each wrapper and its relationship to the transactional pattern.
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.

11.1.5 InformationExchangeControl

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

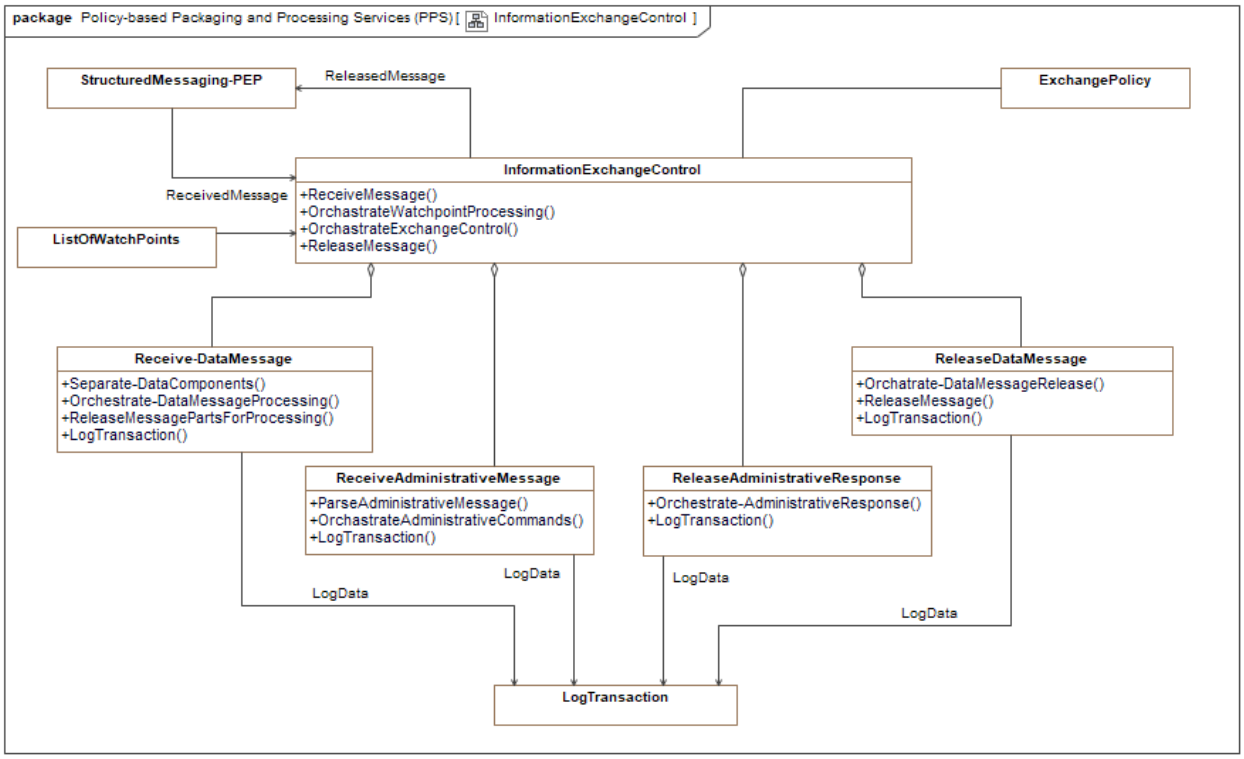


Figure 77 -InformationExchangeControl

`

The following table identifies and describes the elements and operations illustrated in the previous figure - InformationExchangeControl.

Table 43 - InformationExchangeControl Elements	
Element Name	**Element and Operation Descriptions**
ExchangePolicy	The exchange policy specifies the rules and constraints governing the release of information from the PPS and represents a serialization of the information exchange specification from the IEPPV.
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
InformationExchangeControl	The Information Exchange Controls enforce the Information Exchange Specifications (IES) policies (or exchange policies) expressed using the IEPPV. The IES expresses policies (rules and constraints) governing the release of information elements (/messages) permitted for release on each IES (e.g., Community of Interest).
	Element Type: Class Owned Operations: ReceiveMessage: The PPS must provide features that receive data from a PEP. OrchastrateWatchpointProcessing: The PPS must provide features that orchestrate the packaging of all semantics triggered by watchpoints in wrapper memory. OrchastrateExchangeControl: The PPS must provide features that stage the processing of a message received from a PEP. ReleaseMessage: The PPS must provide features that route a formatted message to a specified PEP for release to the authorized recipients.
ListOfWatchPoints	A list of semantic elements corresponding to active information exchange specifications that are triggered by data received in a message.
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.

Table 43 - InformationExchangeControl Elements	
Element Name	**Element and Operation Descriptions**
LogTransaction	The Log Transaction feature compiles fragments of a PPS transaction log history (from message receipt to the release required data element) and prepares the transaction log for release to the TLS.
	Element Type: Class Owned Operations: Receive-LogElement: The PPS must provide features that receive a transaction log fragment from an IEF component. Integrate-LogElement: The PPS must provide features that integrate a log fragment into the transaction log structure, which tracks PEP and PPS activity from the receipt of a data message until the transaction is concluded through the execution of all watchpoint data releases. Format-LogReport: The PPS must provide features that transform the transactional log object into the log report format for release to the TLS and user-specified logging system. Issue-LogReport: The PPS must provide features that push the log report to the TLS for release.
Receive-DataMessage	When the PPS receives an administrative message from an authorized PAP, it stages the processing of the embedded administrative commands.
	Element Type: Class Owned Operations: Separate-DataComponents: The PPS must provide features that separate a message into its parts for processing. Orchestrate-DataMessageProcessing: The PPS must provide features that stage (orchestrate) the parts for processing (e.g., parsing, mapping, transformation, and marshaling). ReleaseMessagePartsForProcessing: The PPS must provide features that release parts of the message to the parsing and mapping components for processing. LogTransaction: The PPS must provide features that prepare and push processing activity log elements for integration into the transactional log report.

`

Table 43 - InformationExchangeControl Elements	
Element Name	**Element and Operation Descriptions**
ReceiveAdministrativeMessage	When it receives an administrative message from an authorized PAP, the PPS stages the processing of the embedded administrative commands.
	Element Type: Class Owned Operations: ParseAdministrativeMessage: The PPS must provide features that decompose an administrative message and extract the commands. This message must be received through a PAP-PEP. OrchastrateAdministrativeCommands: The PPS must provide features that execute PAP commands, including but not limited to: • Activate/deactivate information exchange specifications; • Add an information exchange specification; • Modify and information specification; • Configure and information specification; • Create a filteredSemantic Specification; • Load policy models; • Update Policy models; • Load PPS Configuration; • Report/store PPS Configuration and • Report/sore PPS policies. LogTransaction: The PPS must provide features that prepare and push processing activity log elements for integration into the transactional log report
ReleaseAdministrativeResponse	The PPS provides an administrative response to each PAP command.
	Element Type: Class Owned Operations: Orchestrate-AdministrativeResponse: The PPS must provide features that stage the release of administrative data to the PAP, including: • Configuration Reports; • Policy Reports; • Alerts and warnings and

Table 43 - InformationExchangeControl Elements	
Element Name	**Element and Operation Descriptions**
	• Status Reports. LogTransaction: The PPS must provide features that prepare and push processing activity log elements for integration into the transactional log report
ReleaseDataMessage	Upon receipt of releasable metadata and data from the semantic packaging element, the PPS orchestrates the release of the data to the appropriate exchange channels (Information Exchange Specifications).
	Element Type: Class Owned Operations: Orchatrate-DataMessageRelease: The PPS must provide features that stage the release of data to authorized recipients ReleaseMessage: The PPS must provide features that route a formatted message to the PEP specified in the IES configuration. LogTransaction: The PPS must provide features that prepare and push processing activity log elements for integration into the transactional log report.
StructuredMessaging-PEP	As a specialization of the PEP, the StructuredMessaging-PEP provides external interfaces between: 1. System or user application and a protected data store or 2. systems and applications. The StructuredMessaging-PEP acts as a proxy service that authenticates each user and authorizes the receipt or release of each data message. Clause 10.4 provides details on this interface.
	Element Type: Class Owned Operations: Process_AdministrationCommand: The StructuredMessaging-PEP must provide features that orchestrate the authorization of rejection of an administration command to the PEP or other IEF component in the environment. An administration command must include: • User's Identity or token; • Reference to PAP being used and • Release instructions (e.g., communication channel and messaging protocol) for the administration response.

`

Table 43 - InformationExchangeControl Elements	
Element Name	**Element and Operation Descriptions**

	On receipt of an administrative command from a PAP, the PEP performs the following:
	• Extract and parse the message header and metadata;
	• Gather the user's Attributes;
	• Gather the source of the message;
	• Gather the access confidentiality attributes for the data node;
	• (optionally) Gathers the location and other SA information.
	• package and issue an adjudication request to the PDP or ABAC System;
	• Enforces the PDP or ABAC response:
	  ○ If permitted, package and issue the command(s) to the specified component for processing;
	  ○ If denied, block the request and
	  ○ if indeterminant, enforce user-defined policy and
	• Log the transaction to the TLS.
	Process_UserDataRequest:
	The StructureMessaging-PEP must provide 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 user's 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).
	The PEP then performs the following:
	• Gather the users' attributes;
	• Gather the access confidentiality attributes for the data node;
	• (optionally) Gathers the location and other SA information;
	• package and issue an adjudication request to the PDP or ABAC System;

Table 43 - InformationExchangeControl Elements	
Element Name	**Element and Operation Descriptions**

	• Enforces the PDP or ABAC response:
	  ○ If permitted, he PEP packages and issues a request to the PPS for processing;
	  ○ If denied, Blocks the request;
	  ○ if indeterminant, enforce user-defined policy and
	• Log the transaction to the TLS.
	
	Process_DataMessageReceipt:
	The Messaging-PEP must provide 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:
	• Extract and parse the message header and metadata;
	• Identify the message type.
	• Gather the rules governing the structure and content of the message type.
	• De-construct the message to collect the embedded information elements.
	• Extract the metadata about each information element, which may include:
	  ○ Message metadata - data elements describing the content of the information element (/Message);
	  ○ Data Owner Metadata - data elements identifying the owner/steward of the content of the information element (/Message);
	  ○ Privacy Metadata - data elements describing the private content and release restriction associated with the information element (/Message);
	  ○ Security (Confidentiality) Metadata - data elements describing the classified content and release restriction associated with the information element (/Message) and
	  ○ HandlingInstructions - data elements describing any specialized data/information access, processing, or storage instructions for the information element (/Message);
	• Request the recipient's attributes from the user-specified AD or ICAM system;
	• (Optional) Request the operational context for the exchange from the user's SA or incident management system.
	• Packages and issues an authorization request to the PDP.

`

Table 43 - InformationExchangeControl Elements	
Element Name	**Element and Operation Descriptions**
	• If authorized, decrypt the Payload (information element(s)) provided in the message: • Packages and issues the Metadata and decrypted information element to the PPS for processing and • Logs the transaction to the TLS. Process_DataMessageRelease: The Messaging-PEP must provide features that stage the processing of an information element(s) from a PPS for release to the user-specified middleware. The SMB messaging protocol is stripped by the interface prior to the commencement of processing. The Messaging-PEP then performs the following: • Extract 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. • Request the recipient(s) attributes from the LDAP or ICAM System; • (Optional) Requests the operational context for the exchange from the user's SA or incident management system; • Package and issue an authorization request to the PDP; • If authorized, encrypt the authorized information element(s): ○ Requests Cryptographic Keys and tokens from the user-specified Key Management Service(s) (SSG-Request); ○ Packages a CTS-Request for the encryption of each authorized information element and ○ Packages and issues the Message to the middleware for dissemination and • Logs the transaction to the TLS. Manage_PEP:

Table 43 - InformationExchangeControl Elements	
Element Name	**Element and Operation Descriptions**
	The Messaging-PEP must provide 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: The above functions utilize the following inherited operations to deliver their features. Inherited functions include: Manage-PEPOperations inherited from PEP Execute-AdministrationFunctions inherited from PEP Gather-UserAttributeData inherited from PEP Gather-RecipientLocation inherited from PEP Gather-IdentityData inherited from PEP Generate-DecisionRequest inherited from PEP Enforce-AuthorizationDecisions inherited from PEP Store-PEPConfiguration inherited from PEP Restore-PEPConfiguration inherited from PEP Publish-PEPConfiguration inherited from PEP Load-PEPconfiguration inherited from PEP Generate_ProcessingUUID inherited from PEP inherited from IEF_Component 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

`

11.1.5.1 ReceiveDataMessage

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

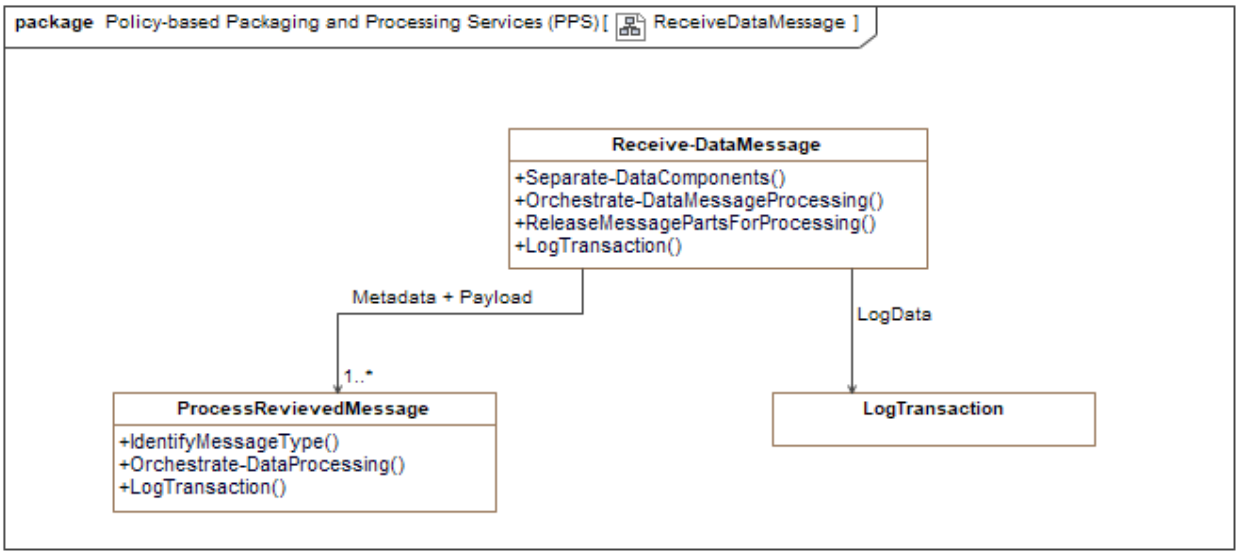


Figure 78 -ReceiveDataMessage

The following table identifies and describes the elements and operations illustrated in the previous figure - ReceiveDataMessage.

Table 44 - ReceiveDataMessage Elements	
Element Name	**Element and Operation Descriptions**
LogTransaction	The Log Transaction feature compiles fragments of a PPS transaction log history (from message receipt to the release required data element) and prepares the transaction log for release to the TLS.
	Element Type: Class Owned Operations: Receive-LogElement: The PPS must provide features that receive a transaction log fragment from an IEF component. Integrate-LogElement: The PPS must provide features that integrate a log fragment into the transaction log structure, which tracks PEP and PPS activity from the receipt of a data message until the transaction is concluded through the execution of all watchpoint data releases. Format-LogReport:

Table 44 - ReceiveDataMessage Elements	
Element Name	**Element and Operation Descriptions**
	The PPS must provide features that transform the transactional log object into the log report format for release to the TLS and user-specified logging system. Issue-LogReport: The PPS must provide features that push the log report to the TLS for release.
ProcessReceivedMessage	Processing received data includes parsing, mapping, transformation, and marshaling data elements and attributes to the user data store. The features transform the data elements from the exchange semantic (canonical exchange model for the received message) to the storage semantic of the data store.
	Element Type: Class Owned Operations: IdentifyMessageType: The PPS must provide features that identify the type of message received from the PEP. Orchestrate-DataProcessing: The PPS must provide features that orchestrate the processing of all elements (e.g., metadata (/manifest) and payloads in the message. LogTransaction: The PPS must provide features that prepare and push processing activity log elements for integration into the transactional log report.
Receive-DataMessage	When the PPS receives an administrative message from an authorized PAP, it stages the processing of the embedded administrative commands.
	Element Type: Class Owned Operations: Separate-DataComponents: The PPS must provide features that separate a message into its parts for processing. Orchestrate-DataMessageProcessing: The PPS must provide features that stage (orchestrate) the parts for processing (e.g., parsing, mapping, transformation, and marshaling). ReleaseMessagePartsForProcessing: The PPS must provide features that release parts of the message to the parsing and mapping components for processing. LogTransaction:

`

Table 44 - ReceiveDataMessage Elements	
Element Name	**Element and Operation Descriptions**
	The PPS must provide features that prepare and push processing activity log elements for integration into the transactional log report.

11.1.5.2 ReleaseDataMessages

The following figure identifies the features and functions provided by a Policy-based Packaging and Processing Service to ensure that messages are only released to recipients who need and are authorized to receive the content.

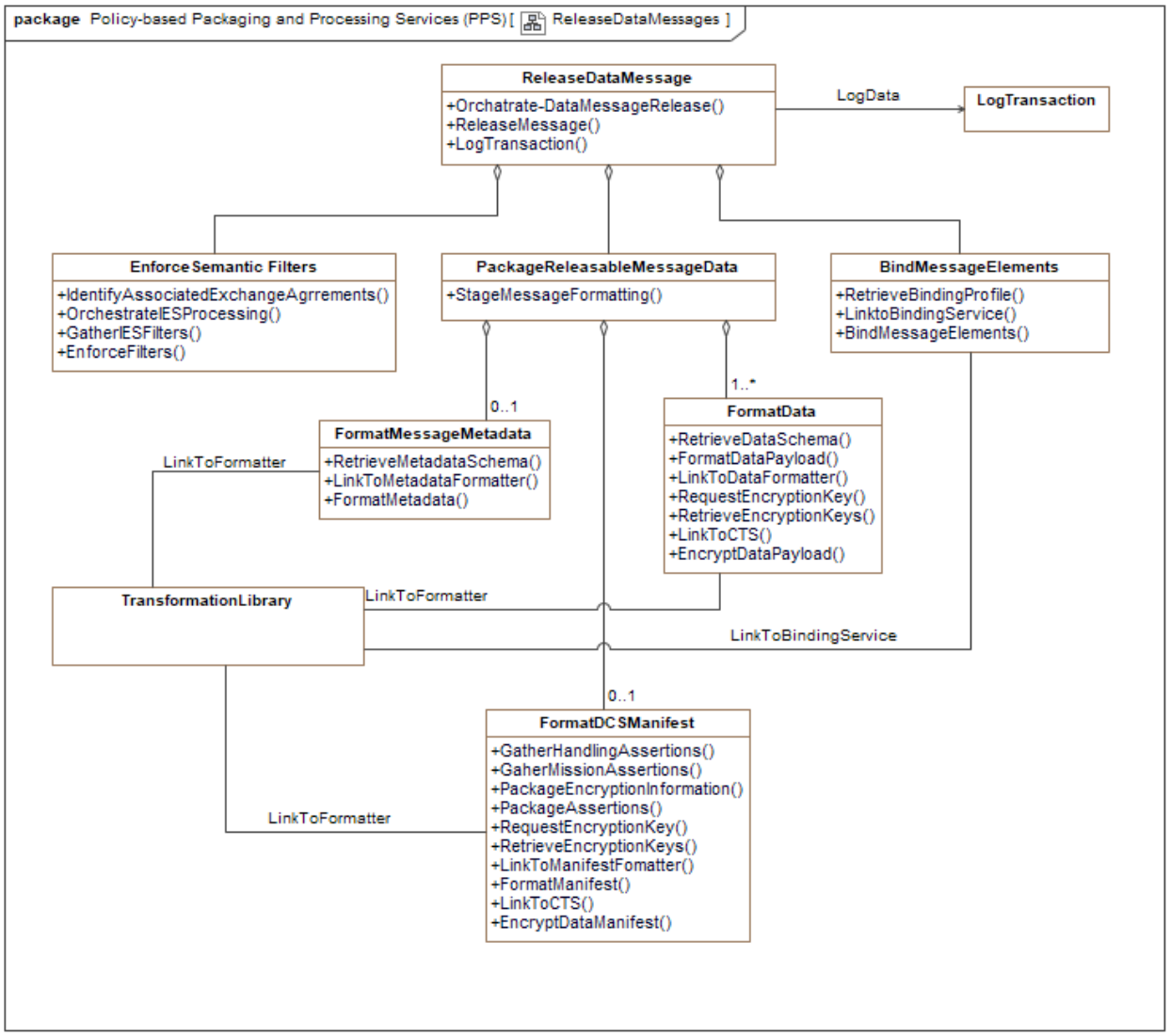


Figure 79 -ReleaseDataMessages

The following table identifies and describes the elements and operations illustrated in the previous figure - ReleaseDataMessages.

Table 45 - ReleaseDataMessages Elements	
Element Name	**Element and Operation Descriptions**
BindMessageElements	The PPS binds the metadata to the Data Messages and Payloads per exchange policy (/IES Configuration). The Binding may include STANAG 4778, ZTDF (/TDO), ACP240, or IEF SAC.
	Element Type: Class Owned Operations: RetrieveBindingProfile:

`

Table 45 - ReleaseDataMessages Elements	
Element Name	**Element and Operation Descriptions**
	The PPS must provide features that retrieve the binding profile (e.g., STANAG 4778, OpenTDF, ZTDF, or ACP 240) specified in the exchange policy (e.g., IES configuration parameters). LinktoBindingService: The PPS must provide features that dynamically link it to the software services that enforce the specified binding profile. Binding services should include STANAG 4778, ZTDF (/TDO), ACP240, or IEF SAC. BindMessageElements: The PPS must provide features that format the message in accordance with the specified binding profile.
EnforceSemantic Filters	The PPS provides a second level of filters that restrict data release to each information-sharing agreement (or IES) based on specified data and metadata filters. These filters are defined in exchange policies for the PPS but are configurable during operations. They provide a guard function for the PPS. They differ from the PEP filters in that the PEP filters limit their operation to metadata bound to the message.
	Element Type: Class Owned Operations: IdentifyAssociatedExchangeAgrrements: The PPS must provide features that identify the active sharing agreements (e.g., IES) for the data set being processed. A specific data set may be applicable to multiple sharing agreements. OrchestrateIESProcessing: The PPS must provide features that stage the verification of the data content's releasability against each exchange policy. Data is only formatted and released if it meets the filter requirements for that IES. Data that does not conform to the filters in the exchange policy is blocked from further processing and release. GatherIESFilters: The PPS must provide features that gather the IES filters to the specific semantic under the IES being evaluated. Filters are gathered from the exchange policies for the PPS. EnforceFilters: The PPS must provide filters to assess the released data using the IES filters, only releasing the data if the filter criteria are met.

Table 45 - ReleaseDataMessages Elements	
Element Name	**Element and Operation Descriptions**
FormatData	The PPS provides the features needed to format the data according to the structure and syntax specified in the data exchange schema or profile.
	Element Type: Class Owned Operations: RetrieveDataSchema: The PPS must provide features that retrieve the message or data schema specified in the IES from the exchange schema library. FormatDataPayload: The PPS must provide features that format the message per the IES configuration. LinkToDataFormatter: The PPS must provide features that link processing to the appropriate formatting service for the message protocol being released. RequestEncryptionKey: The PPS must provide features that request an encryption key from the key management system via the SSG. RetrieveEncryptionKeys: The PPS must provide features that generate or retrieve the encryption keys (e.g., symmetric or PKI). LinkToCTS: The PPS must provide features that link processing to CTS services. EncryptDataPayload: The PPS must provide features that encrypt the data payload.
FormatDCSManifest	The PPS may provide features that aggregate and format heading and mission assertions. Assertions support storing, handling information, labeling, or marking metadata.
	Element Type: Class Owned Operations: GatherHandlingAssertions: The PPS must provide features that gather or generate the handling instructions for the generated message. GaherMissionAssertions: The PPS must provide features that gather or generate the mission assertion for the generated message. PackageEncryptionInformation:

`

Table 45 - ReleaseDataMessages Elements	
Element Name	**Element and Operation Descriptions**
	The PPS must provide features that gather or generate the Encryption information for the generated message. PackageAssertions: The PPS may provide features that package (aggregate and format) heading and mission assertions. Assertions support storing, handling information, labeling, or marking metadata. RequestEncryptionKey: The PPS must provide features that request an encryption key from the key management system via the SSG. RetrieveEncryptionKeys: The PPS must provide features that generate or retrieve the encryption keys (e.g., symmetric or PKI). LinkToManifestFomatter: The PPS must provide features that link processing to the appropriate formatting and binding service for the message protocol being released. FormatManifest: The PPS must provide features that format the message manifest according to the message protocol specification (e.g., ZTDF and ACP240). LinkToCTS: The PPS must provide features that link processing to CTS services. EncryptDataManifest: The PPS must provide features that encrypt the data manifest per binding specifications (e.g., ZTDF and ACP240).
FormatMessageMetadata	The PPS provides features that enable the formatting of metadata to multiple schemas and manifest formats and syntaxes.
	Element Type: Class Owned Operations: RetrieveMetadataSchema: The PPS must provide features that retrieve a metadata schema from a library of schema loaded at runtime. LinkToMetadataFormatter: The PPS must provide features that dynamically load metadata formatting services corresponding to the specified metadata schema and syntax. FormatMetadata:

Table 45 - ReleaseDataMessages Elements	
Element Name	**Element and Operation Descriptions**
	The PPS must provide features that format message metadata per exchange policy.
LogTransaction	The Log Transaction feature compiles fragments of a PPS transaction log history (from message receipt to the release required data element) and prepares the transaction log for release to the TLS.
	Element Type: Class Owned Operations: Receive-LogElement: The PPS must provide features that receive a transaction log fragment from an IEF component. Integrate-LogElement: The PPS must provide features that integrate a log fragment into the transaction log structure, which tracks PEP and PPS activity from the receipt of a data message until the transaction is concluded through the execution of all watchpoint data releases. Format-LogReport: The PPS must provide features that transform the transactional log object into the log report format for release to the TLS and user-specified logging system. Issue-LogReport: The PPS must provide features that push the log report to the TLS for release.
PackageReleasableMessageData	The PPS stages (/orchestrates) the packaging (aggregation and formatting) of message elements for release.
	Element Type: Class Owned Operations: StageMessageFormatting: The PPS must provide features that orchestrate the aggregation and formatting of message elements for release.
ReleaseDataMessage	Upon receipt of releasable metadata and data from the semantic packaging element, the PPS orchestrates the release of the data to the appropriate exchange channels (Information Exchange Specifications).

`

Table 45 - ReleaseDataMessages Elements	
Element Name	**Element and Operation Descriptions**
	Element Type: Class Owned Operations: Orchatrate-DataMessageRelease: The PPS must provide features that stage the release of data to authorized recipients ReleaseMessage: The PPS must provide features that route a formatted message to the PEP specified in the IES configuration. LogTransaction: The PPS must provide features that prepare and push processing activity log elements for integration into the transactional log report.
TransformationLibrary	The set of methods used to transform or filter data during processing and packaging data elements. The library represents a dynamically linked library of methods or operations in the implementation language. (e.g., TransformationLIB.JAR) This transformation library may also be used to dynamically link to formatting and binding services needed to finalize messages for release.
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.

11.2 PPS Configuration

The PPS provides an interface to the IEF Secure Messaging Bus (SMB), which isolates it from external message traffic. Ultimately, it is located within its own Virtual Machine (VM) with the data store it is protecting. The VM is protected by its secure operating system (OS), virtual firewall, and virtual networks (e.g., SMB), which are configured to minimize the vectors of attack to both the PPS and the data store. All communications to and from the PPS are conducted through policy enforcement points tailored to the systems and middleware enabling the data exchanges. This integration of data-centric and traditional cybersecurity provides data as a resource conforming to zero-trust principles.

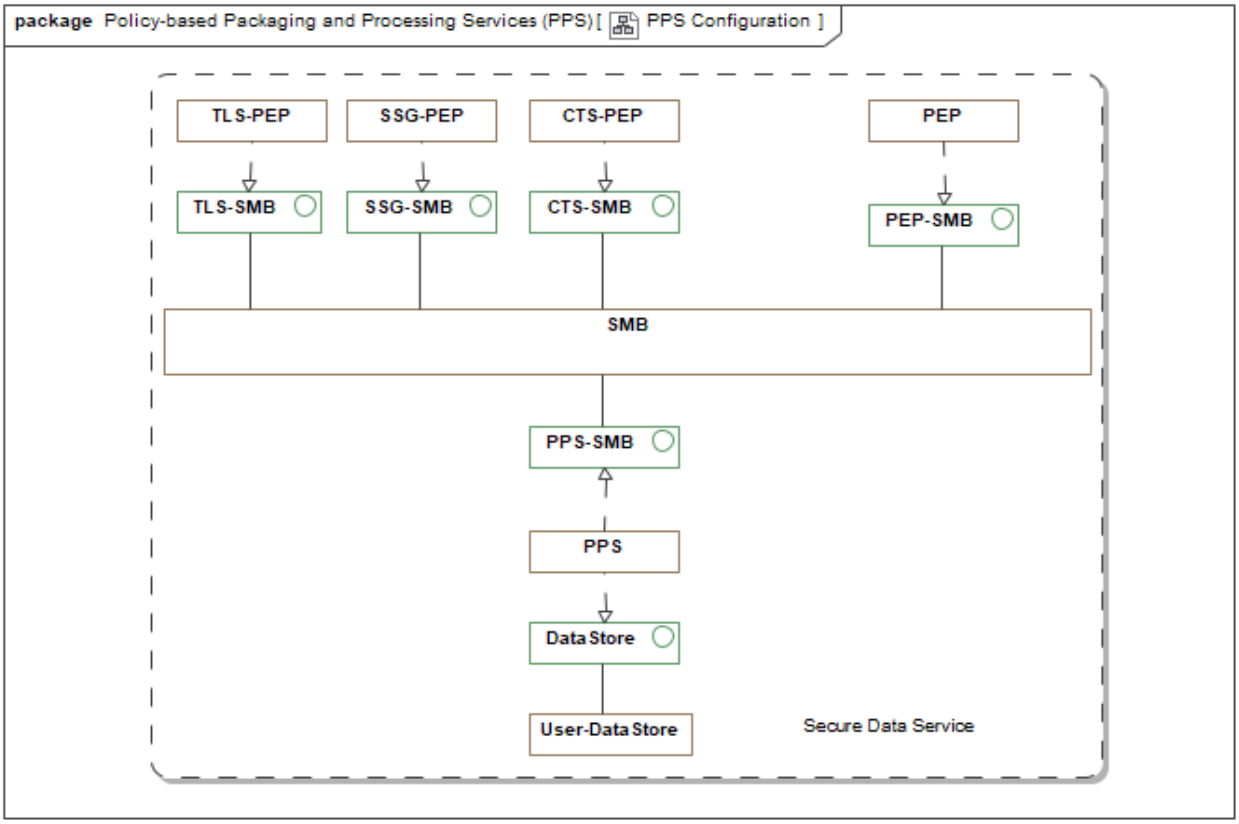


Figure 80 -PPS Configuration

`

12 Security Service Gateway (SSG)

The Security Services Gateway (SSG) provides a single secure interface between IEF components and the user's specified security services and infrastructure. The SSG 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 SSG translates the requests and responses between IEF protocols and user networking and security system protocols.

12.1 SSG Component Operations

The following figure outlines the features provided by an IEF security services gateway (SSG).

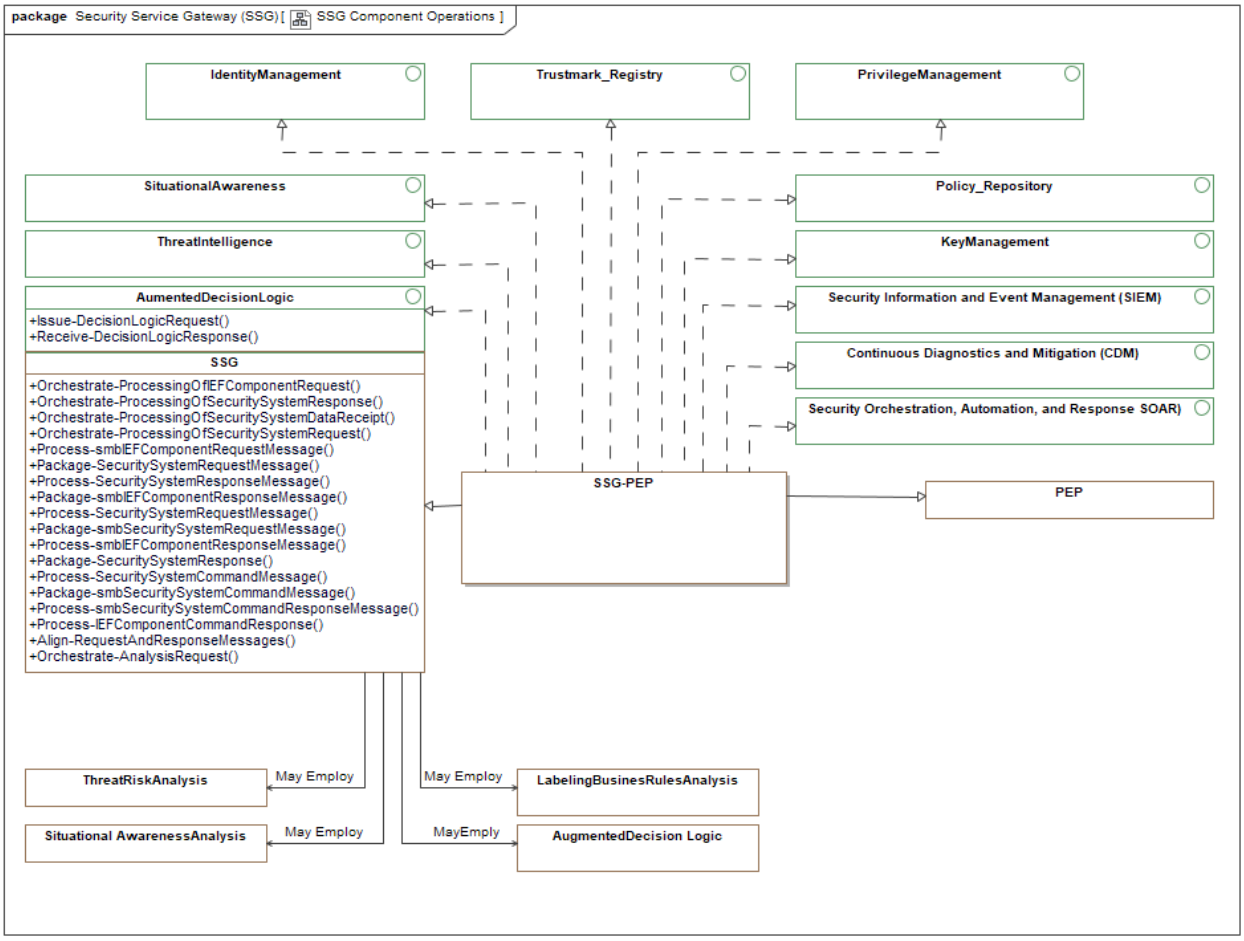


Figure 81 -SSG Component Operations

The following table identifies and describes the elements and operations illustrated in the previous figure - SSG Component Operations.

Table 46 - SSG Component Operations Elements	
Element Name	**Element and Operation Descriptions**
AugmentedDecision Logic	The SSG may provide interface(s) that enable IEF components to interoperate with Analytics, AI, and Machine Learning systems that augment their decision logic. These interfaces would enable IEF components to utilize augmented decision logic to protect the IEF and data.
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
LabelingBusinesRulesAnalysis	The user may choose to implement analytic (e.g., AI, Analytics, or Machine learning) services that receive data sets from IEF components and provision data labels (/metadata) to be integrated into the released data sets.
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
PEP	The Policy Enforcement Point (PEP) intercepts all user and system requests to access or release data elements protected by IEF services. The PEP gathers information about the data elements, sender, recipients, and communication channel, packages an adjudication request to the PDP, and enforces the PDP access or release determination. In a zero-trust environment, the PEP interface authenticates and authorizes each transaction with an external cryptographic service. The PEP must provide one or both of the feature sets defined by the SMB or Direct Connect options.
	Element Type: Class Owned Operations: Manage-PEPOperations: The PEP must provide features that manage the execution of PEP functions in each potential configuration (e.g., Direct connect, SMB connection, or a combination of direct and SMB configurations). Execute-AdministrationFunctions: The PEP must provide features that execute AdministrativeCommands from an authorized Policy Administration Point. PEP administrative functions include: 1. Activate PEP features; 2. Deactivate PEP features; 3. Configure PEP Parameters;

`

Table 46 - SSG Component Operations Elements	
Element Name	**Element and Operation Descriptions**
	4. Archive PEP Operational Environment; 5. Publish PEP Configuration; 6. Store PEP Configuration; and 7. Retrieve PEP Configuration; Gather-UserAttributeData: The PEP must provide features that gather the sender and receiver attributes to determine their authorization to receive data elements. Gather-RecipientLocation: (Optional) The PEP provides features that gather information about the recipients' location (physical or electronic). A recipient's location(s) (e.g., network location, physical location, and device) may impact the user's attributes and the information content they are authorized to receive. These features may only be available if the IEF can request the information from the user's situational awareness, incident management, or network management systems. Gather-IdentityData: The PEP must provide features that gather the identity information for the sender and recipients of the specified information element (s). These features allow users to request this information from the infrastructure services that provide identity management. All requests to the users' specified infrastructure are issued through the Security Services Gateway using an SSG-Request message. Generate-DecisionRequest: The PEP must provide features that generate a decision request to the PDP. Enforce-AuthorizationDecisions: The PEP Provides features that enforce PDP authorization decisions for each data receipt and release. Store-PEPConfiguration: The PEP Provides features that gather and store its configuration parameters in local storage. Restore-PEPConfiguration: The PEP Provides features that retrieve and load its configuration parameters from local storage. Publish-PEPConfiguration: The PEP must provide features that gather and publish its configuration parameters to the PAP.

Table 46 - SSG Component Operations Elements	
Element Name	**Element and Operation Descriptions**
	Load-PEPconfiguration: The PEP must provide features that receive and load its configuration parameters from the PAP. Generate_ProcessingUUID: Upon receiving a message, the PEP must provide features that generate a universally unique identifier (UUID). The structure of the UUID must identify the node receiving the data. If the received message includes a UUID, the PEP must adopt the received UUID. Inherited Operations: The above functions utilize the following inherited operations to deliver their features. Inherited functions include: inherited from IEF_Component 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
Situational AwarenessAnalysis	The user may choose to implement analytic (e.g., AI, Analytics, or Machine learning) services that monitor cyber and operational situational awareness sources and adapt IEF policies and configurations to adapt service operations to changing situations.
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
SSG	The Security Services Gateway provides a generalized integration point between IEF Components and user-specified security services, including ICAM, ABAC, Key Management, Cryptography, SIEM, SOAR, and CDM. The SSG conforms to ZTA principles. The following features apply to each combination of the IEF component

Table 46 - SSG Component Operations Elements	
Element Name	**Element and Operation Descriptions**
	and security system (/service), requiring implementations tailored to the combination or services. The SSG may be implemented as a single integration point for all security systems and services or as an integration point tailored to each user-specified security system (/service) API.
	Element Type: Class Owned Operations: Orchestrate-ProcessingOfIEFComponentRequest: The SSG must provide features that stage the processing (e.g., parsing and mapping) of data or service requests from IEF components to user-specified systems and services. The component then packages the request in the required format and directs it to one or more user-specified security systems (e.g., Threat Intelligence, Cyber SA, SIEM, and SOAR). Orchestrate-ProcessingOfSecuritySystemResponse: The SSG must provide features that stage the processing (e.g., parsing and mapping) of data or service responses from the user systems and services, package the response as an SMB message, and direct the message to one or more IEF components (e.g., PEP, or PPS). Orchestrate-ProcessingOfSecuritySystemDataReceipt: The SSG must provide features that stage the processing (e.g., parsing and mapping) of a security system request for data (e.g., status or configuration), package SMB messages for one or more IEF components, and issue them to the IEF Component(s). Orchestrate-ProcessingOfSecuritySystemRequest: The SSG must provide features that stage the processing (e.g., parsing and mapping) of a security system request for data (e.g., status or configuration), package SMB messages to one or more IEF components, and issue them to the IEF Component(s). Process-smbIEFComponentRequestMessage: The SSG must provide features that process (e.g., parse and map) messages from an IEF component via the SMB requesting data or service from a user-specified security system or service within the operating environment. These features apply to each combination of an IEF component and security system (/service). Package-SecuritySystemRequestMessage: The SSG must provide features that package (e.g., aggregate and format) messages from the IEF components requesting data

Table 46 - SSG Component Operations Elements	
Element Name	**Element and Operation Descriptions**
	or services from the security system (/service) per its API Specification. Process-SecuritySystemResponseMessage: The SSG must provide features that process (e.g., parse and map) a security system (/service) response per its API Specification. Package-smbIEFComponentResponseMessage: The SSG must provide features that package (e.g., aggregate and format) the Security system (/service) response per the SMB response messages for the specified IEF component(s). Process-SecuritySystemRequestMessage: The SSG must provide features that process (e.g., parse and map) a security system (/service) request for data from an IEF component per its API Specification. Package-smbSecuritySystemRequestMessage: The SSG must provide features that package (e.g., aggregate and format) a request for data request (e.g., status and configuration) per the SMB request messages for the specified IEF component(s). Process-smbIEFComponentResponseMessage: The SSG must provide features that process (e.g., parse and map) a security data response message from an IEF component via the SMB. Package-SecuritySystemResponse: The SSG must provide features that package (e.g., aggregate and format) security data response messages from the IEF components to a security system (/service) per its API Specification. Process-SecuritySystemCommandMessage: The SSG may provide features that process (e.g., parse and map) a security system (/service) command message for an IEF component per its API Specification. Command messages from authorized SIEM, SOAR, or CDM systems are intended to direct IEF components to adapt their configuration or policies to mitigate threats or risks to the data environment. Package-smbSecuritySystemCommandMessage: The SSG may provide features that package (e.g., aggregate and format) security system (/service) commands for IEF components per the SMB request messages. Process-smbSecuritySystemCommandResponseMessage: The SSG must provide features that process (e.g., parse and map) a security command response message from an IEF component via the SMB.

`

Table 46 - SSG Component Operations Elements	
Element Name	**Element and Operation Descriptions**
	Process-IEFComponentCommandResponse: The SSG must provide features that package (e.g., aggregate and format) security command response messages from the IEF components to a security system (/service) per its API Specification. Align-RequestAndResponseMessages: The SSG must provide features that align data, service and command messages with their responses from the IEF components or security systems. Orchestrate-AnalysisRequest: The SSG may provide features that stage the integration of the IEF components with Analytical, AI, or Machine Learning systems that augment IEF capabilities and decision-making logic in the areas of threat-risk analysis, situational awareness, data labeling, and decision logic. Inherited Operations: The above functions utilize the following inherited operations to deliver their features. Inherited functions include: inherited from IEF_Component 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
SSG-PEP	The SSG-PEP is a hybrid between the SSG and a PEP, providing ZT security elements to the interfaces between the IEF and the user's security infrastructure. It connects the IEF Services to Security services provided by the user or the local infrastructure (e.g., Cloud Service Provider (CSP)). It enables the IEF components to interoperate with user-specified security services. (e.g., Identity, credential, access management services, access control services, and

Table 46 - SSG Component Operations Elements	
Element Name	**Element and Operation Descriptions**
	essential management services) and infrastructure using a standardized SMB Interface. (Clause 12)
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration. Inherited Operations: The above functions utilize the following inherited operations to deliver their features. Inherited functions include: Manage-PEPOperations inherited from PEP Execute-AdministrationFunctions inherited from PEP Gather-UserAttributeData inherited from PEP Gather-RecipientLocation inherited from PEP Gather-IdentityData inherited from PEP Generate-DecisionRequest inherited from PEP Enforce-AuthorizationDecisions inherited from PEP Store-PEPConfiguration inherited from PEP Restore-PEPConfiguration inherited from PEP Publish-PEPConfiguration inherited from PEP Load-PEPconfiguration inherited from PEP Generate_ProcessingUUID inherited from PEP inherited from IEF_Component 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 inherited from SSG

`

Table 46 - SSG Component Operations Elements	
Element Name	**Element and Operation Descriptions**
	inherited from SSG inherited from SSG inherited from SSG Orchestrate-ProcessingOfIEFComponentRequest inherited from SSG Orchestrate-ProcessingOfSecuritySystemResponse inherited from SSG Orchestrate-ProcessingOfSecuritySystemDataReceipt inherited from SSG Orchestrate-ProcessingOfSecuritySystemRequest inherited from SSG Process-smbIEFComponentRequestMessage inherited from SSG Package-SecuritySystemRequestMessage inherited from SSG Process-SecuritySystemResponseMessage inherited from SSG Package-smbIEFComponentResponseMessage inherited from SSG Process-SecuritySystemRequestMessage inherited from SSG Package-smbSecuritySystemRequestMessage inherited from SSG Process-smbIEFComponentResponseMessage inherited from SSG Package-SecuritySystemResponse inherited from SSG Process-SecuritySystemCommandMessage inherited from SSG Package-smbSecuritySystemCommandMessage inherited from SSG Process-smbSecuritySystemCommandResponseMessage inherited from SSG Process-IEFComponentCommandResponse inherited from SSG Align-RequestAndResponseMessages inherited from SSG Orchestrate-AnalysisRequest inherited from SSG
ThreatRiskAnalysis	The user may choose to implement analytic (e.g., AI, Analytics, or Machine learning) services that monitor threat and risk data sources and adapt IEF policies and configurations to adapt service operations to changing situations.
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.

12.2 SSG Configurations

The following figure identifies the optional IEF Security Services Gateway interfaces, which provide additional integration points between IEF components and user-specified security services and infrastructure, including but not limited to Identity Management, Privilege/Attribute Management, Cryptographic, TrustMark Provider, and Policy Development and Management Environments.

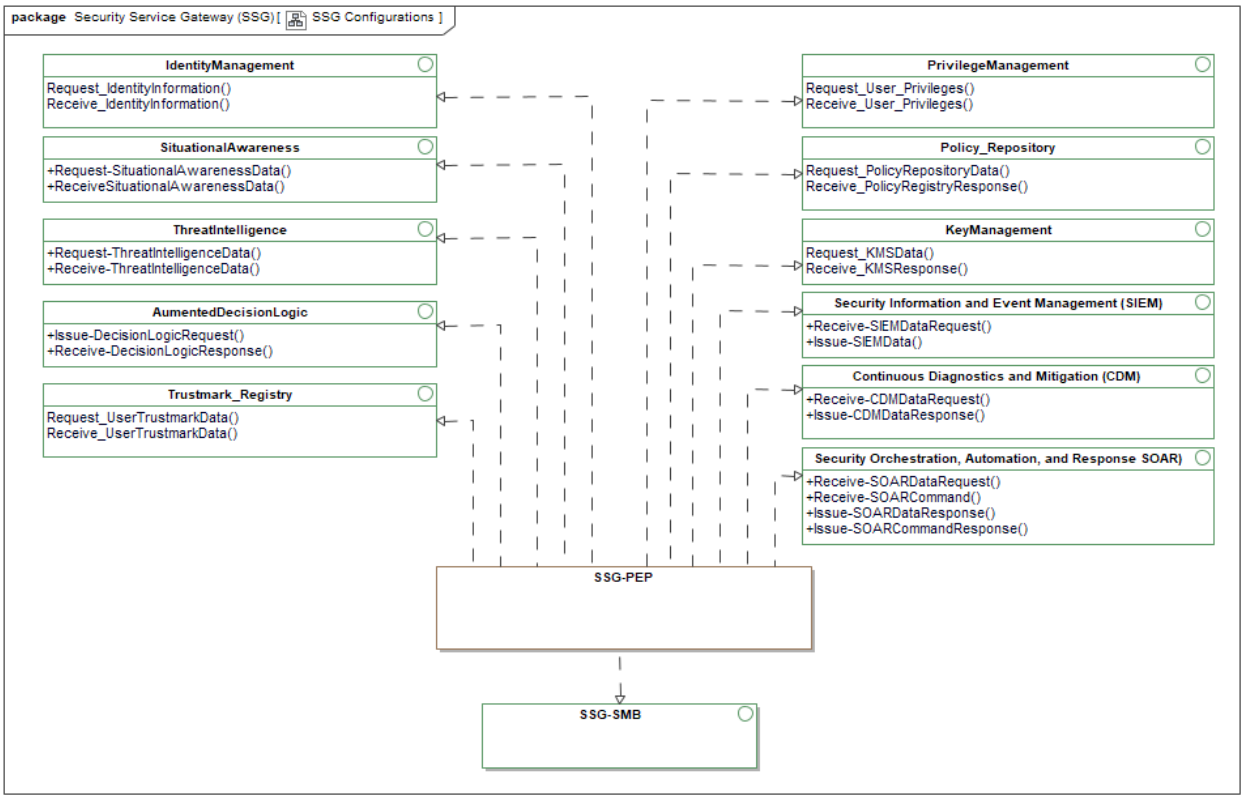


Figure 82 -SSG Configurations

The following table identifies and describes the elements and operations illustrated in the previous figure - SSG Configurations.

Table 47 - SSG Configurations Elements	
Element Name	**Element and Operation Descriptions**
AumentedDecisionLogic	The SSG may provide interface(s) that enable IEF components to interoperate with Analytics, AI, and Machine Learning systems that augment IEF components' decision logic. These interfaces would enable IEF components to utilize augmented decision logic to protect the IEF and data.
	Element Type: Interface Owned Operations: Issue-DecisionLogicRequest: This SSG interface may provide features that receive messages from the SSG, apply the appropriate messaging and network

`

Table 47 - SSG Configurations Elements	
Element Name	**Element and Operation Descriptions**
	protocols, and issue a request for augmented decision logic (/support) from specified analytics, AI, or machine learning systems (/services). Receive-DecisionLogicResponse: This SSG interface must provide features that listen for messages from specified analytics, AI, or machine learning systems (/services), receive the messages, extract the data/information from the applied protocols, and pass it to the SSG for processing.
Continuous Diagnostics and Mitigation (CDM)	The SSG may provide interfaces that enable IEF components to be interoperable with CDM systems and services. It may be separated as an independent SSG or integrated with other Security Services. CDM programs deliver cybersecurity tools, integration services, and dashboards that improve security posture by: • Reducing threat surface; • Increasing visibility into one's cybersecurity posture; • Improving Federal Cybersecurity Response Capabilities and • Streamlining Federal Information Security Modernization Act (FISMA) or other regulatory reporting.
	Element Type: Interface Owned Operations: Receive-CDMDataRequest: The SSG interface must provide features that listen for a message from the user-specified CDM System, receive the CDM data request message, extract the data needed by the responding IEF component, and pass it to the SSG for processing. Issue-CDMDataResponse: This SSG interface must provide features that receive messages from the SSG, bind the messaging and network protocols, and issue the message to the specified CDM system.
IdentityManagement	The SSG may provide interfaces that enable IEF components to interoperate with specified identity management (/identity, credential, and access Management) systems. The User's Identity Management Services are the authoritative source for the user's account and credentials (The IEF does not manage identity). All IEF components requiring access to identity information retrieve that information by communicating through the SSG or ICAM interface.

Table 47 - SSG Configurations Elements	
Element Name	**Element and Operation Descriptions**
	Element Type: Interface Owned Operations: Request_IdentityInformation: This SSG interface must provide features that apply the appropriate messaging and network protocols and issue a request for identity information to the user-specified Identity Management (IdM) Services. Receive_IdentityInformation: The SSG interface provides features that listen for a message from the user-specified Identity Management (IdM) Services, receive the message, extract the data/information needed by the requesting IEF component, and pass it to the SSG for processing.
KeyManagement	The SSG (or PEP) may provide interfaces that enable IEF components to interoperate with user-specified policy Key Management Services (KMS), which manage (e.g., generate, escrow, and provision) the cryptographic keys.
	Element Type: Interface Owned Operations: Request_KMSData: This SSG interface must provide features that apply the appropriate messaging and network protocols and issue requests to the KMS, including Key Generation Key, Key Storage, and Key Retrieval. Receive_KMSResponse: This SSG interface must provide features that listen for messages from the user KMS, intercept them, extract the relevant data, and pass the data to the SSG for processing.
Policy_Repository	The SSG may provide interface(s) that enable IEF components to interoperate with the user's policy development environment or policy management registry/repository to retrieve or publish IEF or DSS policies or IEF component configurations.
	Element Type: Interface Owned Operations: Request_PolicyRepositoryData: This SSG interface must provide features that apply the appropriate messaging and network protocols and issue the request to the user's policy development environment or policy management registry/repository. Receive_PolicyRegistryResponse: This SSG interface must provide features that listen for a message from the user-specified Policy Development or Policy management Environment, receive the policy message, parse

`

Table 47 - SSG Configurations Elements	
Element Name	**Element and Operation Descriptions**
	the message, extract the requisite data, and pass it to the SSG for processing.
PrivilegeManagement	The SSG may provide interface(s) that enable IEF components to interoperate with specified privilege management services or ICAM services. These services provide an authoritative source for users' attributes used to authorize access to IEF services or inform IEF policy enforcement for data receipt, packaging, or release controls.
	Element Type: Interface Owned Operations: Request_User_Privileges: This SSG interface must provide features that apply the appropriate messaging and network protocols and issue a request for user attributes and attributes from the user-specified Privilege (/Attribute/attributes) Management Services. Receive_User_Privileges: The SSG interface must provide features that listen for a message from the user-specified Privileges (/attributes) Management Services, receive the user attribute message, extract the information needed by the requesting IEF component, and pass it to the SSG for processing.
Security Information and Event Management (SIEM)	The SSG may provide an interface that enables IEF components to interoperate with user-specified Security Information and Event Management (SIEM) Systems and provide data needed for threat detection, regulatory compliance, and security incident management. The SIEM system collects and analyzes security and DCS events (both near real-time and historical). SIEM capabilities include log event collection and management, the ability to analyze log events and other data types across disparate sources, and operational capabilities that enable incident management, dashboards, and reporting.
	Element Type: Interface Owned Operations: Receive-SIEMDataRequest: This SSG interface must provide features that listen for a SIEM Data request message from the user-specified SIEM System, receive the message, extract the information needed by the responding IEF component, and pass the data to the SSG for processing. Issue-SIEMData:

Table 47 - SSG Configurations Elements	
Element Name	**Element and Operation Descriptions**
	This SSG interface must provide features that apply the appropriate messaging and network protocols and issue IEF component data to the user-specified SIEM System.
Security Orchestration, Automation, and Response SOAR)	The SSG may provide an interface that enables IEF components to interoperate with user-specified Security Orchestration, Automation, and Response SOAR). SOAR enables the collection of data needed by the security operations team. It collects logs and alerts from IEF, SIEM, and other security systems and technologies. It performs incident analysis and triage using human, Analytics, and Machine Learning to define, prioritize, and drive standardized incident response.
	Element Type: Interface Owned Operations: Receive-SOARDataRequest: The SSG interface must provide features that listen for a SOAR data request message from the user-specified SOAR System, receive the data request message, extract the information needed by the responding IEF component, and pass the data to the SSG for processing. Receive-SOARCommand: The SSG interface must provide features that listen for a SOAR Command message from the user-specified SIEM System, receive the data request message, extract the information needed by the responding IEF component, and pass the data to the SSG for processing. Issue-SOARDataResponse: This SSG interface must provide features that apply the appropriate messaging and network protocols and issue IEF component data to the user-specified SOAR System. Issue-SOARCommandResponse: This SSG interface must provide features that apply the appropriate messaging and network protocols and issue IEF component command responses to the user-specified SOAR System.
SituationalAwareness	The SSG may provide an interface that enables IEF components to interoperate with user-specified cyber and operational situational awareness (SA) systems to inform policy decisions.
	Element Type: Interface Owned Operations: Request-SituationalAwarenessData:

`

Table 47 - SSG Configurations Elements	
Element Name	**Element and Operation Descriptions**
	This SSG interface must provide features that apply the appropriate messaging and network protocols and issue IEF component data requests to the user-specified SA System. ReceiveSituationalAwarenessData: The SSG interface must provide features that listen for SA data messages from the user-specified SOAR System, receive the data message, extract the information needed by the IEF component, and pass the data to the SSG for processing.
SSG-SMB	SMB interface enables the SSG to interoperate with other IEF components (e.g., PEP, PPS, PAP, PDP, TLS, and CTS) in the configuration. In a Zero-Trust implementation, it is responsible for enforcing access and release control policy between IEF components. Messages (Clause 16) applicable to the SSG-SMB interface include: • SSG-Request (receive); • SSG-Response (send); • PAP-Command(receive); • PAP-CommandResponse (send); • PAP-AlertWarning (send); and • TLS-LogMessage (send).
	Element Type: Interface This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
ThreatIntelligence	The SSG may provide an interface that enables IEF components to interoperate with user-specified cyber and operational threat-risk intelligence systems to inform policy decisions.
	Element Type: Interface Owned Operations: Request-ThreatIntelligenceData: This SSG interface must provide features that apply the appropriate messaging and network protocols and issue IEF component data requests to the user-specified Threat Intelligence System. Receive-ThreatIntelligenceData: The SSG interface must provide features that listen for threat-risk data messages from the user-specified Threat-risk

Table 47 - SSG Configurations Elements	
Element Name	**Element and Operation Descriptions**
	intelligence system, receive the data message, extract the information needed by the IEF component, and pass the data to the SSG for processing.
Trustmark_Registry	The SSG may provide an interface that enables IEF components to interoperate with specified Trustmark registries and services that provision federated third-party identity, access, and policy information. IEF components requiring third-party identity, access, and policy information get that information by communicating over the SMB to the SSG.
	Element Type: Interface Owned Operations: Request_UserTrustmarkData: This SSG interface must provide features that apply the appropriate messaging and network protocols and issue IEF component data requests to the user-specified Trustmark Repository. As a TrustMark Relying Party, the SSG Interface provides features that apply the appropriate messaging and network protocols and issue a request to the specified TrustMark Provider for the TrustMark Interoperability Profile(s). For more information on Trusmarks, go to https://trustmark.gtri.gatech.edu/. Receive_UserTrustmarkData: The SSG interface must provide features that listen for Trustmark data messages from the user-specified Threat-risk intelligence system, receive the data message, extract the information needed by the IEF component, and pass the data to the SSG for processing. The SSG Interface provides features that listen for a message from the TrustMark Provider (registry), receive the Trust Interoperability Profile(s), extract the information needed by the requesting IEF component, and pass the information to the SSG for processing.

`

13 Cryptographic Transformation Service (CTS)

When an information asset becomes subject to IEF protection, the information is encrypted at rest and in transit (when received and released). The Cryptographic Transformation Service (CTS) integrates cryptographic methods and algorithms into an IEF configuration using internal (encryption/decryption) or external (user-specified) services.

13.1 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) are encrypted at rest and in transit and decrypted for use within an authorized client- service;
- Email messages, including attachments, are encrypted when they are stored/prepared for delivery and decrypted when a user accesses the e-mail using an authorized e-mail client.
- IM/Chat messages are encrypted as an IM server exchanges them. IM messages are decrypted when viewed through an authorized IM -client.
- Structured payloads (messages, including attachments) are encrypted for storage and transmission and decrypted for use within an authorized client service.

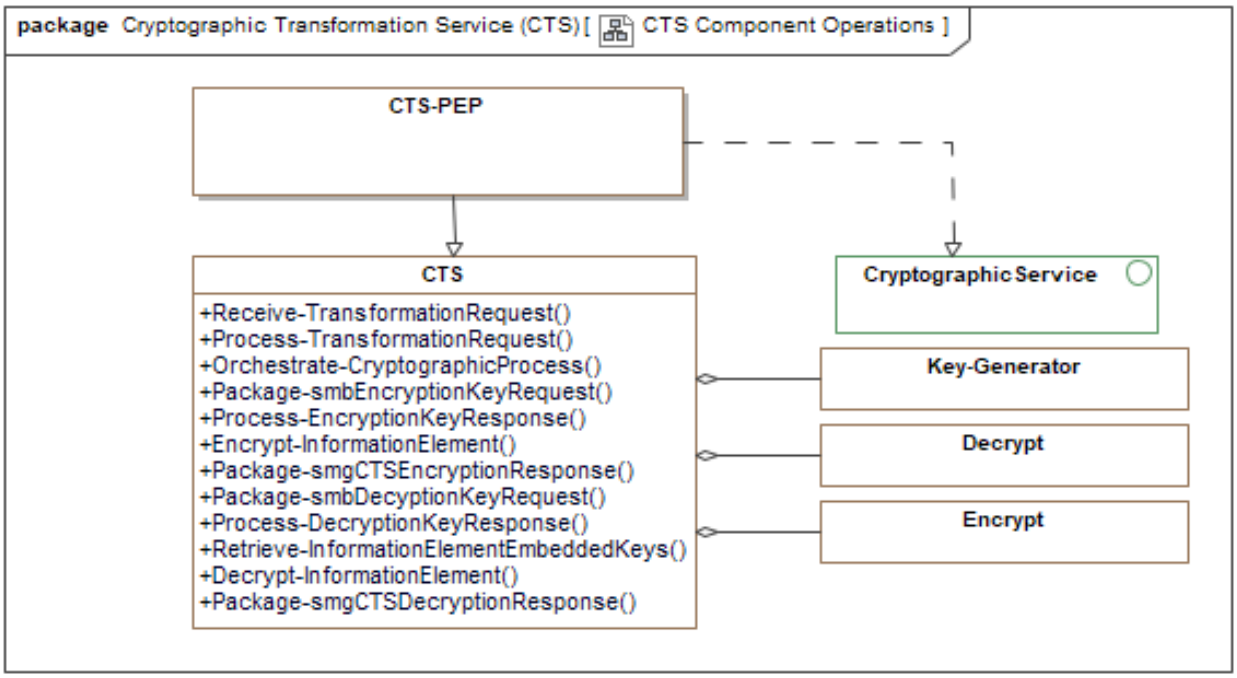


Figure 83 -CTS Component Operations

The following table identifies and describes the elements and operations illustrated in the previous figure - CTS Component Operations.

Table 48 - CTS Component Operations Elements	
Element Name	**Element and Operation Descriptions**
CTS	The Cryptographic Transformation Service (CTS) provides a common interface to cryptographic (encryption and decryption) services required by IEF components. The CTS may utilize internal encryption or decryption services or provide an interface to a user-specified system or services. All encryption services are expected to be FIPS-complaint software modules, services, or appliances.
	Element Type: Class Owned Operations: Receive-TransformationRequest: The CTS must provide features that receive transformation (/encryption/decryption) requests from the SMB. Process-TransformationRequest: The CTS must provide features that parse and map the elements of a transformation request. Orchestrate-CryptographicProcess: The CTS must provide features that stage the encryption or decryption processes using internal services or user-specified systems or appliances. Package-smbEncryptionKeyRequest: The CTS must provide features that package an SMB message to SSG requesting encryption keys from the user-specified KeyManagement system or service. Process-EncryptionKeyResponse: The CTS must provide features that parse and map the encryption key response message from the SSG via the SMB. Encrypt-InformationElement: The Cryptographic service must provide features that transform an information element into an unintelligible form using internal or user-specified FIPS-compliant methods, algorithms, services, systems, or appliances. Package-smgCTSEncryptionResponse: The CTS must provide features that aggregate and format a CTS response message to be issued to the requesting IEF component via the SMB. Package-smbDecyptionKeyRequest: The CTS must provide features that package an SMB message to SSG requesting encryption keys from the user-specified KeyManagement system or service. Process-DecryptionKeyResponse:

`

Table 48 - CTS Component Operations Elements	
Element Name	**Element and Operation Descriptions**
	The CTS must provide features that parse and map the encryption key response message from the SSG via the SMB. Retrieve-InformationElementEmbeddedKeys: The CTS must provide features that enable the retrieval of key information from the information element binding manifest. Decrypt-InformationElement: The CTS must provide 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. Package-smgCTSDecryptionResponse: The CTS must provide features that aggregate and format a CTS response message to be issued to the requesting IEF component via the SMB. Inherited Operations: The above functions utilize the following inherited operations to deliver their features. Inherited functions include: inherited from IEF_Component 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
CTS-PEP	This CTS-PEP connects the IEF Services to user-specified cryptographic services through a ZTA-enabled interface. The CTS operates in the same fashion as the SSG, and it is tailored to cryptographic operations.

Table 48 - CTS Component Operations Elements	
Element Name	**Element and Operation Descriptions**
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration. Inherited Operations: The above functions utilize the following inherited operations to deliver their features. Inherited functions include: Manage-PEPOperations inherited from PEP Execute-AdministrationFunctions inherited from PEP Gather-UserAttributeData inherited from PEP Gather-RecipientLocation inherited from PEP Gather-IdentityData inherited from PEP Generate-DecisionRequest inherited from PEP Enforce-AuthorizationDecisions inherited from PEP Store-PEPConfiguration inherited from PEP Restore-PEPConfiguration inherited from PEP Publish-PEPConfiguration inherited from PEP Load-PEPconfiguration inherited from PEP Generate_ProcessingUUID inherited from PEP inherited from IEF_Component 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 inherited from CTS inherited from CTS inherited from CTS

`

Table 48 - CTS Component Operations Elements	
Element Name	**Element and Operation Descriptions**
	Receive-TransformationRequest inherited from CTS Process-TransformationRequest inherited from CTS Orchestrate-CryptographicProcess inherited from CTS Package-smbEncryptionKeyRequest inherited from CTS Process-EncryptionKeyResponse inherited from CTS Encrypt-InformationElement inherited from CTS Package-smgCTSEncryptionResponse inherited from CTS Package-smbDecyptionKeyRequest inherited from CTS Process-DecryptionKeyResponse inherited from CTS Retrieve-InformationElementEmbeddedKeys inherited from CTS Decrypt-InformationElement inherited from CTS Package-smgCTSDecryptionResponse inherited from CTS
Decrypt	The PPS may include the FIPS-complaint methods and algorithms needed to decrypt an information or data element.
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
Encrypt	The PPS may include the FIPS-complaint methods and algorithms needed to encrypt an information or data element.
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
Key-Generator	A key generation service that will create the needed unique cryptographic keys for each element being encrypted.
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.

13.2 CTS Configurations

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, are 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 an IM server exchanges them and
- 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 will return 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.

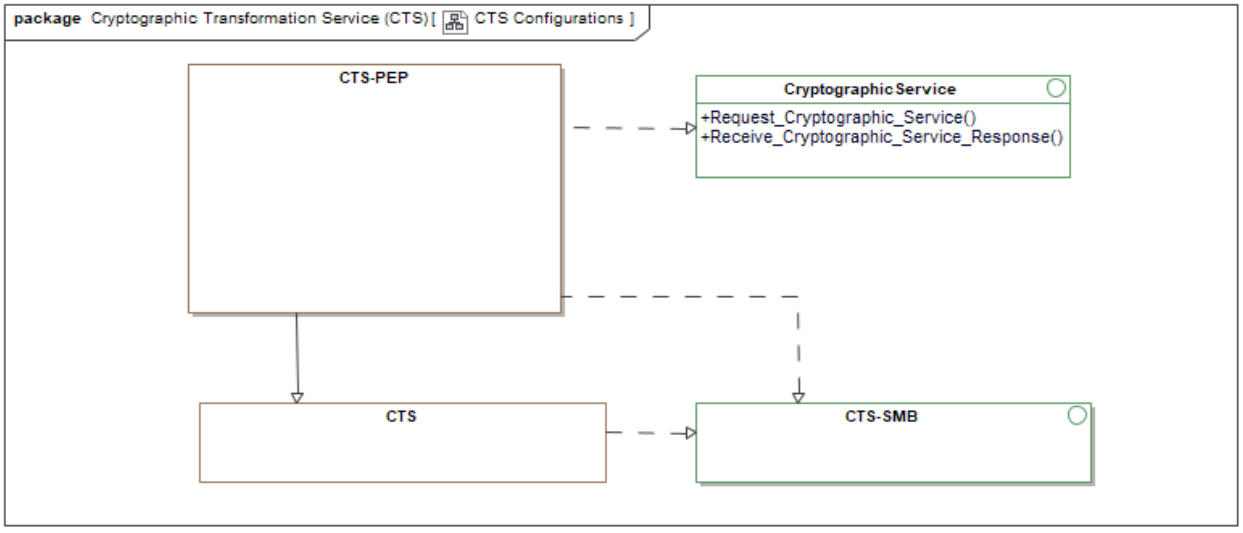


Figure 84 -CTS Configurations

The following table identifies and describes the elements and operations illustrated in the previous figure - CTS Configurations.

`

Table 49 - CTS Configurations Elements	
Element Name	**Element and Operation Descriptions**
CryptographicService	The cryptographic service interface provides features that enable the PEP or CTS to interoperate with user-specified cryptographic services. The requirements and functions provided by this interface are left to the user (/implementer/integrator) to define based on the capabilities and API of the specified cryptographic system.
	Element Type: Interface Owned Operations: Request_Cryptographic_Service: This SSG interface must provide features that apply the appropriate messaging and network protocols and issue a request for identity information to the user-specified cryptographic services. Receive_Cryptographic_Service_Response: The SSG interface provides features that listen for a message from the user-specified cryptographic Services, receive it, extract the data/information the requesting IEF component needs, and pass it to the SSG for processing.
CTS-SMB	The CTS-SMB interface enables the CTS to interoperate with other IEF components (e.g., PAP, PEP, PPS, SSG, PDP, TLS, and CTS) in the configuration. In a Zero-Trust implementation, it is responsible for enforcing access and release control policy between IEF components. Messages (Clause 16) applicable to the CTS-SMB interface include: • PAP-Command (receive); • PAP-Command-response (send); • CTS-Request (receive); • CTS-Response (send); and • CTS-LogMessage (send).
	Element Type: Interface This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.

14 Trusted Logging Service (TLS)

The Trusted Logging Service (TLS) is the policy enforcement point that persists event IEF component (/transaction) reports to enable runtime security incidents, event monitoring, reporting, and forensic auditing. The TLS provides a standard interface on the SMB for PAP, PDP, PEP, PPS, SSG, and SMB to persist their log reports.

A configuration parameter within the IES configuration sets the logging level to be performed based on the user-defined resource utilization and performance specifications. The logging level may be set for each component or the environment.

14.1 TLS Component Operations

The following figure identifies the features and functions provided by the TLS.

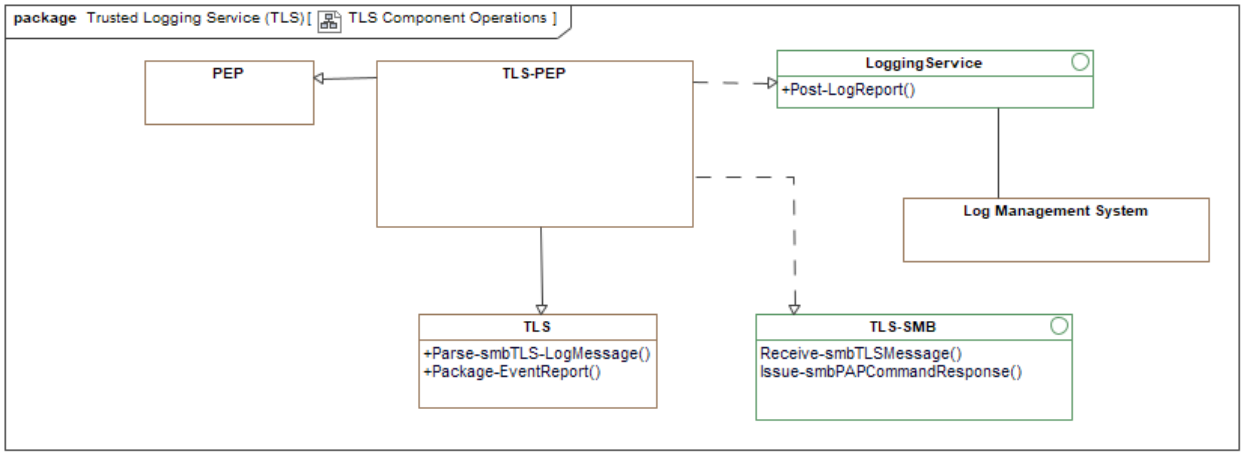


Figure 85 -TLS Component Operations

The following table identifies and describes the elements and operations illustrated in the previous figure - TLS Component Operations.

Table 50 - TLS Component Operations Elements	
Element Name	**Element and Operation Descriptions**
Log Management System	A log management system provides continuous gathering, storing, processing, synthesizing, and analyzing transaction and event log data from disparate programs and applications to optimize system performance, identify security and technical issues, better manage resources, and improve regulatory and policy compliance. Log management systems deliver features that address the following: 1. Log data Collection: Services that aggregate data from the OS, applications, DCS services, servers, users, endpoints, or other relevant sources. 2. Operational and Security Monitoring: Services that track security and operational transactions, events, and activities and present them in a human-understandable form.

`

Table 50 - TLS Component Operations Elements	
Element Name	**Element and Operation Descriptions**
	3. Analysis: Services that review and analyze log data collections and proactively identify bugs, security threats, or other issues. 4. Retention: Services that ensure log data or summaries are retained according to policy. 5. Indexing or Search: Tools that filter, sort, analyze, or search data across all log files. 6. Reporting: Services that automate visualization, reporting, and auditing of log data as it relates to operational performance, resource allocation, security, or regulatory compliance. 7. Security and Integrity: Services that protect log data from unauthorized access, use, manipulation, and appropriation.
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration.
PEP	The Policy Enforcement Point (PEP) intercepts all user and system requests to access or release data elements protected by IEF services. The PEP gathers information about the data elements, sender, recipients, and communication channel, packages an adjudication request to the PDP, and enforces the PDP access or release determination. In a zero-trust environment, the PEP interface authenticates and authorizes each transaction with an external cryptographic service. The PEP must provide one or both of the feature sets defined by the SMB or Direct Connect options.
	Element Type: Class Owned Operations: Manage-PEPOperations: The PEP must provide features that manage the execution of PEP functions in each potential configuration (e.g., Direct connect, SMB connection, or a combination of direct and SMB configurations). Execute-AdministrationFunctions: The PEP must provide features that execute AdministrativeCommands from an authorized Policy Administration Point. PEP administrative functions include: 1. Activate PEP features; 2. Deactivate PEP features;

Table 50 - TLS Component Operations Elements	
Element Name	**Element and Operation Descriptions**
	3. Configure PEP Parameters; 4. Archive PEP Operational Environment; 5. Publish PEP Configuration; 6. Store PEP Configuration; and 7. Retrieve PEP Configuration; Gather-UserAttributeData: The PEP must provide features that gather the sender and receiver attributes to determine their authorization to receive data elements. Gather-RecipientLocation: (Optional) The PEP provides features that gather information about the recipients' location (physical or electronic). A recipient's location(s) (e.g., network location, physical location, and device) may impact the user's attributes and the information content they are authorized to receive. These features may only be available if the IEF can request the information from the user's situational awareness, incident management, or network management systems. Gather-IdentityData: The PEP must provide features that gather the identity information for the sender and recipients of the specified information element (s). These features allow users to request this information from the infrastructure services that provide identity management. All requests to the users' specified infrastructure are issued through the Security Services Gateway using an SSG-Request message. Generate-DecisionRequest: The PEP must provide features that generate a decision request to the PDP. Enforce-AuthorizationDecisions: The PEP Provides features that enforce PDP authorization decisions for each data receipt and release. Store-PEPConfiguration: The PEP Provides features that gather and store its configuration parameters in local storage. Restore-PEPConfiguration: The PEP Provides features that retrieve and load its configuration parameters from local storage. Publish-PEPConfiguration: The PEP must provide features that gather and publish its configuration parameters to the PAP. Load-PEPconfiguration:

`

Table 50 - TLS Component Operations Elements	
Element Name	**Element and Operation Descriptions**
	The PEP must provide features that receive and load its configuration parameters from the PAP. Generate_ProcessingUUID: Upon receiving a message, the PEP must provide features that generate a universally unique identifier (UUID). The structure of the UUID must identify the node receiving the data. If the received message includes a UUID, the PEP must adopt the received UUID. Inherited Operations: The above functions utilize the following inherited operations to deliver their features. Inherited functions include: inherited from IEF_Component 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
TLS	The Trusted Logging Service (TLS) provides logging services for all IEF components. In this configuration, the TLS provides logging as a native service within the IEF configuration.
	Element Type: Class Owned Operations: Parse-smbTLS-LogMessage: The TLS provides features that parse an IEF Component Log Report. Package-EventReport: The TLS provides features that aggregate and format event or transaction records to be issued to the user-specified log management system.

Table 50 - TLS Component Operations Elements	
Element Name	**Element and Operation Descriptions**
	Inherited Operations: The above functions utilize the following inherited operations to deliver their features. Inherited functions include: inherited from IEF_Component 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
TLS-PEP	This PEP (/proxy) connects the IEF Services to logging services provided by the user or the local infrastructure (e.g., Cloud Service Provider (CSP)). The TLS-PEP is the integration point between IEF components and external logging services. (Clause 15)
	Element Type: Class This element provides the user, implementor, or integrator with an extension point to the specification where the IEF is tailored for a specific product or service configuration. Inherited Operations: The above functions utilize the following inherited operations to deliver their features. Inherited functions include: Manage-PEPOperations inherited from PEP Execute-AdministrationFunctions inherited from PEP Gather-UserAttributeData inherited from PEP Gather-RecipientLocation inherited from PEP Gather-IdentityData inherited from PEP Generate-DecisionRequest inherited from PEP Enforce-AuthorizationDecisions inherited from PEP Store-PEPConfiguration inherited from PEP Restore-PEPConfiguration inherited from PEP Publish-PEPConfiguration inherited from PEP

`

Table 50 - TLS Component Operations Elements	
Element Name	**Element and Operation Descriptions**
	Load-PEPconfiguration inherited from PEP Generate_ProcessingUUID inherited from PEP inherited from IEF_Component 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 Parse-smbTLS-LogMessage inherited from TLS Package-EventReport inherited from TLS

14.2 TLS Configurations

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

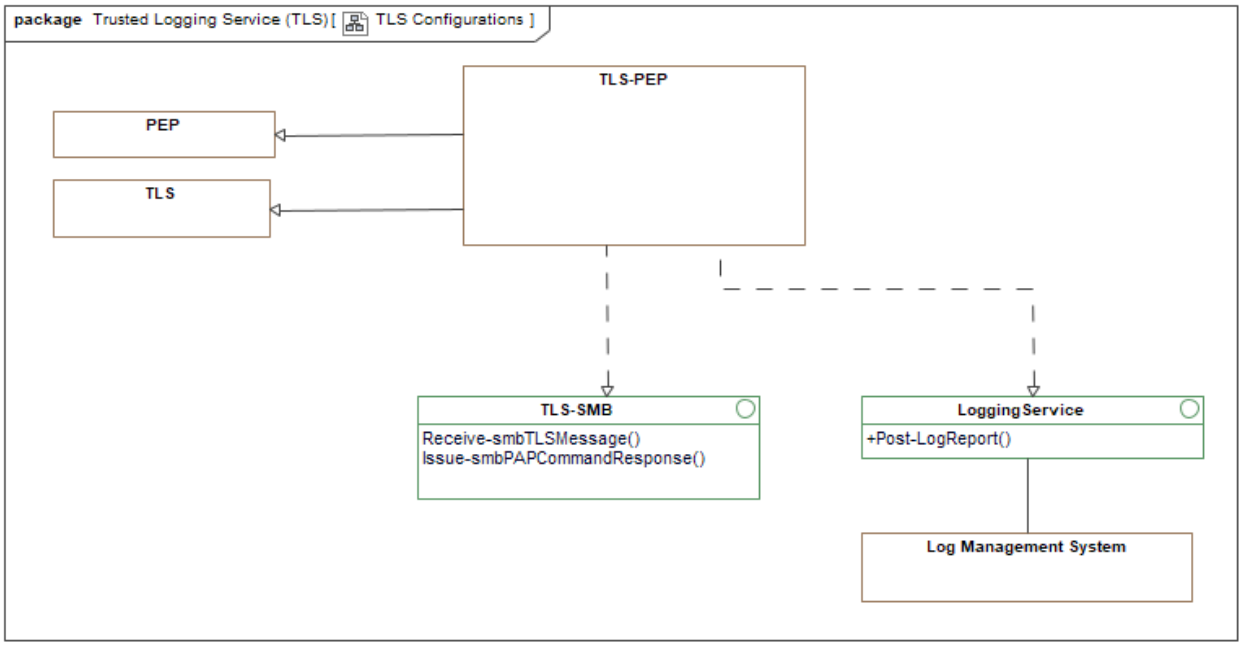


Figure 86 -TLS Configurations

The following table identifies and describes the elements and operations illustrated in the previous figure - TLS Configurations.

Table 51 - TLS Configurations Elements	
Element Name	**Element and Operation Descriptions**
LoggingService	This TLS interface (see notes below) must provide features that enable the PEP or TLS to interoperate with the user-specified log management service (/system). The requirements and functions provided by this interface are left to the user (/implementer/integrator) to define based on the capabilities and API of the specified logging system.
	Element Type: Interface Owned Operations: Post-LogReport: The TLS interface must provide features that post PEP Log Messages to the user-specified logging service according to the log management system API.
TLS-SMB	SMB interface enables the TLS to interoperate with other IEF components (e.g., PEP, SSG, PDP, PPS, and CTS) in the configuration. In a Zero-Trust implementation, it is responsible for enforcing access and release control policy between IEF components. Messages (Clause 16) applicable to the TLS-SMB interface include: • TLS-LogMessage (receive); • PAP-Command (receive); and

`

Table 51 - TLS Configurations Elements	
Element Name	**Element and Operation Descriptions**
	• PAP-CommandResponse (send).
	Element Type: Interface Owned Operations: Receive-smbTLSMessage: This TLS interface must provide features that listen for a message from the IEF component over the SMB, receive the message, extract the information needed by the requesting IEF component, and pass it to the TLS for processing. Issue-smbPAPCommandResponse: This TLS interface must provide features that bind the messaging and network protocols and issue the message to the PAP.

15 Secure Messaging Bus (SMB)

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

The information exchanges between services utilize industry-accepted, open standards based on XML. It is the responsibility of the IEF Secure Messaging Bus (SMB) to deliver these messages between IEF components securely. 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 SMB is responsible 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.

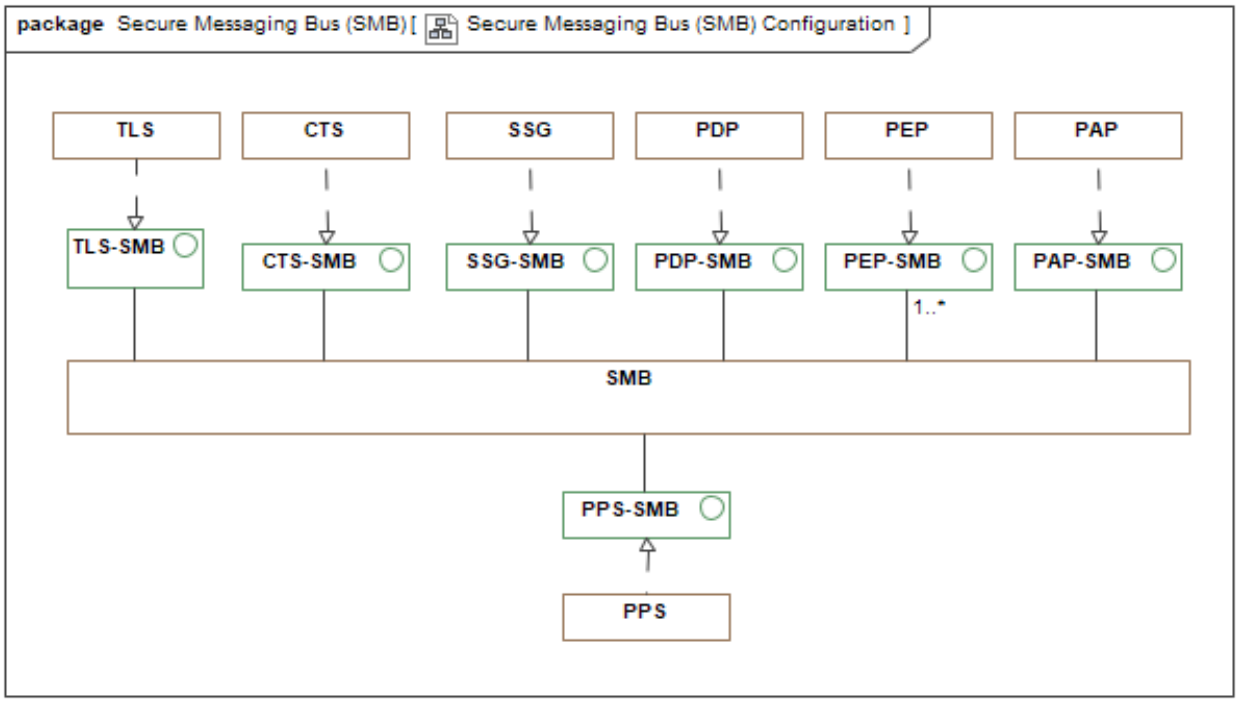
Technology demonstration projects (TDP) have been conducted using two different standards-based messaging solutions:

- XMPP network integrates IEF components that deliver IEF services (components) for file-share, email, and instant (Text) messaging. Defence Research and Development Canada conducted this TDP.
- 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-driven data-centric data (/information) and safeguarding architecture.

15.1 Secure Messaging Bus (SMB) Configuration

The Secure Messaging Bus (SMB) represents a data exchange or middleware solution that securely exchanges data messages (e.g., encrypted messages) between the IEF components using secure methods and protocols (e.g., DDS with security extensions). The user or implementor selects their preferred SMB infrastructure. The following figure illustrates the IEF connections to the SMB. These elements are described in previous clauses.



`

Figure 87 -Secure Messaging Bus (SMB) Configuration

16 SMB Data Structures (/Messages)

The following sections define the structures for the messages exchanges between IEF Components.

16.1 Secure Messaging Bus (SMB)

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

The information exchanges between services utilize industry-accepted, open standards based on XML. It is the responsibility of the IEF Secure Messaging Bus (SMB) to deliver these messages between IEF components securely. 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 SMB is responsible 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 integrates IEF components that deliver IEF services (components) for file-share, email, and instant (Text) messaging. Defence Research and Development Canada conducted this TDP.
- 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-driven data-centric data (/information) and safeguarding architecture.

16.1.1 SMB Message

The following figure and table identify the minimum set of messages and metadata exchanged between IEF components during operations. The message's prefix identifies the primary actor in the message exchange.

`

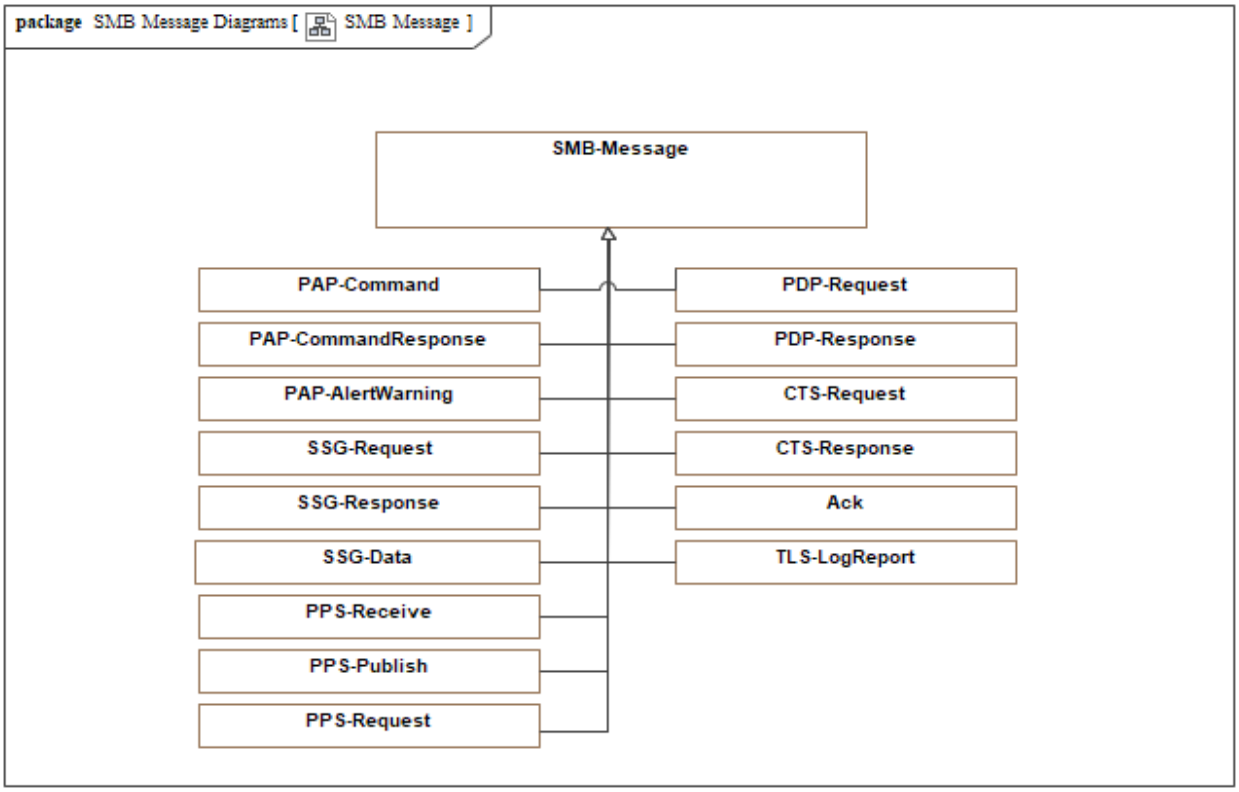


Figure 88 -SMB Message

The following table describes the message elements illustrated in the previous figure - SMB Message.

Table 52 - SMB Message Classes	
Element Name	**Attributes**
Ack	The following figure identifies the data and information elements sent to the originating component to acknowledge the receipt of a message.
CTS-Request	A message to the CTS that requests the transformation (encryption or decryption) of the specified information element.
CTS-Response	A message from a CTS to an IEF component responding to a cryptographic transformation request.
PAP-AlertWarning	A message from an IEF Component to the PAP (and the TLS) informs the user (/administrator) of an unauthorized request to access or release information, an unauthorized request to operate, or other error conditions generated by these requests.
PAP-Command	A message from the PAP to an IEF component directing the component to perform a specified operation or action.

Table 52 - SMB Message Classes	
Element Name	**Attributes**
PAP-CommandResponse	A message from an IEF component to the PAP provides the results of a PAP's command message.
PDP-Request	A message from an IEF component to a PDP requesting authorization to perform a specific action on specified information element(s) targeting a specified set of recipients.
PDP-Response	A PDP message to a PEP providing the result(s) of its policy adjudication.
PPS-Publish	A message from a PPS to a structure messaging PEP directing that PEP to validate and verify that the PPS is authorized to issue the information element(s) and that the specified recipient(s) are authorized to receive and access the enclosed content. The message also directs the communication channels and protocols to be applied.
PPS-Receive	A message from a structured messaging PEP to a PPS forwarding authorized data and information elements from an authorized messaging service or middleware.
PPS-Request	A message from a PEP to the PPS to request the release of information based on a specified semantic element. The request contains the base (unique) identifier for the requested information elements and the identifier for semantic policies to be applied.
SMB-Message	Minimum set of data attributes passed between IEF components across the Secure Messaging Bus (SMB).
SSG-Data	A message from an IEF component to a user-specified security system, e.g., SIEM, CDM, SOAR.
SSG-Request	A message issued by an IEF component to the SSG 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.
SSG-Response	A message from the SSG to the IEF Component providing the requested data or Information elements.
TLS-LogReport	A 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

`

Table 52 - SMB Message Classes	
Element Name	**Attributes**
	record is encrypted in motion and storage and assigned a chained digital signature.

16.1.2 SMB Message Attributes

This clause identifies and describes the core attributes included in all SMB messages.

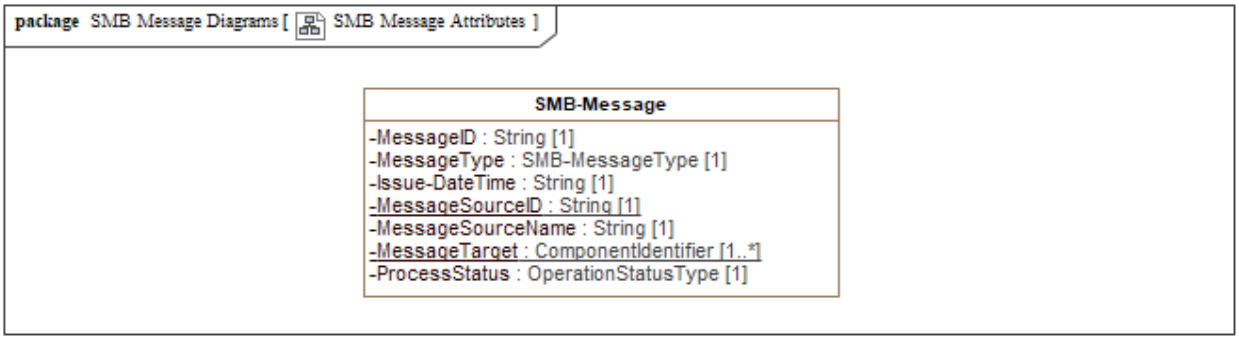


Figure 89 -SMB Message Attributes

The following table describes the message elements illustrated in the previous figure - SMB Message Attributes.

Table 53 - SMB Message Attributes Classes	
Element Name	**Attributes**
SMB-Message	Minimum set of data attributes passed between IEF components across the Secure Messaging Bus (SMB). Issue-DateTime (type: String) [1..1]: Identifies when the Message was issued. MessageID (type: String) [1..1]: Unique identifier for the message. MessageSourceID (type: String) [1..1]: Unique identifier for the component issuing the message. MessageSourceName (type: String) [1..1]: Name of the component issuing the message. MessageTarget (type: ComponentIdentifier) [1..*]: Unique identifier for the component expected to receive the message.

Table 53 - SMB Message Attributes Classes	
Element Name	**Attributes**
	MessageType (type: SMB-MessageType) [1..1]: Identifies the type of message that will enable the receiving component to initiate the appropriate processing sequence. ProcessStatus (type: OperationStatusType) [1..1]: Returns the operational status of the component.

16.1.3 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.

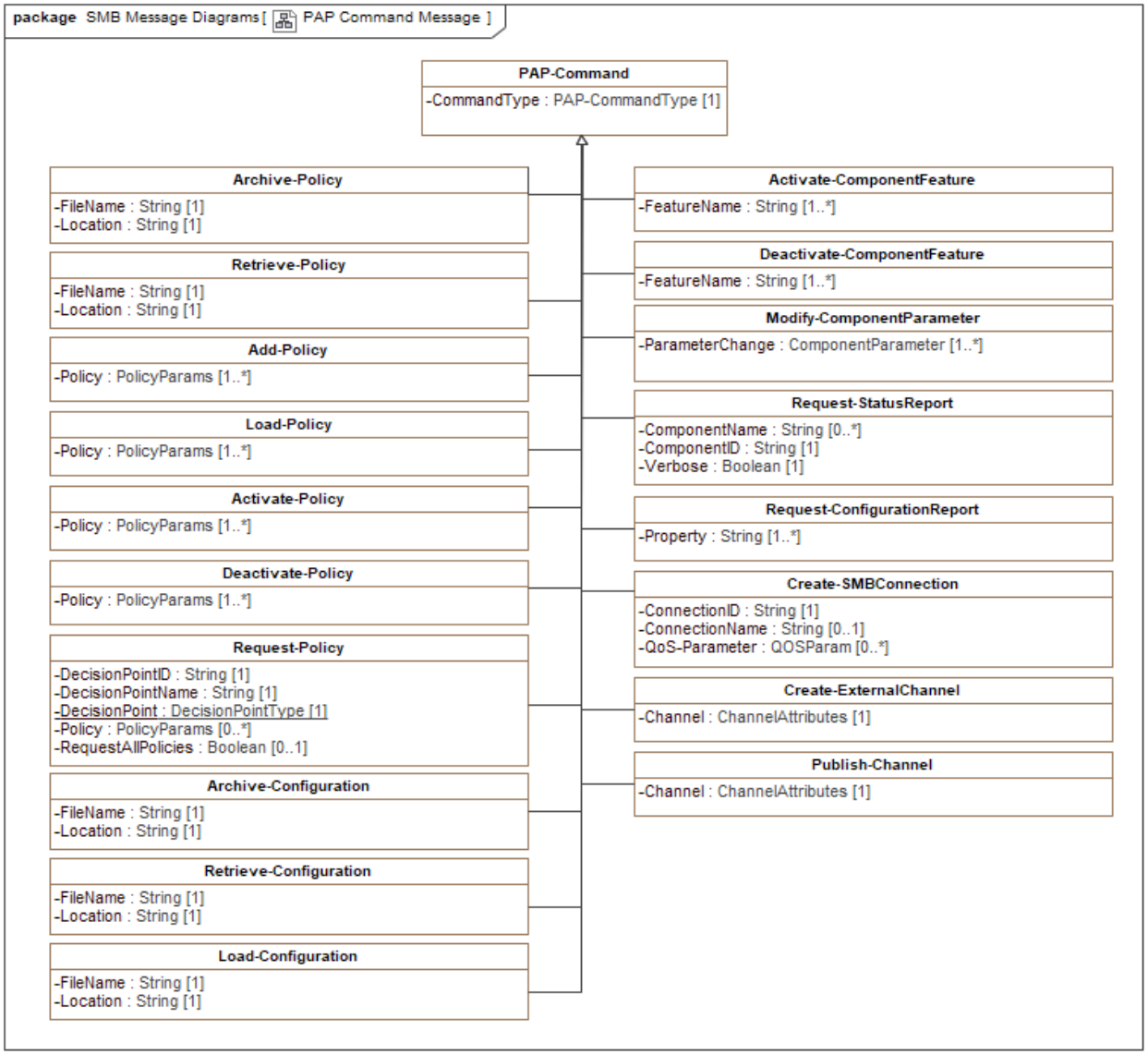


Figure 90 -PAP Command Message

`

The following table describes the message elements illustrated in the previous figure - PAP Command Message.

Table 54 - PAP Command Message Classes	
Element Name	**Attributes**
Activate-ComponentFeature	A message from a PAP that directs an IEF Component to activate one or more of its features. FeatureName (type: String) [1..*]: Name of the feature to be Activated. "All" directs the component to activate itself and all internal features.
Activate-Policy	A message from a PAP that directs a PPS or PDP to change the state of a specified set of policies in its environment from inactive to active. Policy (type: PolicyParams) [1..*]: Reference to policy parameters or attributes.
Add-Policy	A message from a PAP that directs a PPS or PDP to add the policies in the message to its policy environment. These policies are held in a temporary processing area until the user (/administrator) directs the component to load the policies. Policy (type: PolicyParams) [1..*]: Reference to policy parameters or attributes.
Archive-Configuration	A message from a PAP that directs an IEF component to persist its current operating configuration to a specified location. FileName (type: String) [1..1]: Name of the file that will contain the archived configuration. Location (type: String) [1..1]: Location of a file within the IEF persistent store.
Archive-Policy	A message from a PAP that directs a PPS or PDP to package (aggregate and format) its current policy environment and post the data as a file to the environment archive. FileName (type: String) [1..1]: Name of the file that will contain the archived Policy. Location (type: String) [1..1]: Location of a file within the IEF persistent store.
Create-ExternalChannel	A message from a PAP directs a Messaging-PEP to create a connection with the user-specified messaging infrastructure. Channel (type: ChannelAttributes) [1..1]: Unique identifier for the information exchange connection.

Table 54 - PAP Command Message Classes	
Element Name	**Attributes**
Create-SMBConnection	A message from a PAP that directs an IEF component to establish and register a connection to the SMB. ConnectionID (type: String) [1..1]: Unique identifier for the information exchange connection on the SMB. ConnectionName (type: String) [0..1]: A human-readable name is used to connect the information on the SMB. QoS-Parameter (type: QOSParam) [0..*]: Quality of Service Parameter for the connection of the SMB. Provided as "ParameterName: ParameterValue."
Deactivate-ComponentFeature	A message from a PAP that directs an IEF Component to deactivate one or more component features. FeatureName (type: String) [1..*]: Name of the feature to be deactivated. "All" directs the component to deactivate itself and all internal features.
Deactivate-Policy	A message from a PAP that directs a PPS or PDP to change the state of a specified set of policies in its environment from active to inactive. Policy (type: PolicyParams) [1..*]: Reference to policy parameters or attributes.
Load-Configuration	A message from a PAP that directs an IEF component to load a set of operating parameters and hold them in its processing area to reset its operational settings. 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.
Load-Policy	A message from a PAP that directs a PDP or a PPS to load one or more policies from its processing area to its policy environment. Policy (type: PolicyParams) [1..*]: Reference to policy parameters or attributes.
Modify-ComponentParameter	A message from a PAP that directs an IEF component to change the value of one or more of its configuration parameters. ParameterChange (type: ComponentParameter) [1..*]: Reference to a set of component parameters.
PAP-Command	A message from the PAP to an IEF component directing the component to perform a specified operation or action. CommandType (type: PAP-CommandType) [1..1]: Type of command being issued.

`

Table 54 - PAP Command Message Classes	
Element Name	**Attributes**
Publish-Channel	A message from a PAP to publish information on one or more channels the component uses to share data. Channel (type: ChannelAttributes) [1..1]: Reference to a set of channel parameters.
Request-ConfigurationReport	A message from a PAP that directs an IEF component to report its configuration. Property (type: String) [1..*]: Name of the parameter to be reported. "ALL" reports all component configuration parameters.
Request-Policy	A message from a PAP that directs a PPS or PDP to provide information on one or more policies. DecisionPoint (type: DecisionPointType) [1..1]: Identifies the type of Decision Point. DecisionPointID (type: String) [1..1]: Unique Identifier of the decision point. DecisionPointName (type: String) [1..1]: Name of the decision point. Policy (type: PolicyParams) [0..*]: Unique identifier for the policy or a set of policies to be reported on. RequestAllPolicies (type: Boolean) [0..1]: Request all policies within the decision point memory.
Request-StatusReport	A 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. ComponentID (type: String) [1..1]: Unique identifier of the component. ComponentName (type: String) [0..*]: Name of the component (/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, while the basic report provides an overall status for the component.
Retrieve-Configuration	A message from a PAP that directs an IEF component to get a configuration file (or archive) from a specified location in the IEF environment. The component will extract the configuration file from

Table 54 - PAP Command Message Classes	
Element Name	**Attributes**
	the SAC, ingest the configuration policies environment, and wait for the instructions to activate the configuration. FileName (type: String) [1..1]: Name of the file that will contain the archived configuration. Location (type: String) [1..1]: Location of a file within the IEF persistent store.
Retrieve-Policy	A message from a PAP that directs a PPS or PDP to get one or more policies from a specified file. The PPS or PDP parses and maps the file's content to integrate the policies and waits for instructions to activate the new policies. 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.

16.1.4 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.

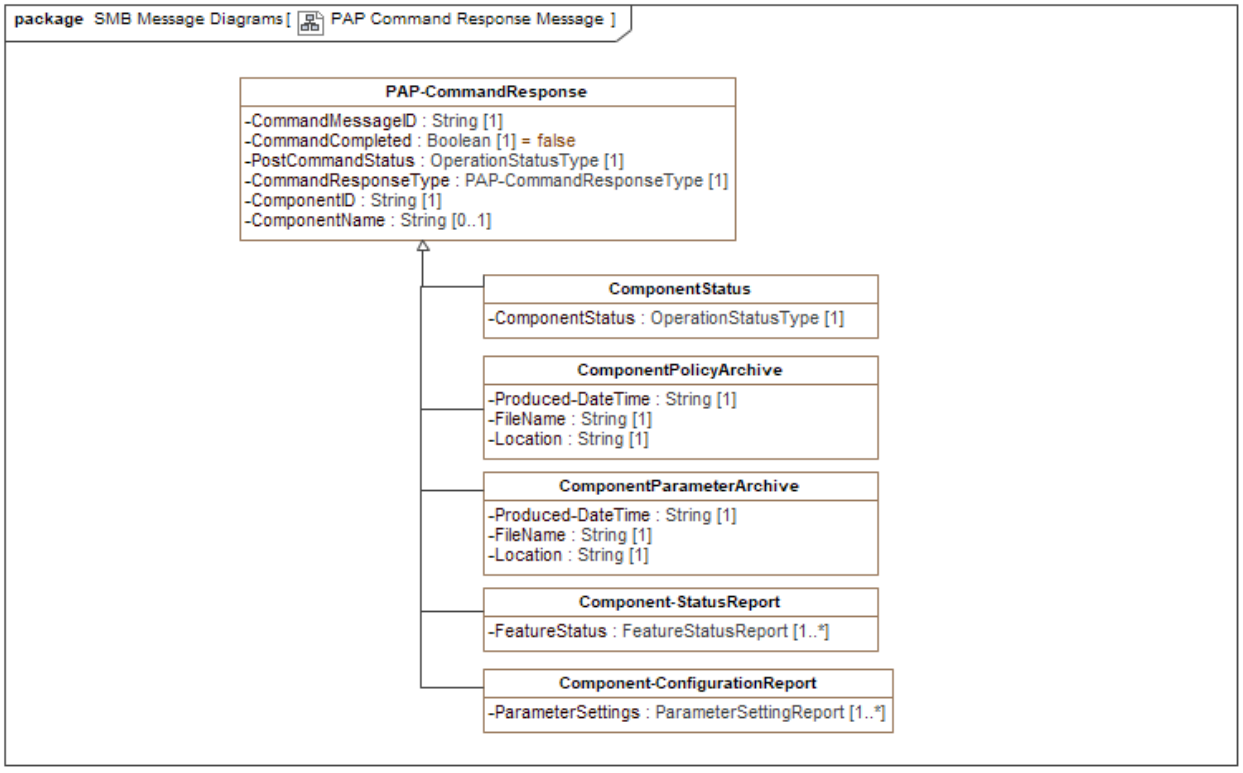


Figure 91 -PAP Command Response Message

`

The following table describes the message elements illustrated in the previous figure - PAP Command Response Message.

Table 55 - PAP Command Response Message Classes	
Element Name	**Attributes**
Component-ConfigurationReport	A message from an IEF Component that contains the IEF Component configuration for the **PAP**. ParameterSettings (type: ParameterSettingReport) [1..*]: An array (/list/map) of Parameter settings.
Component-StatusReport	A message from an IEF component to the TLS containing a report on the component's current status for each of its features. FeatureStatus (type: FeatureStatusReport) [1..*]: An array of features and their status values.
ComponentParameterArchive	A message from an IEF Component that contains the name and location of the file containing the current component configuration parameters. 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 are the parameters set or generated.
ComponentPolicyArchive	A message from 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. 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.
ComponentStatus	A message from an IEF component to the PAP that contains the overall status of the component. ComponentStatus (type: OperationStatusType) [1..1]: The overall status of the component.

Table 55 - PAP Command Response Message Classes	
Element Name	**Attributes**
PAP-CommandResponse	A message from an IEF component to the PAP provides the results of a PAP's command message. CommandCompleted (type: Boolean) [1..1]: Indicates whether or not the command executed correctly. CommandMessageID (type: String) [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: String) [1..1]: The identifier of the responding IEF component. ComponentName (type: String) [0..1]: A 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.

16.1.5 PAP Alert Warning 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.

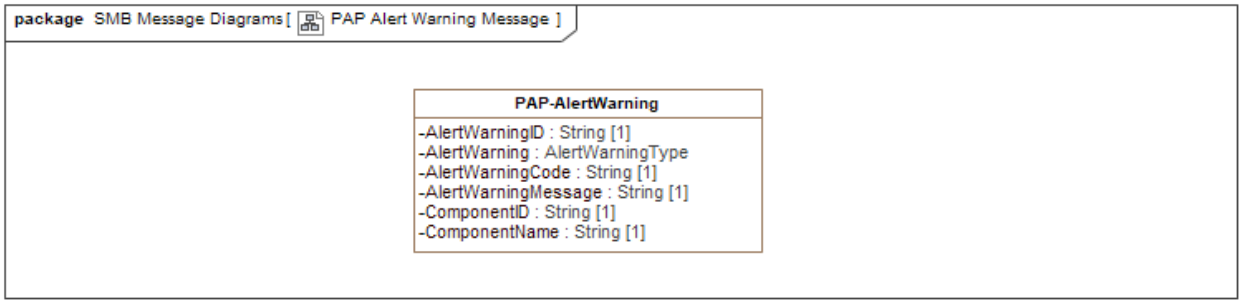


Figure 92 -PAP Alert Warning Message

The following table describes the message elements illustrated in the previous figure - PAP Alert Warning Message.

`

Table 56 - PAP Alert Warning Message Classes	
Element Name	**Attributes**
PAP-AlertWarning	A message from an IEF Component to the PAP (and the TLS) informs the user (/administrator) of an unauthorized request to access or release information, an unauthorized request to operate, or other error conditions generated by these requests. AlertWarning (type: AlertWarningType) [1..1]: Type of AlertWarning being sent. AlertWarningCode (type: String) [1..1]: Unique identifier for the Alert Warning Message. AlertWarningID (type: String) [1..1]: Unique identifier for the AlertWarning. AlertWarningMessage (type: String) [1..1]: Human readable text for the AlertWarning Message. ComponentID (type: String) [1..1]: Unique identifier for the component sending the AlertWarning Message. ComponentName (type: String) [1..1]: Name of the component sending the AlertWarning Message.

16.1.6 PDP Request Message

The following figure identifies the data and information elements issued by an IEF component requesting authorization to perform a specified function (/operation).

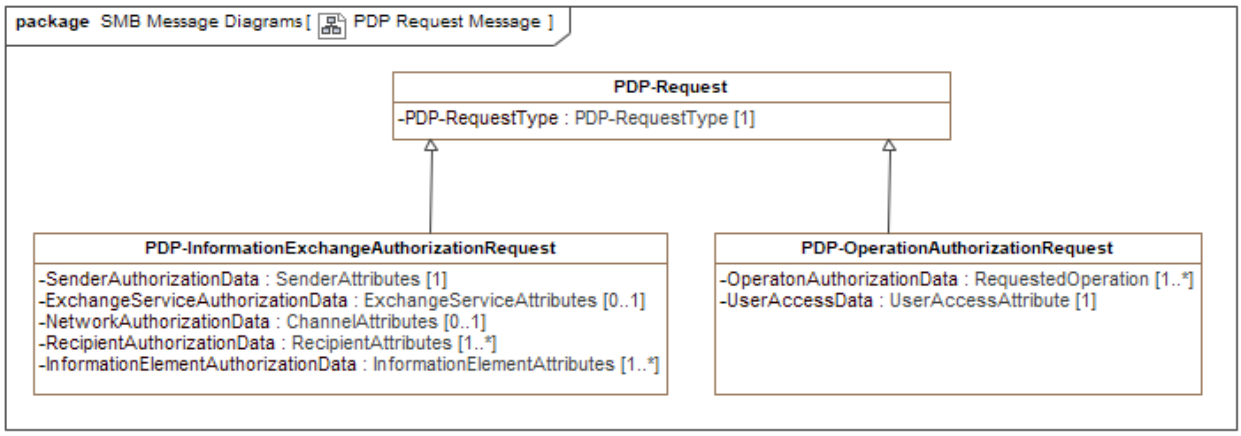


Figure 93 -PDP Request Message

The following table describes the message elements illustrated in the previous figure - PDP Request Message.

Table 57 - PDP Request Message Classes	
Element Name	**Attributes**
PDP-InformationExchangeAuthorizationRequest	A message from a PEP to a PDP requesting authorization to release specified information element(s) to a specified set of recipients. ExchangeServiceAuthorizationData (type: ExchangeServiceAttributes) [0..1]: Data used to authorize an exchange service. InformationElementAuthorizationData (type: InformationElementAttributes) [1..*]: Data used to authorize an information element. NetworkAuthorizationData (type: ChannelAttributes) [0..1]: Data used to authorize a network or a channel. RecipientAuthorizationData (type: RecipientAttributes) [1..*]: Data used to authorize a recipient. SenderAuthorizationData (type: SenderAttributes) [1..1]: Data used to authorize a sender. Inherited Attributes: Issue-DateTime is inherited from SMB-Message MessageID is inherited from SMB-Message MessageSourceID is inherited from SMB-Message MessageSourceName is inherited from SMB-Message MessageTarget is inherited from SMB-Message MessageType is inherited from SMB-Message PDP-RequestType is inherited from PDP-Request ProcessStatus is inherited from SMB-Message
PDP-OperationAuthorizationRequest	A message from a PAP to a PDP to request authorization for a user's (/administrator) request to execute a specified function or operation. OperatonAuthorizationData (type: RequestedOperation) [1..*]: Data used to authorize an operation. UserAccessData (type: UserAccessAttribute) [1..1]: Data used to authorize a user for a requested operation. Inherited Attributes: Issue-DateTime is inherited from SMB-Message MessageID is inherited from SMB-Message MessageSourceID is inherited from SMB-Message MessageSourceName is inherited from SMB-Message

`

Table 57 - PDP Request Message Classes	
Element Name	**Attributes**
	MessageTarget is inherited from SMB-Message MessageType is inherited from SMB-Message PDP-RequestType is inherited from PDP-Request ProcessStatus is inherited from SMB-Message
PDP-Request	A message from an IEF component to a PDP requesting authorization to perform a specific action on specified information element(s) targeting a specified set of recipients. PDP-RequestType (type: PDP-RequestType) [1..1]: Inherited Attributes: Issue-DateTime is inherited from SMB-Message MessageID is inherited from SMB-Message MessageSourceID is inherited from SMB-Message MessageSourceName is inherited from SMB-Message MessageTarget is inherited from SMB-Message MessageType is inherited from SMB-Message ProcessStatus is inherited from SMB-Message

16.1.7 PDP Response Message

The following figure identifies the data and information elements issued by the PDP to a PEP in response to an Information Exchange (IE) authorization request.

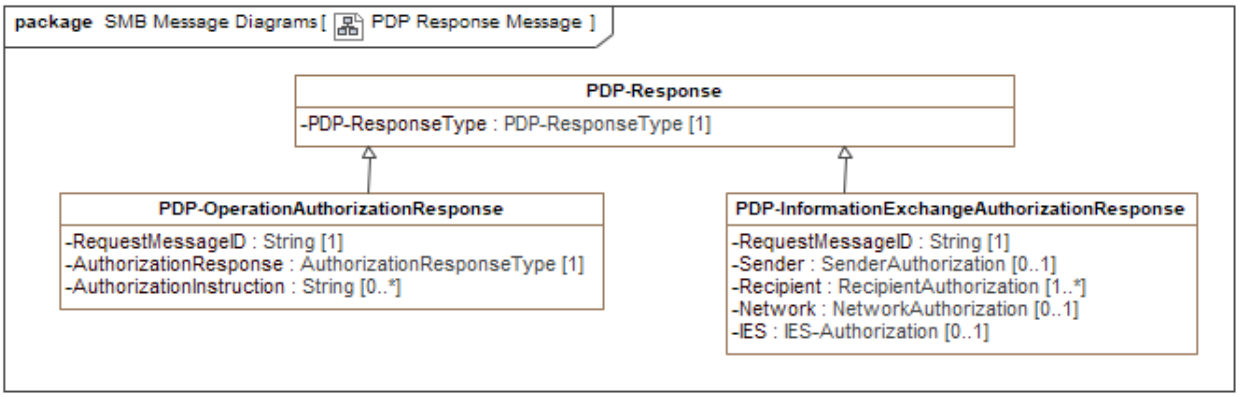


Figure 94 -PDP Response Message

The following table describes the message elements illustrated in the previous figure - PDP Response Message.

Table 58 - PDP Response Message Classes	
Element Name	**Attributes**
PDP-InformationExchangeAuthorizationResponse	A message from a PDP to a PEP providing the result(s) of its policy adjudication. IES (type: IES-Authorization) [0..1]: Information describing the attributes of the information exchange service being used. Network (type: NetworkAuthorization) [0..1]: Information describing the attributes of the network being used. Recipient (type: RecipientAuthorization) [1..*]: Information describing the attributes of the User receiving the information. RequestMessageID (type: String) [1..1]: The Identifier for PDP-AuthorizationRequest Message. Sender (type: SenderAuthorization) [0..1]: Information describing the attributes of the User sending the information to the channel, IES, or Network. Inherited Attributes: Issue-DateTime is inherited from SMB-Message MessageID is inherited from SMB-Message MessageSourceID is inherited from SMB-Message MessageSourceName is inherited from SMB-Message MessageTarget is inherited from SMB-Message MessageType is inherited from SMB-Message PDP-ResponseType is inherited from PDP-Response ProcessStatus is inherited from SMB-Message
PDP-OperationAuthorizationResponse	A PDP message to the PAP providing the result(s) of its policy adjudication of a user's request to perform a specified function or operation. AuthorizationInstruction (type: String) [0..*]: Instruction to be performed by the PAP to authorize the requested function or operation. AuthorizationResponse (type: AuthorizationResponseType) [1..1]: The result of the PDP's adjudication on the user's authorization to execute the function or operation. RequestMessageID (type: String) [1..1]: The Identifier for AuthorizationRequestMessage. Inherited Attributes: Issue-DateTime is inherited from SMB-Message

`

Table 58 - PDP Response Message Classes	
Element Name	**Attributes**
	MessageID is inherited from SMB-Message MessageSourceID is inherited from SMB-Message MessageSourceName is inherited from SMB-Message MessageTarget is inherited from SMB-Message MessageType is inherited from SMB-Message PDP-ResponseType is inherited from PDP-Response ProcessStatus is inherited from SMB-Message
PDP-Response	A PDP message to a PEP providing the result(s) of its policy adjudication. PDP-ResponseType (type: PDP-ResponseType) [1..1]: Identifies whether the response is an operational (e.g., access control) or data (/information) release authorization. Inherited Attributes: Issue-DateTime is inherited from SMB-Message MessageID is inherited from SMB-Message MessageSourceID is inherited from SMB-Message MessageSourceName is inherited from SMB-Message MessageTarget is inherited from SMB-Message MessageType is inherited from SMB-Message ProcessStatus is inherited from SMB-Message

16.1.8 PPS Receive Message

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

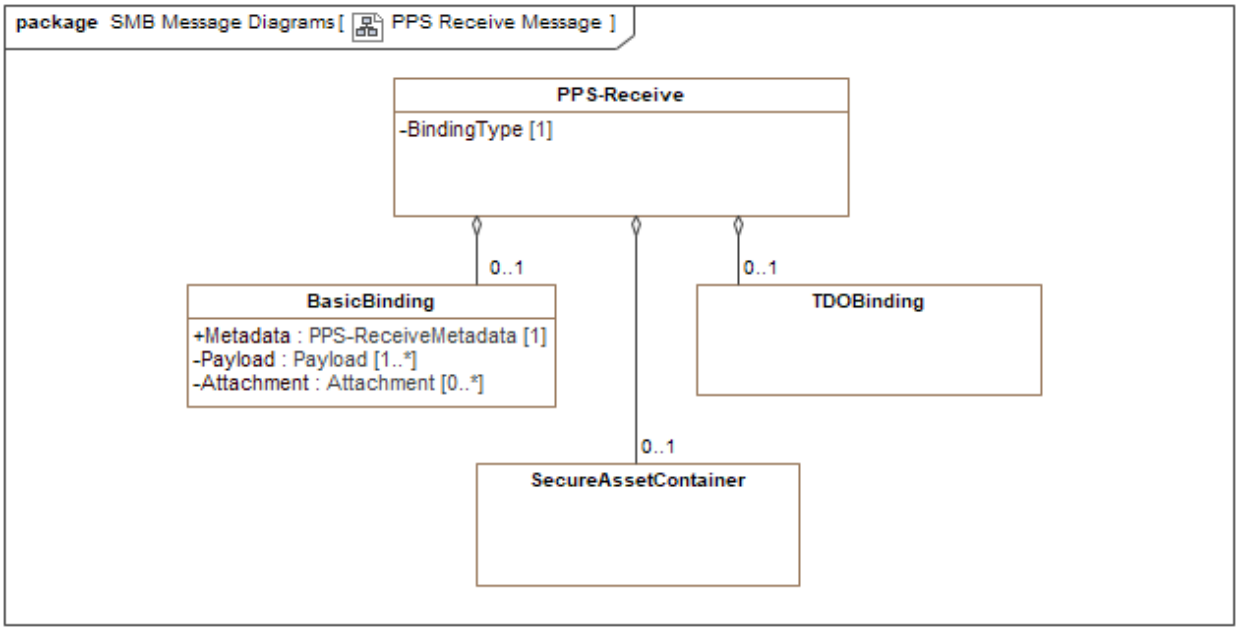


Figure 95 -PPS Receive Message

The following table describes the message elements illustrated in the previous figure - PPS Receive Message.

Table 59 - PPS Receive Message Classes	
Element Name	**Attributes**
BasicBinding	A basic binding for XML and JSON messages. Attachment (type: Attachment) [0..*]: A data or information object related to a payload. Metadata (type: PPS-ReceiveMetadata) [1..1]: Data attributes (Name-value pairs) that describe the content of other data elements in the message. Metadata may include data about the data or information resource in the following areas: • Identifying metadata: Uniquely identifies the data resource(s). • Descriptive metadata: Used to discover or identify the data resource. • Structural metadata: Describing the types, versions, relationships, and other characteristics of the resource. • Administrative metadata: Providing the resource type, permissions, and when and how it was created. • Reference metadata: Providing information about the content and quality of the data or information. Statistical metadata: describing the processes used to collect, process, or produce the data.

`

Table 59 - PPS Receive Message Classes	
Element Name	**Attributes**
	• Legal metadata: Providing information about the creator, copyright holder, and public licensing, if provided. • Security metadata: describing the confidentiality and sensitivity of the data resource(s). • Provenance metadata: providing data ownership and the processing history of the data resource(s). • Geospatial metadata: providing a geospatial reference for the data resource. Metadata is not strictly bound to one of these categories, as it can describe data resources differently. Annex E provides a suggested minimum set of metadata. Payload (type: Payload) [1..*]: A data or information object.
PPS-Receive	A message from a structured messaging PEP to a PPS forwarding authorized data and information elements from an authorized messaging service or middleware. BindingType (type:) [1..1]: Identifies whether the message uses a basic binding type or a Trusted Data Object (TDO) structure.
SecureAssetContainer	The Secure Access container (SAC) is an information element containing the required set of metadata elements in a data file (e.g., text file) and either an encrypted or non-encrypted payload. The SAC is a container (e.g., ZIP File) that binds the metadata to the data, enabling the PEP to access the metadata whether or not the payload is encrypted. The SAC contains one of the payloads or encrypted payloads.
TDOBinding	A data encoding that enables data tagging and cryptographic security features to be applied to one or more data payloads. See ZTDF and ACP240 references for additional information.

16.1.9 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.

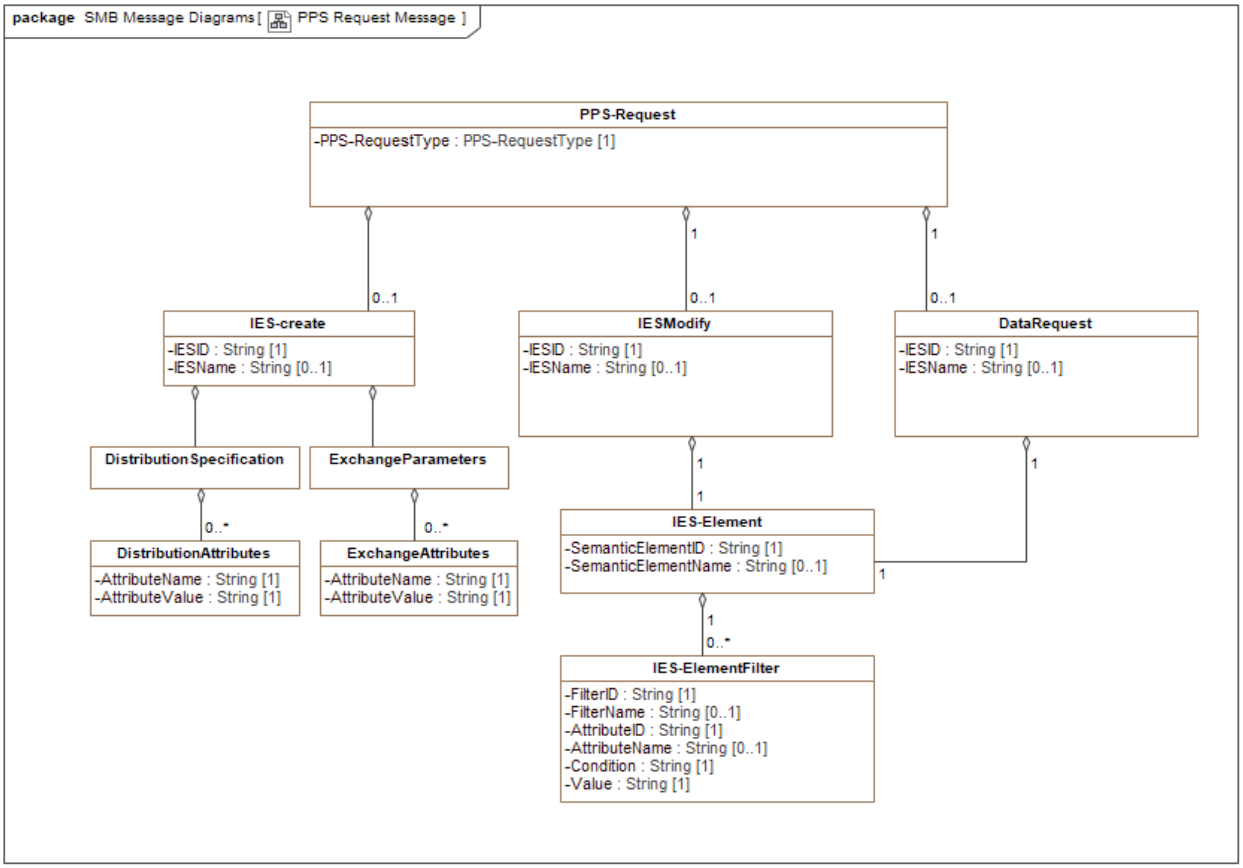


Figure 96 -PPS Request Message

The following table describes the message elements illustrated in the previous figure - PPS Request Message.

Table 60 - PPS Request Message Classes	
Element Name	**Attributes**
DataRequest	A message that provides the parameters for a data request in a request-response data exchange. This request is permitted from any authorized system or user application. The request identifies the semantics, filters, and protocols for packaging and releasing the information. IESID (type: String) [1..1]: The unique identifier for the contract (information exchange specification) the user seeks to participate in. IESName (type: String) [0..1]: The name of the contract (information exchange specification) that the user is seeking to participate in.
DistributionAttributes	Container for a distribution Attribute; AttributeName (type: String) [1..1]: Distribution attribute name.

`

Table 60 - PPS Request Message Classes	
Element Name	**Attributes**
	AttributeValue (type: String) [1..1]: Distribution attribute value.
DistributionSpecification	A set of attributes that describe the distribution channel used under the IES to exchange data. Each IES can define its communication channel. Attributes may include: • AppliedMiddleware; • Assigned PEP Identifier; • Assigned PEP Name; • Channel Identifier; • Channel Name; • Encrypted Metadata and • Encrypted Payload. The IEF-RA provides a list of attributes and values for the PPS as an informational element. (See PPS element Parameter (tag-values) for the IEF-RA (Informational)
ExchangeAttributes	AttributeName (type: String) [1..1]: Name of the exchange attribute. AttributeValue (type: String) [1..1]: Value of the exchange attribute.
ExchangeParameters	A set of attributes that describe and govern the data exchange of the IES. Attributes may be defined in the following areas: • Scheduling; • Exchange; • Processing; • Packaging; • Security; • Logging and • Cryptography. The IEF-RA provides a list of attributes and values for the PPS as an informational element. (See PPS element Parameter (tag-values) for the IEF-RA (Informational)
IES-create	Message that provides the parameters for creating a new information exchange specification. This request is only accepted from an authorized PAP as a PAP command. IESID (type: String) [1..1]:

Table 60 - PPS Request Message Classes	
Element Name	**Attributes**
	The unique identifier for the contract (information exchange specification) the user seeks to participate in. IESName (type: String) [0..1]: The name of the contract (information exchange specification) that the user is seeking to participate in.
IES-Element	Meta description of an IES information element. SemanticElementID (type: String) [1..1]: The Unique Identifier for the base semantic Element. SemanticElementName (type: String) [0..1]: The name of the base semantic Element.
IES-ElementFilter	Characteristics for the filter assigned to the base SemanticElement to produce the required data. AttributeID (type: String) [1..1]: A unique identifier for the attribute used in the filter is needed. AttributeName (type: String) [0..1]: Name of the attribute used in the filter. Condition (type: String) [1..1]: A boolean condition that satisfies the filter. FilterID (type: String) [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.
IESModify	Message that provides the parameters for modifying an information exchange specification during operations. This request is only accepted from an authorized PAP as a PAP command. IESID (type: String) [1..1]: The unique identifier for the contract (information exchange specification) that the user is seeking to participate in. IESName (type: String) [0..1]: The name of the contract (information exchange specification) that the user is seeking to participate in.
PPS-Request	A message from a PEP to the PPS to request the release of information based on a specified semantic element. The request contains the base (unique) identifier for the requested information elements and the identifier for semantic policies to be applied. PPS-RequestType (type: PPS-RequestType) [1..1]:

`

Table 60 - PPS Request Message Classes	
Element Name	**Attributes**
	Identifies if the received request is for data, creation of an IES, or a prescribed modification.

16.1.10 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.

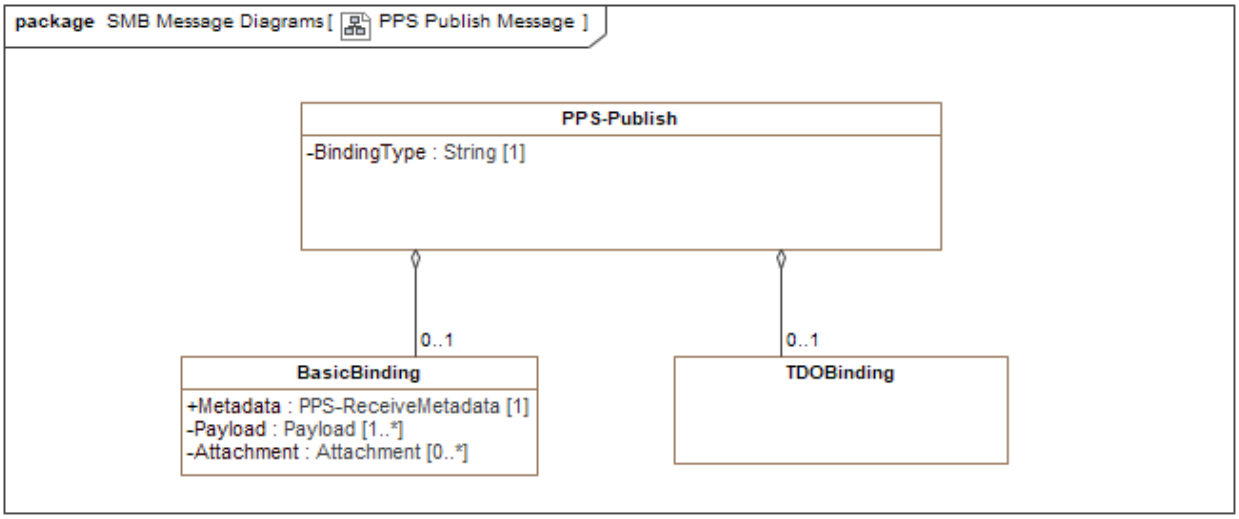


Figure 97 -PPS Publish Message

The following table describes the message elements illustrated in the previous figure - PPS Publish Message.

Table 61 - PPS Publish Message Classes	
Element Name	**Attributes**
BasicBinding	A basic binding for XML and JSON messages. Attachment (type: Attachment) [0..*]: A data or information object related to a payload. Metadata (type: PPS-ReceiveMetadata) [1..1]: Data attributes (Name-value pairs) that describe the content of other data elements in the message. Metadata may include data about the data or information resource in the following areas: • Identifying metadata: Uniquely identifies the data resource(s).

Table 61 - PPS Publish Message Classes	
Element Name	**Attributes**
	• Descriptive metadata: Used to discover or identify the data resource. • Structural metadata: Describing the types, versions, relationships, and other characteristics of the resource. • Administrative metadata: Providing the resource type, permissions, and when and how it was created. • Reference metadata: Providing information about the content and quality of the data or information. Statistical metadata: describing the processes used to collect, process, or produce the data. • Legal metadata: Providing information about the creator, copyright holder, and public licensing, if provided. • Security metadata: describing the confidentiality and sensitivity of the data resource(s). • Provenance metadata: providing data ownership and the processing history of the data resource(s). • Geospatial metadata: providing a geospatial reference for the data resource. Metadata is not strictly bound to one of these categories, as it can describe data resources differently. Annex E provides a suggested minimum set of metadata. Payload (type: Payload) [1..*]: A data or information object.
PPS-Publish	A message from a PPS to a structure messaging PEP directing that PEP to validate and verify that the PPS is authorized to issue the information element(s) and that the specified recipient(s) are authorized to receive and access the enclosed content. The message also directs the communication channels and protocols to be applied. BindingType (type: String) [1..1]: Identifying whether the message required a basic or TDO binding,
TDOBinding	A data encoding that enables data tagging and cryptographic security features to be applied to one or more data payloads. See ZTDF and ACP240 references for additional information.

16.1.11 SSG Request Message

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

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

`

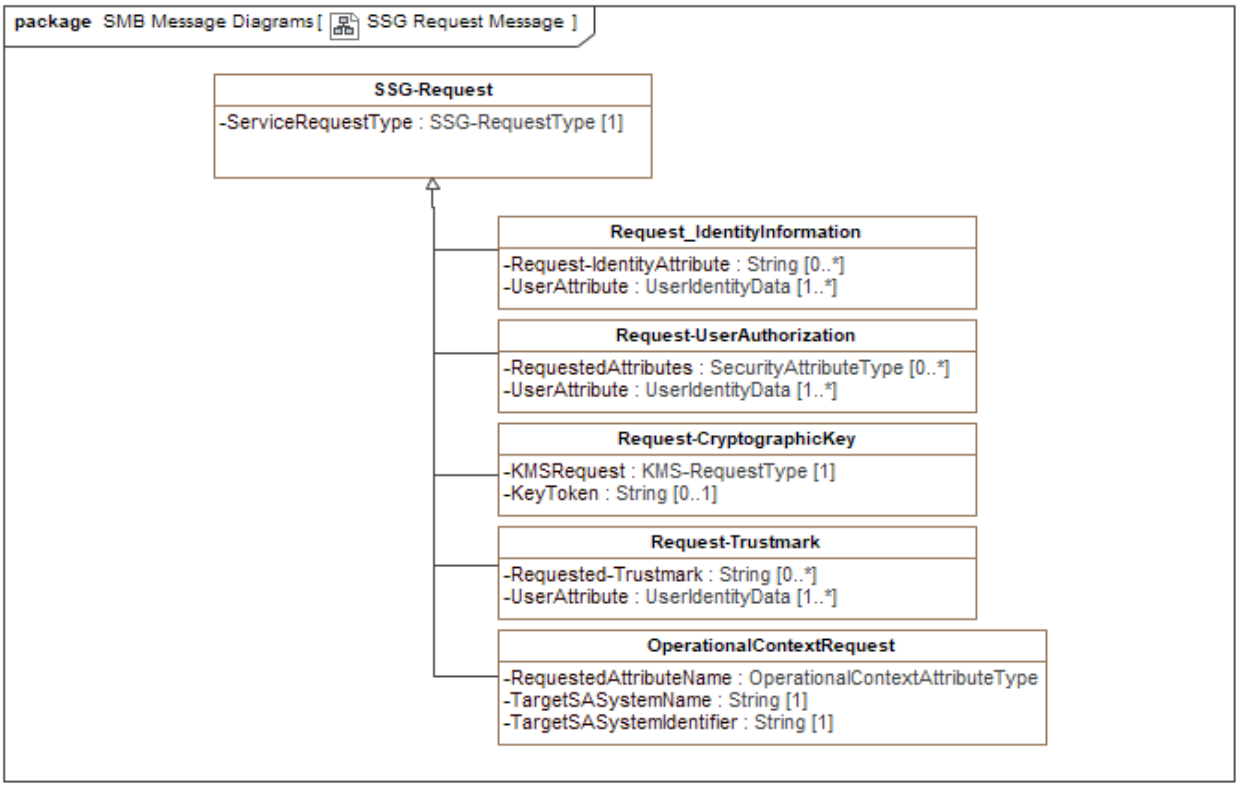


Figure 98 -SSG Request Message

The following table describes the message elements illustrated in the previous figure - SSG Request Message.

Table 62 - SSG Request Message Classes	
Element Name	**Attributes**
OperationalContextRequest	This component requests information about the current operational context/situation. It is used in an environment where ISS policies can be tailored to address changes in the operational context. RequestedAttributeName (type: OperationalContextAttributeType) [1..1]: Identities the operational context attribute being requested. TargetSASystemIdentifier (type: String) [1..1]: Identifier for the target SA system. TargetSASystemName (type: String) [1..1]: Name of the target SA systems.
Request-CryptographicKey	A component request for a cryptographic key from the user-specified and provisioned key management services. This request can request the

Table 62 - SSG Request Message Classes	
Element Name	**Attributes**
	generation of a new key and token pair or the retrieval of an existing key based on the token provided. KeyToken (type: String) [0..1]: Unique identifier for a cryptographic key that the user stores as specified and provisioned key escrow service. KMSRequest (type: KMS-RequestType) [1..1]: Identifies the type of request being made to the KMS.
Request-Trustmark	Component request to a user-specified TrustMark Registry for a user's attributes (/trust marks/policies)—Placeholder for integrating a TrustMark framework. Requested-Trustmark (type: String) [0..*]: Identifies the type(s) of Trustmarks being requested. If the attribute is not sent - the SSG requests all authorized user Trustmarks. UserAttribute (type: UserIdentityData) [1..*]: The user attributed used to obtain information.
Request-UserAuthorization	Component requests users' attributes (/privileges/attributes) from the identity or ICAM system. RequestedAttributes (type: SecurityAttributeType) [0..*]: Identifies the type of attributes (attributes, privileges) being requested. If the attribute list is not sent, the SSG will request all user attributes. UserAttribute (type: UserIdentityData) [1..*]: Identities the attributes to be requested from the identity or ICAM system.
Request_IdentityInformation	Component request for identity information for one or more users. Request-IdentityAttribute (type: String) [0..*]: Identifies the type of identity information being requested. If an attribute is not sent, the SSG requests all user attributes. It is up to the user to identify types available to the PDP for adjudication. UserAttribute (type: UserIdentityData) [1..*]: Identities the attributes to be requested from the identity or ICAM system.
SSG-Request	A message issued by an IEF component to the SSG 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. ServiceRequestType (type: SSG-RequestType) [1..1]: Identifies the type of information being requested from an external user-specified security service.

`

16.1.12 SSG Response Message

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

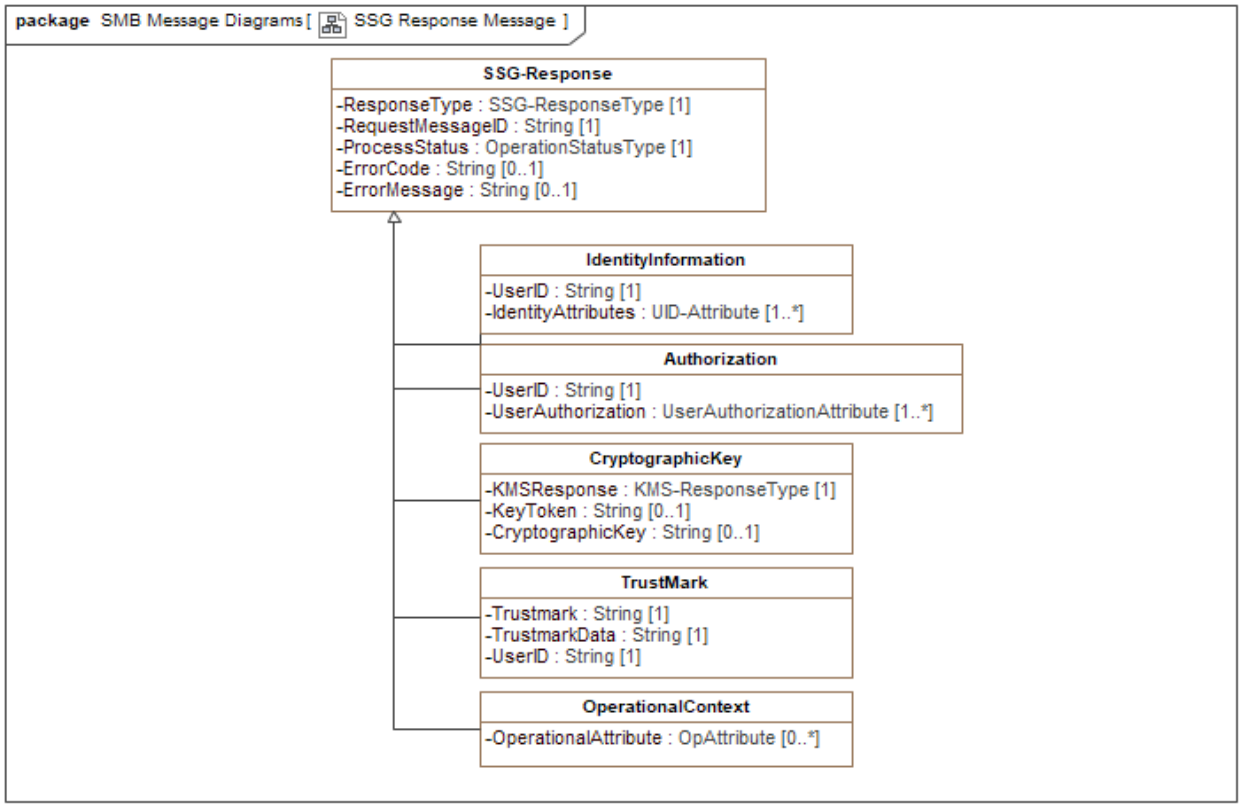


Figure 99 -SSG Response Message

The following table describes the message elements illustrated in the previous figure - SSG Response Message.

Table 63 - SSG Response Message Classes	
Element Name	**Attributes**
Authorization	A message that returns the user attributes, rights, or privileges to the requesting component. UserAuthorization (type: UserAuthorizationAttribute) [1..*]: An array or list of privileges/attributes for a specified set of users. UserID (type: String) [1..1]:
CryptographicKey	**SSG** message that returns a requested cryptographic key to the requesting component. CryptographicKey (type: String) [0..1]:

Table 63 - SSG Response Message Classes	
Element Name	**Attributes**
	Value of the Cryptographic Key. KeyToken (type: String) [0..1]: Token for retrieving the cryptographic key from escrow. KMSResponse (type: KMS-ResponseType) [1..1]: Response type from the key management service.
IdentityInformation	A message that returns user identity information for one or more users to the requesting component. IdentityAttributes (type: UID-Attribute) [1..*]: List of user identity attributes. UserID (type: String) [1..1]: The user's unique identifier.
OperationalContext	A message that returns situational awareness or incident data to the requesting component. OperationalAttribute (type: OpAttribute) [0..*]: Data describing operational context.
SSG-Response	A message from the SSG to the IEF Component providing the requested data or Information elements. 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: String) [1..1]: Unique identifier for the message that requested the information. ResponseType (type: SSG-ResponseType) [1..1]: Identifies the SSG message type, which is used to stage the appropriate processing sequence for the message.
TrustMark	**SSG** message that returns TrustMark attributes to the requesting component. Placeholder for a specialized integration with a Trustmark Registry. Trustmark (type: String) [1..1]: TrustmarkData (type: String) [1..1]: UserID (type: String) [1..1]:

`

16.1.13 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 an information element's transformation (i.e., encryption or decryption).

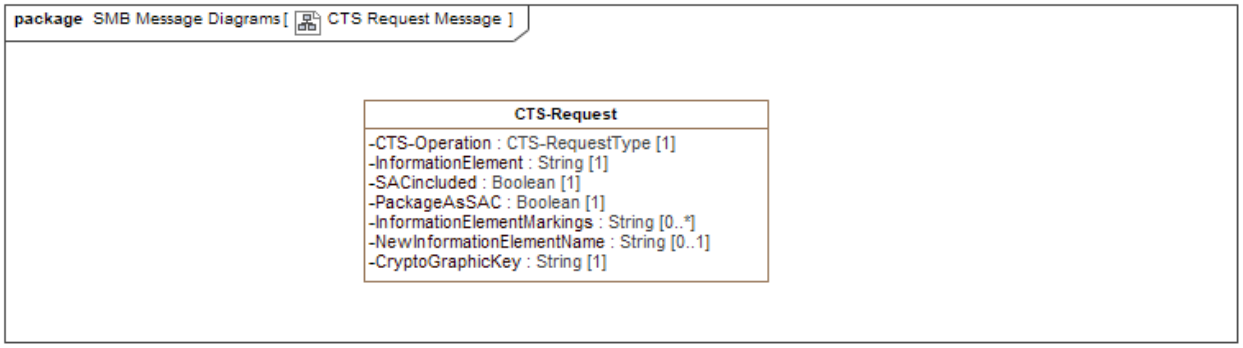


Figure 100 -CTS Request Message

The following table describes the message elements illustrated in the previous figure - CTS Request Message.

Table 64 - CTS Request Message Classes	
Element Name	**Attributes**
CTS-Request	A message to the CTS that requests the transformation (encryption or decryption) of the specified information element. CryptoGraphicKey (type: String) [1..1]: It identifies that the information element contained in the message is an SAC. CTS-Operation (type: CTS-RequestType) [1..1]: The transformation being requested (i.e., encryption or decryption). InformationElement (type: String) [1..1]: The information element that is subject to being transformed by the CTS. InformationElementMarkings (type: String) [0..*]: The marking for the information element must be included within the SAC (See Secure Access Container). Markings are only required when an SAC is being prepared and has not already been provided. Each string has the name-value pair for the marking. NewInformationElementName (type: String) [0..1]:

Table 64 - CTS Request Message Classes	
Element Name	**Attributes**
	This attribute provides a new file name for the information element in the SAC. The CTS must 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 information element within a Secure Access Container (SAC). SACincluded (type: Boolean) [1..1]: Identifies the information element contained in the message as a SAC.

16.1.14 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 information element.

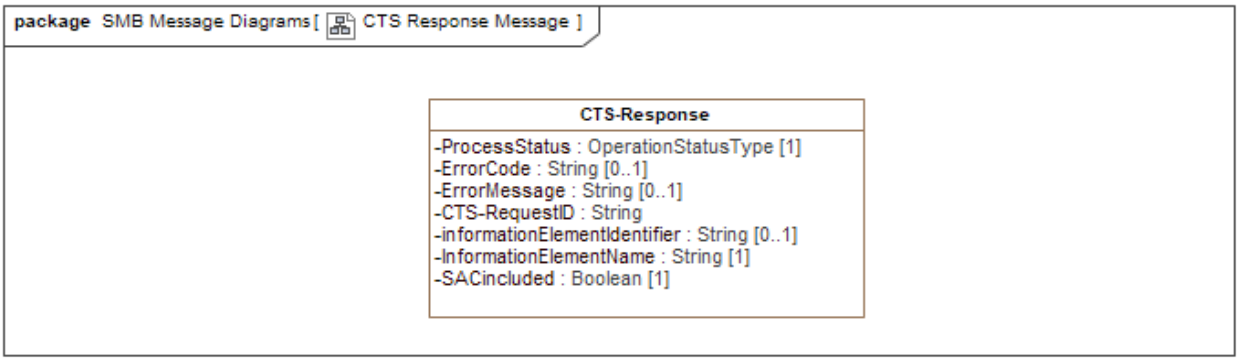


Figure 101 -CTS Response Message

The following table describes the message elements illustrated in the previous figure - CTS Response Message.

Table 65 - CTS Response Message Classes	
Element Name	**Attributes**
CTS-Response	A message from a CTS to an IEF component responding to a cryptographic transformation request. CTS-RequestID (type: String) [1..1]: The unique identifier for the corresponding CTS-Request Message. ErrorCode (type: String) [0..1]: If unable to transform the information element as requested, provide the unique ErrorCode for the issue encountered. ErrorMessage (type: String) [0..1]:

`

Table 65 - CTS Response Message Classes	
Element Name	**Attributes**
	If unable to transform the information element as requested, provide the text describing the issue encountered. informationElementIdentifier (type: String) [0..1]: Transformed information element. This element is not included if an error is encountered during the transformation. InformationElementName (type: String) [1..1]: ProcessStatus (type: OperationStatusType) [1..1]: SACincluded (type: Boolean) [1..1]: Identifies that the information element included in the message is an SAC.

16.1.15 TLS Log Report Message

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

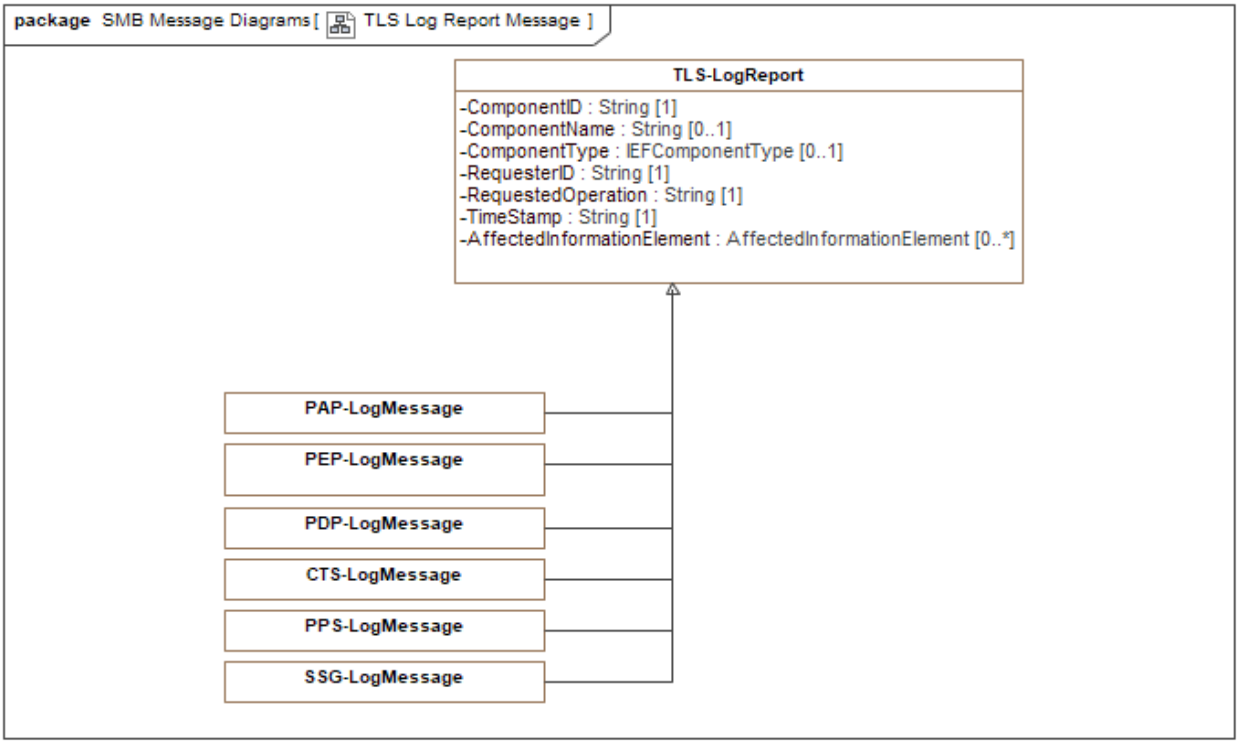


Figure 102 -TLS Log Report Message

The following table describes the message elements illustrated in the previous figure - TLS Log Report Message.

Table 66 - TLS Log Report Message Classes	
Element Name	**Attributes**
CTS-LogMessage	The user specified elements of a CTS log message for the TLS.
PAP-LogMessage	The user specified elements of a PDP log message for the TLS.
PDP-LogMessage	The user specified elements of a PDP log message for the TLS.
PEP-LogMessage	The user specified elements of a PEP log message for the TLS.
PPS-LogMessage	The user specified elements of a PPS log message for the TLS. InformationExchangeSpecification (type: String) [1..1]: Identifies the InformationExchangeSpecification exercised during the transaction.
SSG-LogMessage	User-specified elements of an SSG log message to the TLS.
TLS-LogReport	A 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. AffectedInformationElement (type: AffectedInformationElement) [0..*]: Identifies the information elements affected by the transaction being logged. ComponentID (type: String) [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]:

`

Table 66 - TLS Log Report Message Classes	
Element Name	**Attributes**
	The operation requested on the data asset. RequesterID (type: String) [1..1]: The identity of the user that requested the data transaction. TimeStamp (type: String) [1..1]: A timestamp (date and time) generated by the IEF components (typically system time) indicates when each transaction was handled.

16.1.16 Ack Message

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

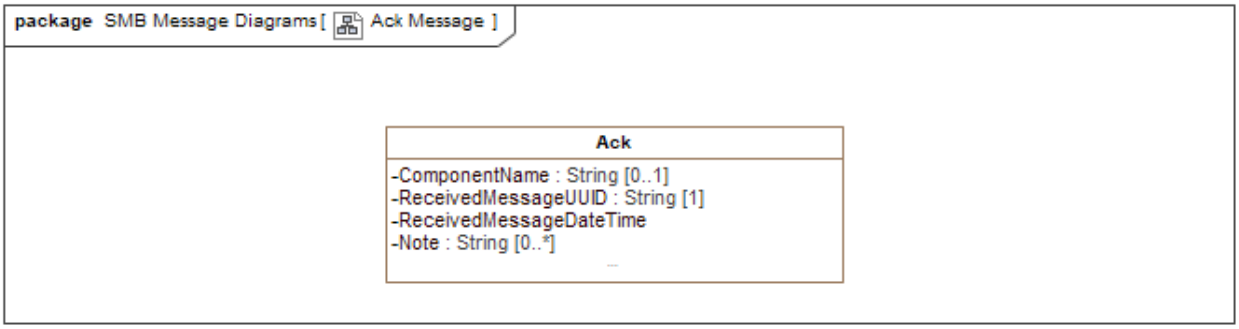


Figure 103 -Ack Message

The following table describes the message elements illustrated in the previous figure - Ack Message.

Table 67 - Ack Message Classes	
Element Name	**Attributes**
Ack	The following figure identifies the data and information elements sent to the originating component to acknowledge the receipt of a message.

16.2 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 to enable the user to adopt the metadata standards that best suit their organizations and communities.

16.2.1 Message Metadata

Metadata elements that may be included in the message envelope or header. These Data Elements are typically sent in the clear (no encryption) so that the sender's and recipient's infrastructure (e.g., PEP) can ascertain attributes to access, process, store, or share the message content.

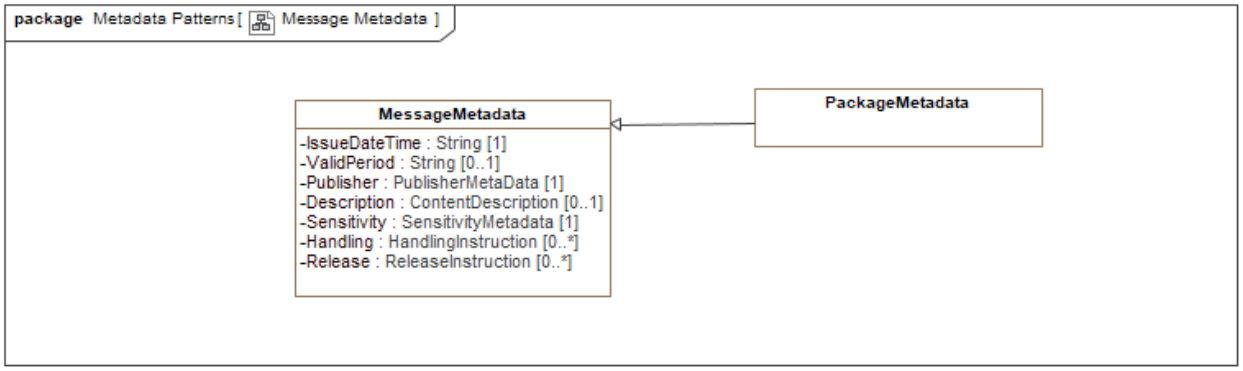


Figure 104 -Message Metadata

The following table describes the message elements illustrated in the previous figure - Message Metadata.

Table 68 - Message Metadata Classes	
Element Name	**Attributes**
MessageMetadata	Metadata elements that may be included in the message envelope or header. These Data Elements are typically sent in the clear (no encryption) so that the sender's and recipient's infrastructure (e.g., PEP) can ascertain attributes to access, process, store, or share the message content. Description (type: ContentDescription) [0..1]: Brief description of the information element or message. Handling (type: HandlingInstruction) [0..*]: Handling instructions to the recipient of the information element or message. IssueDateTime (type: String) [1..1]: The TimeStamp identifies when the information element 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]:

`

Table 68 - Message Metadata Classes	
Element Name	**Attributes**
	Sensitivity of the information element or message. ValidPeriod (type: String) [0..1]: The period or duration for which the data in the information element is valid. The string is a composite of two DateTime strings separated by " / ".
PackageMetadata	Data (tags and markings) that describes the information in the information package.

16.2.2 Information Element Metadata

The following figure identifies the metadata elements for an information payment within a message structure.

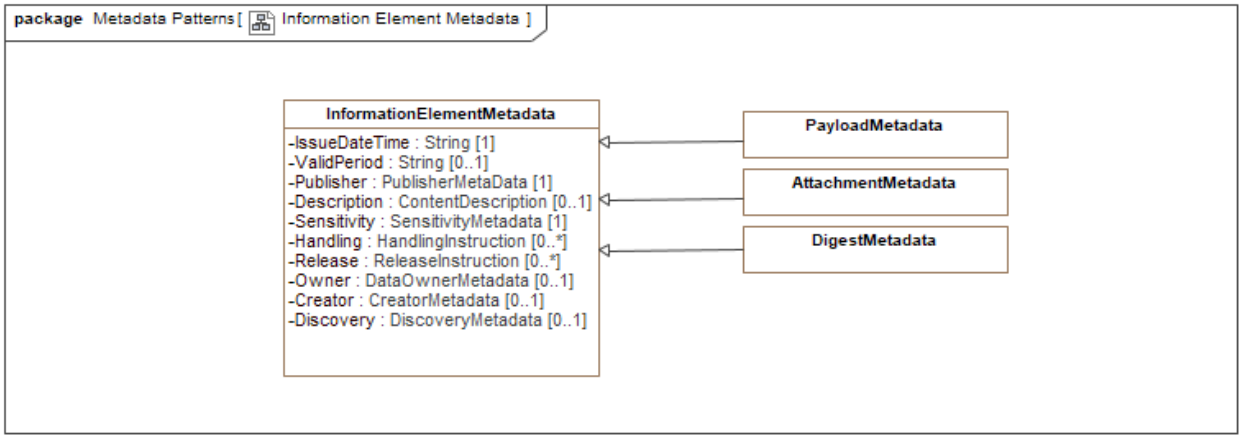


Figure 105 -Information Element Metadata

The following table describes the message elements illustrated in the previous figure - Information Element Metadata.

Table 69 - Information Element Metadata Classes	
Element Name	**Attributes**
AttachmentMetadata	A set of metadata elements may be attached to a message attachment.
DigestMetadata	Set of metadata elements that may be attached to a message digest. Specialization of InformationElementMetadata.
InformationElementMetadata	Metadata describing an information element. Creator (type: CreatorMetadata) [0..1]:

Table 69 - Information Element Metadata Classes	
Element Name	**Attributes**
	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 that enable the discovery of the information element or message. Handling (type: HandlingInstruction) [0..*]: Handling instructions to the recipient for the information element or message. IssueDateTime (type: String) [1..1]: The TimeStamp identifies when the information element 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 instructions 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 information element. 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 InformationElementMetadata.

`

Annex A: IEF-RA XSD and Data Types

Annex A.1 SMB Message XSDs (Informational)

The IEF-RA provides an XML PSM to exchange messages between the IEF components. These XSDs evolved during operational testing between 2018 and 2023. The XSDs are considered informational as they continue evolving with the IEF components and testing with Canadian Government partners.

The XSDs are aligned with the models provided in Clause 16 (above). The XSDs are in a separate file: “mars-2025-02-05 SMB Messages XSD (Informational).zip”.

Annex A.2 PPS Policy Memory Models (Informational)

The PPS uses two memory-based object stores to maintain the operational semantic and exchange policies. To aid in developing IEF implementations, we provide the core elements of the memory models used during testing (2018 – 2023). These models are provided as informational elements because the Packaging and Processing Service Specification would define the normative versions of the PPS elements. They remained very stable artefacts during that testing period.

The two models are:

1. The Semantic Policy Model (Annex A.2.1) contains the data (rules and constraints) needed to govern the processing and packaging of data elements authorised to be received and released and
2. The Information Exchange Specification Model (IES) or exchange Policy model (Annex A.2.2) contains data needed to govern the exchange of data and information elements (/messages) to specified recipients and

Annex A.2.1 - Semantic Policy Memory Model

The following figure provides a meta-model for semantic policies held in PPS memory. The memory model was successfully tested at NATO CWIX between 2019 and 2023. It is provided as an informational element of the IEF-RA specification simply because it is a core part of a future Packaging and Processing Service (PPS) specification. The publication of the PPS specification will deprecate the information in this annex.

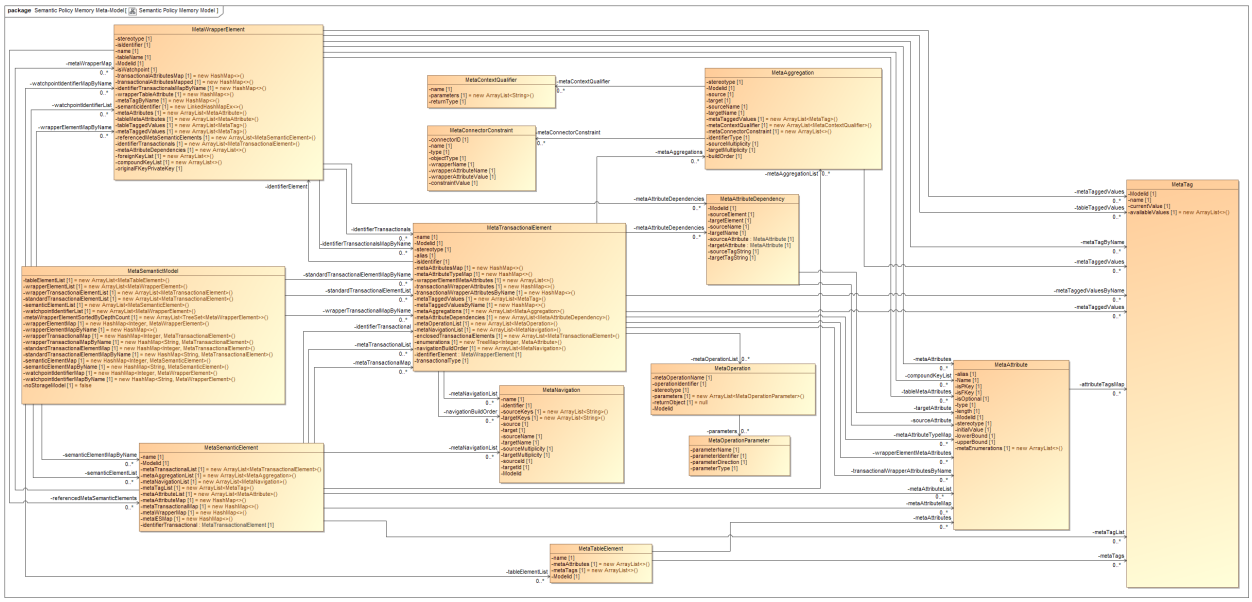


Figure 106 -Semantic Policy Memory Model

The following table describes the message elements illustrated in the previous figure - Semantic Policy Memory Model.

Table 70 - Semantic Policy Memory Model Classes	
Element Name	**Attributes**
MetaAggregation	Defines an aggregation arc linking the focal element of the subtended elements it aggregates into the semantic pattern. buildOrder (type:) [1..1]: Identifies the position in the order in which the subtended element is aggregated. identifierType (type:) [1..1]: Identifies if the aggregation is an identifier, watchpoint, or watchpoint-identifier. metaConnectorConstraint (type:) [1..1]: List of navigation constraints assigned to the aggregation based on a specific data value. metaConnectorConstraint (type: MetaConnectorConstraint) [0..*]: metaContextQualifier (type:) [1..1]: List of filters (context qualifiers) assigned to the aggregation. metaContextQualifier (type: MetaContextQualifier) [0..*]: metaTaggedValues (type:) [1..1]: List of parameters assigned to the aggregation. metaTaggedValues (type: MetaTag) [0..*]: Modelid (type:) [1..1]: UID of the tag within the meta model. source (type:) [1..1]: The object being integrated. sourceMultiplicity (type:) [1..1]: Multiplicity of the data elements being aggregated. sourceName (type:) [1..1]: The name of the element to be aggregated. stereotype (type:) [1..1]: The stereotype of the aggregation arc. (e.g., Identifier, watchpoint, watchpoint-identifier) target (type:) [1..1]: The target for the integration of the source object. targetMultiplicity (type:) [1..1]: Multiplicity of the aggregating element (always equal to 1). targetName (type:) [1..1]:

`

Table 70 - Semantic Policy Memory Model Classes	
Element Name	**Attributes**
	The name of the element that includes the aggregated data element.
MetaAttribute	Defines an attribute within the PPS (semantic) memory model. alias (type:) [1..1]: The attribute's alias. attributeTagsMap (type: MetaTag) [0..*]: initialValue (type:) [1..1]: The attribute's initial or default value. isFKey (type:) [1..1]: The attribute is identified as a foreign key within the element. isOptional (type:) [1..1]: The attribute is identified as optional within the element. isPKey (type:) [1..1]: The attribute is identified as a primary key within the element. length (type:) [1..1]: The attribute's length, if needed. lowerBound (type:) [1..1]: The attribute's lower bound. metaEnumerations (type:) [1..1]: List of allowable enumerated values. Modelid (type:) [1..1]: UID of the tag within the meta model. Name (type:) [1..1]: The Attribute's name. stereotype (type:) [1..1]: The attribute's stereotype. type (type:) [1..1]: The attribute's data type. upperBound (type:) [1..1]: The attribute's upper bound.
MetaAttributeDependency	Defines an attribute dependency within the PPS (semantic) policy model. Modelid (type:) [1..1]: UID of the tag within the meta model. sourceAttribute (type: MetaAttribute) [1..1]: sourceAttribute (type: MetaAttribute) [1..1]:

Table 70 - Semantic Policy Memory Model Classes	
Element Name	**Attributes**
	The data associated with the source attribute of the dependency arc. sourceElement (type:) [1..1]: The object is identified as the data source of the dependency arc. sourceName (type:) [1..1]: The name of the source object. sourceTagString (type:) [1..1]: Role name of the source end of the dependency arc. targetAttribute (type: MetaAttribute) [1..1]: targetAttribute (type: MetaAttribute) [1..1]: The data associated with the target attribute of the dependency arc. targetElement (type:) [1..1]: The object is identified as the target of the dependency arc. targetName (type:) [1..1]: The name of the target object. targetTagString (type:) [1..1]: Role name of the target end of the dependency arc.
MetaConnectorConstraint	Defines an aggregation constraint limits processing to data elements containing instance data to complete the semantic. connectorID (type:) [1..1]: UUID for the connector (/aggregation arc). constraintValue (type:) [1..1]: The value that constrains the aggregation. name (type:) [1..1]: Nam of the aggregation Arc. objectType (type:) [1..1]: Object type at the source of the aggregation arc. type (type:) [1..1]: The arc type. wrapperAttributeName (type:) [1..1]: The attribute name used to constrain the aggregation. wrapperAttributeValue (type:) [1..1]: The actual value of the attribute used to constrain the aggregation. wrapperName (type:) [1..1]:

`

Table 70 - Semantic Policy Memory Model Classes	
Element Name	**Attributes**
	The name of the identifying wrapper element for the focal element of the policy.
MetaContextQualifier	Defines a context qualifier used to redact (/filter) data elements in the semantic patterns. name (type:) [1..1]: Name of the context qualifier (/operation). parameters (type:) [1..1]: List of parameters assigned to the operation used to redact (filter/transform) the data in the subtended data element. returnType (type:) [1..1]: The context qualifier operation uses the return type of operation.
MetaNavigation	Defines a navigation arc that guides the aggregation of data elements and retains referential integrity in the data. identifier (type:) [1..1]: Identifier for the navigation arc. Modelid (type:) [1..1]: UID of the tag within the meta model. name (type:) [1..1]: Name of the navigation arc. source (type:) [1..1]: The object to be aggregated. sourceId (type:) [1..1]: The UID for the data source for the aggregation. sourceKeys (type:) [1..1]: List of identifiers (/keys) used to connect the source data element to be aggregated. The multiple keys enable the use of compound keys from the data store. sourceMultiplicity (type:) [1..1]: The multiplicity of the data object to be aggregated. sourceName (type:) [1..1]: The name of the data object to be aggregated. target (type:) [1..1]: The target of the data aggregation. targetId (type:) [1..1]: The UID for the target of the aggregation. targetKeys (type:) [1..1]:

Table 70 - Semantic Policy Memory Model Classes	
Element Name	**Attributes**
	List of identifiers (/keys) used to connect the target data element for the aggregated element (e.g., focal element). The multiple keys enable the use of compound keys from the data store. targetMultiplicity (type:) [1..1]: The multiplicity of the target objects holding the aggregation. (this multiplicity is always 1) targetName (type:) [1..1]: The name of the data object to hold the aggregation.
MetaOperation	Defines a transformation (/operation) in a PPS (semantic) policy model. metaOperationName (type:) [1..1]: The name of the operation. Modelid (type:) [1..1]: UID of the tag within the meta model. operationIdentifier (type:) [1..1]: UId for the operation. parameters (type:) [1..1]: List of parameters parameters for the operation. parameters (type: MetaOperationParameter) [0..*]: returnObject (type:) [1..1]: The return object from the operation. stereotype (type:) [1..1]: The stereotype of the operation.
MetaOperationParameter	The definition of a parameter in an operation in a PPS (semantic) policy model. parameterDirection (type:) [1..1]: The operational direction Input, output) of the param in the operation. parameterIdentifier (type:) [1..1]: The UID for the parameter. parameterName (type:) [1..1]: The operation parameter name. parameterType (type:) [1..1]: Defines an operation parameter in the PPS Semantic Policy model.
MetaSemanticElement	Defines a semantic (/data) pattern (/policy) for a message for a specified data exchange withing a specified domain (/community) of interest.

`

Table 70 - Semantic Policy Memory Model Classes	
Element Name	**Attributes**
	identifierTransactional (type: MetaTransactionalElement) [1..1]: The transactional element holds the primary key (/identified) for the semantic build (/construction). identifierTransactional (type: MetaTransactionalElement) [1..1]: metaAggregationList (type: MetaAggregation) [0..*]: metaAggregationList (type:) [1..1]: List of aggregation arcs originating from the semantic element. metaAttributeList (type: MetaAttribute) [0..*]: metaAttributeList (type:) [1..1]: List of attributes included in the semantic element. metaAttributeMap (type: MetaAttribute) [0..*]: metaAttributeMap (type:) [1..1]: List of attributes included in the semantic element. metaIESMap (type: MetaIES) [0..*]: metaIESMap (type:) [1..1]: List of exchange specifications that include the semantic element. metaNavigationList (type: MetaNavigation) [0..*]: metaNavigationList (type:) [1..1]: List of navigation arcs included in the semantic element. metaTagList (type:) [1..1]: List of properties (/tag values) for the semantic element. metaTagList (type: MetaTag) [0..*]: metaTransactionalList (type:) [1..1]: List of transactional elements encompassed by the semantic element. metaTransactionalList (type: MetaTransactionalElement) [0..*]: metaTransactionalMap (type:) [1..1]: List of transactional elements aggregated in the semantic. metaTransactionalMap (type: MetaTransactionalElement) [0..*]: metaWrapperMap (type:) [1..1]: List of wrapper elements in the semantic aggregation. metaWrapperMap (type: MetaWrapperElement) [0..*]: Modelid (type:) [1..1]: UID of the tag within the meta model. name (type:) [1..1]: Name of the semantic.

Table 70 - Semantic Policy Memory Model Classes	
Element Name	**Attributes**
MetaSemantictModel	Defines a semantic policy model for a specified domain of interest. metaWrapperElementSortedByDepthCount (type:) [1..1]: List of wrapper elements (/sorted by depth count) associated with the semantic policy model. noStorageModel (type:) [1..1]: Identifies whether or not the model uses a storage model. semanticElementList (type:) [1..1]: List of semantic elements associated with the semantic policy model. semanticElementList (type: MetaSemanticElement) [0..*]: semanticElementMap (type:) [1..1]: List of transactional elements associated with the semantic policy model. semanticElementMapByName (type:) [1..1]: List of transactional elements associated with the semantic policy model. semanticElementMapByName (type: MetaSemanticElement) [0..*]: standardTransactionalElementList (type:) [1..1]: List of transactional elements associated with the semantic policy model. standardTransactionalElementList (type: MetaTransactionalElement) [0..*]: standardTransactionalElementMap (type:) [1..1]: standardTransactionalElementMapByName (type:) [1..1]: List of transactional elements associated with the semantic policy model. standardTransactionalElementMapByName (type: MetaTransactionalElement) [0..*]: tableElementList (type:) [1..1]: List of table elements associated with the semantic policy model. tableElementList (type: MetaTableElement) [0..*]: watchpointIdentifierList (type:) [1..1]: List of watchpoints associated with the semantic policy model. watchpointIdentifierList (type: MetaWrapperElement) [0..*]: watchpointIdentifierMap (type:) [1..1]: List of watchpoint elements associated with the semantic policy model. watchpointIdentifierMapByName (type:) [1..1]:

`

Table 70 - Semantic Policy Memory Model Classes	
Element Name	**Attributes**
	List of watchpoint elements (/wrapper element) by name. watchpointIdentifierMapByName (type: MetaWrapperElement) [0..*]: wrapperElementList (type:) [1..1]: List of wrapper elements associated with the semantic policy model. wrapperElementMap (type:) [1..1]: List of wrapper elements associated with the semantic policy model. wrapperElementMapByName (type:) [1..1]: List of wrapper elements associated with the semantic policy model. wrapperElementMapByName (type: MetaWrapperElement) [0..*]: wrapperTransactionalElementList (type:) [1..1]: List of wrapper transactional elements associated with the semantic policy model. wrapperTransactionalMap (type:) [1..1]: List of wrapper transactional elements associated with the semantic policy model. wrapperTransactionalMapByName (type:) [1..1]: List of wrapper transactional elements associated with the semantic policy model. wrapperTransactionalMapByName (type: MetaTransactionalElement) [0..*]:
MetaTableElement	Defines a table element underpinning the wrapper elements in the pattern (/policy). metaAttributes (type: MetaAttribute) [0..*]: metaAttributes (type:) [1..1]: List of attributes in the table of the storage model. metaTags (type:) [1..1]: List of properties (/tag values) applied to the table in the storage model. metaTags (type: MetaTag) [0..*]: Modelid (type:) [1..1]: name (type:) [1..1]: Name of the table in the storage model.
MetaTag	Defines a property (/tagged values) in the PPS (semantic) memory model. availableValues (type:) [1..1]:

Table 70 - Semantic Policy Memory Model Classes	
Element Name	**Attributes**
	Allowable values for the tag. currentValue (type:) [1..1]: Current value of the tag. Modelid (type:) [1..1]: UID of the tag within the meta model. name (type:) [1..1]: Name of the Tag.
MetaTransactionalElement	Defines a transactional pattern (/policy) used to aggregate, transform, label, and redact elements in the semantic pattern (/policy). The transactional elements are temporary, memory-based constructs discarded after packaging is complete. alias (type:) [1..1]: Alternate name for the transactional element. enclosedTransactionalElements (type:) [1..1]: List of transactional elements enclosed by this transactional element. enumerations (type:) [1..1]: This of enumerations used by this transactional element. identifierElement (type: MetaWrapperElement) [1..1]: The wrapper element holds the primary key for constructing the transactional element. identifierElement (type: MetaWrapperElement) [1..1]: isIdentifier (type:) [1..1]: Identifies whether or not the transactional element is a semantic identifier (holds the primary identifier for the semantic). metaAggregations (type: MetaAggregation) [0..*]: metaAggregations (type:) [1..1]: List of aggregations associated with the transactional element. metaAttributeDependencies (type: MetaAttributeDependency) [0..*]: metaAttributeDependencies (type:) [1..1]: List of attribute dependencies between the transactional element attributes and its subtended elements. metaAttributesMap (type:) [1..1]: List of attributes held by the transactional element. metaAttributeTypeMap (type: MetaAttribute) [0..*]: metaAttributeTypeMap (type:) [1..1]:

`

Table 70 - Semantic Policy Memory Model Classes	
Element Name	**Attributes**
	List of attribute types. metaNavigationList (type: MetaNavigation) [0..*]: metaNavigationList (type:) [1..1]: List of navigations (/referential links) associated with the transactional element. metaOperationList (type: MetaOperation) [0..*]: metaOperationList (type:) [1..1]: List of transformations (/operations) enabled by the transactional element. metaTaggedValues (type: MetaTag) [0..*]: metaTaggedValues (type:) [1..1]: List of transactional element parameters (/tag values). metaTaggedValuesByName (type: MetaTag) [0..*]: metaTaggedValuesByName (type:) [1..1]: List of parameter names (/tag values) for the transactional element. Modelid (type:) [1..1]: UID of the tag within the meta model. name (type:) [1..1]: Name of the transactional element. navigationBuildOrder (type: MetaNavigation) [0..*]: navigationBuildOrder (type:) [1..1]: The built (/construction) order for the subtended elements. stereotype (type:) [1..1]: Stereotype of the transactional element. transactionalType (type:) [1..1]: Type of Transactional (e.g., Standard or Wrapper) transactionalWrapperAttributes (type:) [1..1]: List of attributes within the transactional element. transactionalWrapperAttributesByName (type: MetaAttribute) [0..*]: transactionalWrapperAttributesByName (type:) [1..1]: List of names within the transactional element. wrapperElementMetaAttributes (type: MetaAttribute) [0..*]: wrapperElementMetaAttributes (type:) [1..1]: List wrapper element (/identifier) attributes.

Table 70 - Semantic Policy Memory Model Classes	
Element Name	**Attributes**
MetaWrapperElement	Defines a for a wrapper element attributes in the semantic policy model. The wrapper element holds one instance (e.g., 1 row of data from an RDBMS), one data object, or none file to be aggregated into one or more semantic patterns (/policies) compoundKeyList (type: MetaAttribute) [0..*]: compoundKeyList (type:) [1..1]: The list of compound keys (references) used to navigate transactional patterns. The list is derived from the underlying table and key definitions in the database representation and contained in the semantic policy model. foreignKeyList (type:) [1..1]: List of foreign private keys (references) used to navigate from the wrapper to other elements in a transactional pattern. The list is derived from the underlying table and key definitions in the database representation and contained in the semantic policy model. identifierTransactionals (type: MetaTransactionalElement) [0..*]: identifierTransactionals (type:) [1..1]: List of transactional elements using the wrapper element as an identifier. identifierTransactionalsMapByName (type: MetaTransactionalElement) [0..*]: identifierTransactionalsMapByName (type:) [1..1]: List of Transactional Elements referencing the wrapper element. isIdentifier (type:) [1..1]: Identifies the wrapper element as an identifier (holding the identifier [primary key] for a transactional build). isWatchpoint (type:) [1..1]: Identifies that the wrapper element is a watchpoint (trigger) for releasing data under a specified semantic and Information Exchange Specification. metaAttributeDependencies (type: MetaAttributeDependency) [0..*]: metaAttributeDependencies (type:) [1..1]: List of dependency arcs originating from this wrapper element. metaAttributes (type: MetaAttribute) [0..*]: metaAttributes (type:) [1..1]: List of attributes enclosed by the wrapper element. metaTagByName (type: MetaTag) [0..*]: metaTagByName (type:) [1..1]:

`

Table 70 - Semantic Policy Memory Model Classes	
Element Name	**Attributes**
	List of wrapper parameters (tag Values). metaTaggedValues (type: MetaTag) [0..*]: metaTaggedValues (type:) [1..1]: List of wrapper parameters (tag Values). Modelid (type:) [1..1]: UID of the tag within the meta model. name (type:) [1..1]: Name of the wrapper element in the semantic policy model. originalFKeyPrivateKey (type:) [1..1]: List of foreign private keys (references) used to navigate from the wrapper to other elements in a transactional pattern. The list is derived from the underlying table and key definitions in the database representation and contained in the semantic policy model. referencedMetaSemanticElements (type: MetaSemanticElement) [0..*]: referencedMetaSemanticElements (type:) [1..1]: List of semantic elements referencing the wrapper element. semanticIdentifier (type:) [1..1]: List of semantic elements integrating (/aggregating) this wrapper element. stereotype (type:) [1..1]: The wrapper element stereotype. tableMetaAttributes (type: MetaAttribute) [0..*]: tableMetaAttributes (type:) [1..1]: List of table attributes associated with the wrapper element attributes. tableName (type:) [1..1]: The name of the storage model table is associated with this wrapper element. tableTaggedValues (type: MetaTag) [0..*]: tableTaggedValues (type:) [1..1]: List of properties (/tag values) for the table associated with this wrapper element. transactionalAttributesMap (type:) [1..1]: List of attributes associated with the wrapper transactional element. transactionalAttributesMapped (type:) [1..1]: List of transactional attributes referencing the wrapper element.

Table 70 - Semantic Policy Memory Model Classes	
Element Name	**Attributes**
	wrapperTableAttribute (type:) [1..1]: List of table attributes associated with the wrapper element attributes.

Annex A.2.2 - Exchange Policy Memory Model

The following figure provides a meta-model for exchange policies held in PPS memory. The memory model was successfully tested at NATO CWIX between 2019 and 2023 and Bold Quest in 2024. It is provided as an informational element of the IEF-RA specification simply because it is a core part of a future Packaging and Processing Service (PPS) specification. The publication of the PPS specification will deprecate the information in this annex.

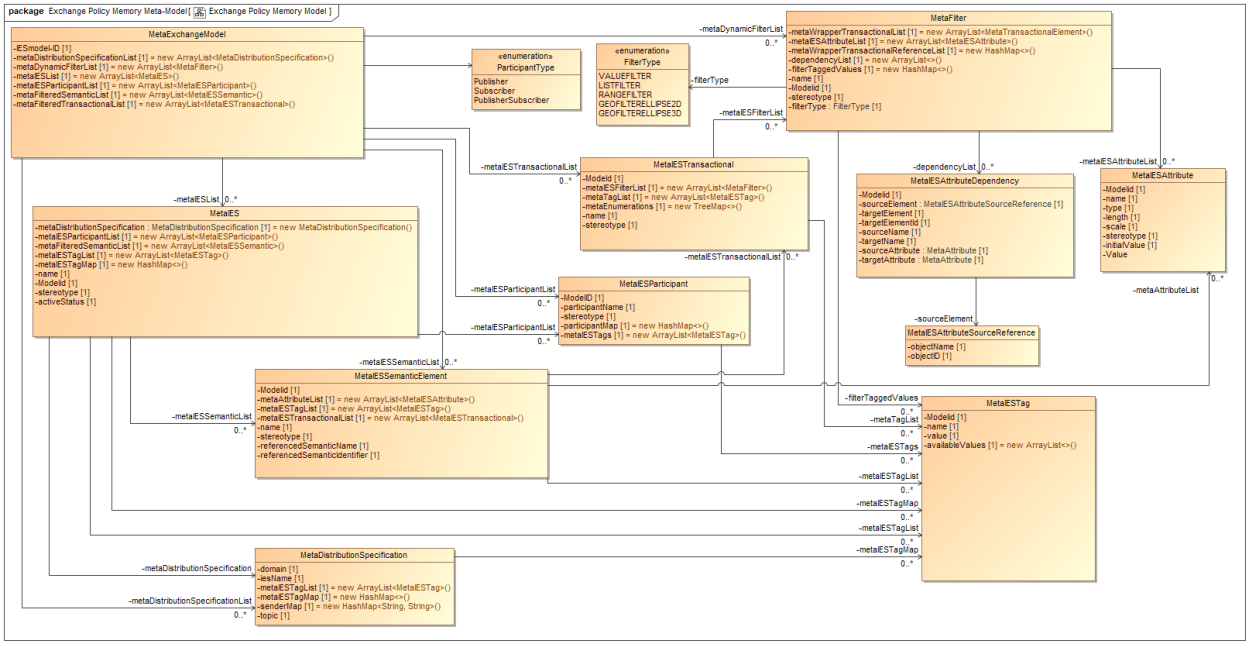


Figure 107 -Exchange Policy Memory Model

The following table describes the message elements illustrated in the previous figure - Exchange Policy Memory Model.

Table 71 - Exchange Policy Memory Model Classes	
Element Name	**Attributes**
MetaDistributionSpecification	Defines the parameters for the exchange specification distribution services. domain (type:) [1..1]: Defines the domain of interest for the data exchange.

`

Table 71 - Exchange Policy Memory Model Classes	
Element Name	**Attributes**
	iesName (type:) [1..1]: I name of the exchange specification. metaIESTagList (type:) [1..1]: List of parameters for the distribution specifications. metaIESTagMap (type:) [1..1]: List of parameters for the distribution specifications. metaIESTagMap (type: MetaIESTag) [0..*]: senderMap (type:) [1..1]: List of publishers for this distribution. topic (type:) [1..1]: The topic used for the exchange.
MetaExchangeModel	Defines the structure and content of the PPS exchange model. IESmodel-ID (type:) [1..1]: UID for the exchange model. metaDistributionSpecificationList (type: MetaDistributionSpecification) [0..*]: metaDistributionSpecificationList (type:) [1..1]: List of distribution specifications in the exchange model. metaDynamicFilterList (type: MetaFilter) [0..*]: metaDynamicFilterList (type:) [1..1]: List of dynamic (/guard) filters in the exchange model. metaFilteredSemanticList (type:) [1..1]: List of the filtered semantic elements included in the exchange model. metaFilteredTransactionalList (type:) [1..1]: List of filtered transactional elements associated with the exchange model. metaIESList (type: MetaIES) [0..*]: metaIESList (type:) [1..1]: List of information exchange specifications in the exchange model. metaIESParticipantList (type:) [1..1]: List of participants associated with the exchanges in the model. metaIESParticipantList (type: MetaIESParticipant) [0..*]: metaIESSemanticList (type: MetaIESSemanticElement) [0..*]: metaIESTransactionalList (type: MetaIESTransactional) [0..*]: policyModel (type: MetaIESPolicyModel) [1..1]:

Table 71 - Exchange Policy Memory Model Classes	
Element Name	**Attributes**
MetaFilter	Definition of a dynamic or guard filter within the exchange policies. dependencyList (type:) [1..1]: List of dependency arcs used by the dynamic or guard filters. dependencyList (type: MetaIESAttributeDependency) [0..*]: filterTaggedValues (type:) [1..1]: List of filter properties (/tag values) used by the dynamic or guard filters. filterTaggedValues (type: MetaIESTag) [0..*]: filterType (type: FilterType) [1..1]: The types of filtering used. filterType (type: FilterType) [1..1]: metaIESAttributeList (type:) [1..1]: List of attributes used by the dynamic or guard filters. metaIESAttributeList (type: MetaIESAttribute) [0..*]: metaWrapperTransactionalList (type:) [1..1]: List of wrapper transactional used by the dynamic or guard filters. metaWrapperTransactionalList (type: MetaTransactionalElement) [0..*]: metaWrapperTransactionalReferenceList (type:) [1..1]: List of wrapper transactional used by the dynamic or guard filters. Modelid (type:) [1..1]: UID of the tag within the meta-model. name (type:) [1..1]: Name of the filter. stereotype (type:) [1..1]: Stereotype of the filer
MetaIES	Defines the IES in a PPS exchange model. activeStatus (type:) [1..1]: Identifies if the IES is activated or deactivated. metaDistributionSpecification (type: MetaDistributionSpecification) [1..1]: metaDistributionSpecification (type: MetaDistributionSpecification) [1..1]: Identifies the distribution specification used by the IES. metaFilteredSemanticList (type:) [1..1]: List of filtered semantics carried within the exchange.

`

Table 71 - Exchange Policy Memory Model Classes	
Element Name	**Attributes**
	metaIESParticipantList (type:) [1..1]: List of participants using the IES. metaIESParticipantList (type: MetaIESParticipant) [0..*]: metaIESSemanticList (type: MetaIESSemanticElement) [0..*]: metaIESTagList (type:) [1..1]: A list of properties (/tag values) for the exchange specification. metaIESTagList (type: MetaIESTag) [0..*]: metaIESTagMap (type:) [1..1]: A list of properties (/tag values) for the exchange specification. metaIESTagMap (type: MetaIESTag) [0..*]: Modelid (type:) [1..1]: UID of the tag within the meta-model. name (type:) [1..1]: Name of the IES. stereotype (type:) [1..1]: The stereotype of the IES.
MetaIESAttribute	The definition of an attribute used as part of a dynamic or guard filter. initialValue (type:) [1..1]: The default value. length (type:) [1..1]: The attribute length. Modelid (type:) [1..1]: UID of the tag within the meta-model. name (type:) [1..1]: Name of the attribute. scale (type:) [1..1]: The attribute scale. stereotype (type:) [1..1]: The attribute stereotype. type (type:) [1..1]: The attributes type. Value (type:) [1..1]: The value of the attribute.
MetaIESAttributeDependency	Definition for an attribute dependencies within a dynamic (/guard) filter. Modelid (type:) [1..1]:

Table 71 - Exchange Policy Memory Model Classes	
Element Name	**Attributes**
	UID of the tag within the meta-model. sourceAttribute (type: MetaAttribute) [1..1]: The attributes used in int the filter. sourceAttribute (type: MetaAttribute) [1..1]: sourceElement (type: MetaIESAttributeSourceReference) [1..1]: The source element of the data attributed is used in the filter. sourceElement (type: MetaIESAttributeSourceReference) [1..1]: sourceName (type:) [1..1]: The name of the filter element. targetAttribute (type: MetaAttribute) [1..1]: The attribute identified in the filter element. targetAttribute (type: MetaAttribute) [1..1]: targetElement (type:) [1..1]: The element holding the attribute used in the filter. targetElementId (type:) [1..1]: The UID for the element holding the attribute used in the filter. targetName (type:) [1..1]: The name of the element holding the attribute used in the filter.
MetaIESAttributeSourceReference	Defines the reference for a source element in a filter attribute dependency. objectID (type:) [1..1]: The model UID for the reference source. objectName (type:) [1..1]: The name of the reference source.
MetaIESParticipant	Defines a participant to an exchange specification. metaIESTags (type:) [1..1]: List of parameters (/tag values) associated with the participant. metaIESTags (type: MetaIESTag) [0..*]: ModelID (type:) [1..1]: UID of the tag within the meta model. participantMap (type:) [1..1]: List of IESs the participant participantName (type:) [1..1]: Name of the participant. stereotype (type:) [1..1]:

`

Table 71 - Exchange Policy Memory Model Classes	
Element Name	**Attributes**
	The stereotype for the participant.
MetaIESSemanticElement	metaAttributeList (type:) [1..1]: List of attributes contained in the filtered semantic element. metaAttributeList (type: MetaIESAttribute) [0..*]: metaIESTagList (type:) [1..1]: List of parameters (/tag values) in an exchange specification. metaIESTagList (type: MetaIESTag) [0..*]: metaIESTransactionalList (type:) [1..1]: List of filtered transactional elements included in the filtered semantic element. metaIESTransactionalList (type: MetaIESTransactional) [0..*]: Modelid (type:) [1..1]: UID of the tag within the meta-model. name (type:) [1..1]: Name of the filtered semantic element. referencedSemanticIdentifier (type:) [1..1]: The filtered semantic element references the semantic element's identifier (primary key) referencedSemanticName (type:) [1..1]: The filtered semantic element references the name of the semantic element. stereotype (type:) [1..1]: The stereotype of the filtered semantic element.
MetaIESTag	Definition of a parameter (/tag value) of an IES element. availableValues (type:) [1..1]: The allowable values for the property. Modelid (type:) [1..1]: UID of the tag within the meta-model. name (type:) [1..1]: The name of the property. value (type:) [1..1]: The value of the property.
MetaIESTransactional	Definition of a filtered transactional in an IES construct. metaEnumerations (type:) [1..1]: List of enumerations (permitted values) used by the filtered transactions. metaIESFilterList (type: MetaFilter) [0..*]:

Table 71 - Exchange Policy Memory Model Classes	
Element Name	**Attributes**
	metaIESFilterList (type:) [1..1]: List of filters assigned to the filtered transactional element. metaTagList (type: MetaIESTag) [0..*]: metaTagList (type:) [1..1]: List of parameters (/tag values) assigned to the filtered transactional element. Modeld (type:) [1..1]: UID of the tag within the meta model. name (type:) [1..1]: Name of the transactional element. stereotype (type:) [1..1]: Stereotype of the filtered transactional element.

Annex A.3 – PPS Policy XSD (Informational)

The IEF-RA provides an XML PSM for the semantic and exchange policy exchange files used during operational testing between 2018 and 2023. The policy files were generated from IEPPV models created for each of the data domains used during operational testing. The policy files were ingested by the PPS and enforced during operational.

The XSDs are considered informational as they will evolve with the Packaging and Processing Service (PPS), which is the normative sources for the DCS policies it enforces. The XSDs for the semantic and exchange policies are provided in “mars-2025-02-06 PPS Policy Schema (Informational).zip”.

Annex A.4 Enumerations (Informational)

The following clauses describe the data enumerations used in this specification. They represent a foundational set of elements a user may extend 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).

Annex A.4.1 - Command Type Enumerations

The following figure identifies the Command Type Enumerations used in the IEF RA Specification.

`

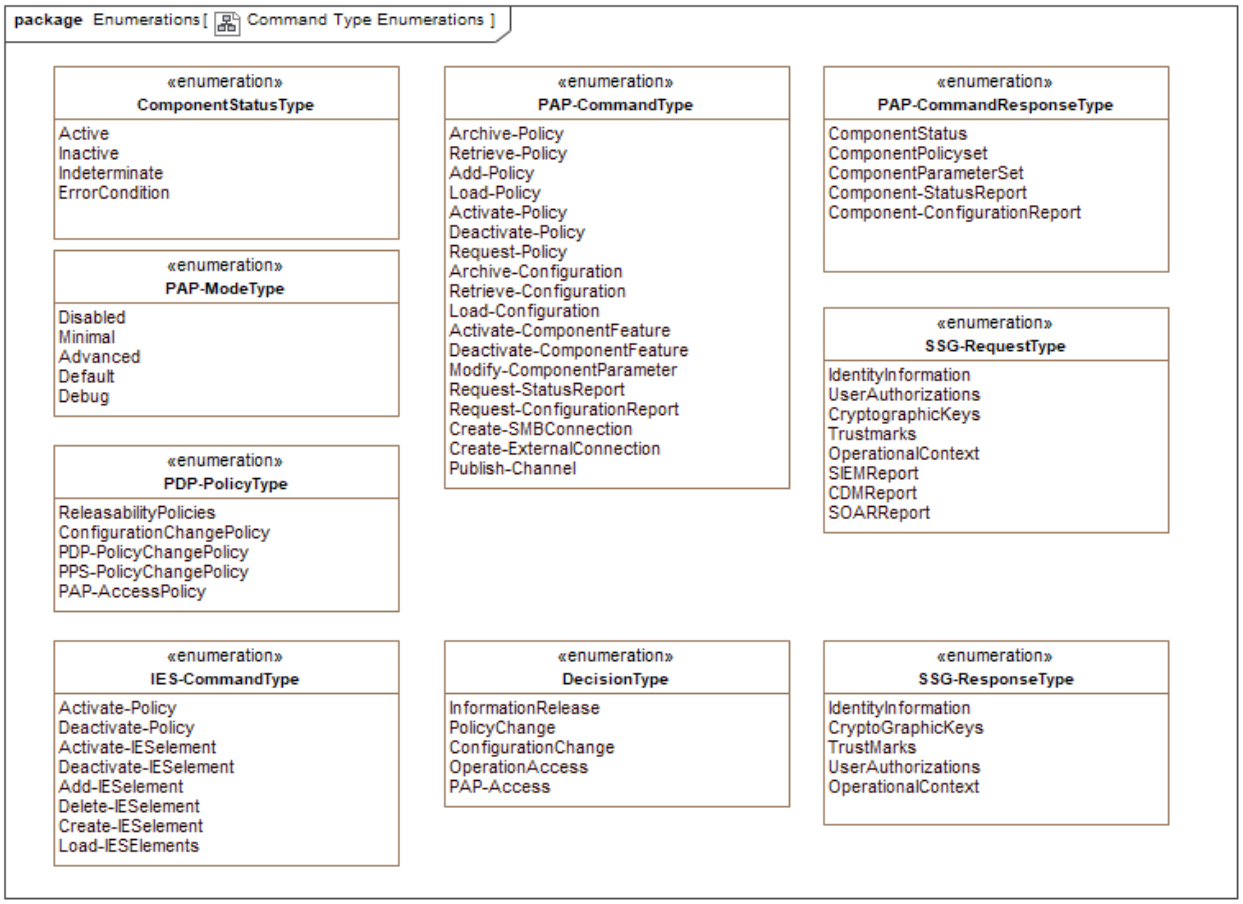


Figure 108 -Command Type Enumerations

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

Table 72 - Command Type Enumerations	
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.

Table 72 - Command Type Enumerations	
Type Name	**Values**
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 information element 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.

`

Table 72 - Command Type Enumerations	
Type Name	**Values**
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 the PAP issues to an IEF Component. Activate-ComponentFeature: Direct an IEF Component to activate one or more of its features. Activate-Policy: Direct 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. Add-Policy: Direct an IEF Component (e.g., PDP or PPS) to add the policies in the message to its policy environment. These policies are held in a temporary processing area until the user (/administrator) directs the component to load them. Archive-Configuration: Direct an IEF component to persist its current operating configuration to a specified location. Archive-Policy: Direct an IEF Component (i.e., PDP and PPS) to package (aggregate and format) its current policy environment and post the data as a file to the environment archive. Create-ExternalConnection: Direct a Messaging-PEP to create a connection with the user-specified messaging infrastructure. Create-SMBConnection: Direct an IEF component to establish and register a connection to the SMB.

Table 72 - Command Type Enumerations	
Type Name	**Values**
	Deactivate-ComponentFeature: A message from a PAP that directs an IEF Component to deactivate one or more component features. Deactivate-Policy: Direct 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. Load-Configuration: Directs an IEF component to load a set of operating parameters and hold them in its processing area to reset its operational settings. Load-Policy: Direct a PDP or a PPS to load one or more policies from its processing area to its policy environment. Modify-ComponentParameter: Direct an IEF component to change the value of one or more configuration parameters. Publish-Channel: Directs a PEP to publish information on one or more channels the component uses to share data. Request-ConfigurationReport: Direct an IEF component to report its configuration. Request-Policy: Direct an IEF component (i.e., PPS or PDP) to provide information on one or more policies. Request-StatusReport: Direct an IEF Component to report the current operating status of one or more of its features to the PAP. Retrieve-Configuration: Direct an IEF component to get a configuration file (or archive) from a specified location in the IEF environment. The component will extract the configuration file from the SAC, ingest the configuration policies environment, and wait for the instructions to activate the configuration. Retrieve-Policy: Direct an IEF component (i.e., PPS and PDP) to get one or more policies from a specified file stored at a specified location in the environment. The component will extract the policies, ingest the policies, and wait for the instructions to activate the policies.

`

Table 72 - Command Type Enumerations	
Type Name	**Values**
PAP-ModeType	Defines the modes of operation for a PAP. Advanced: This mode enables complete operational control governed by policy and attributes. It is available where administrators (operators) do not have security clearance equal to or higher than the highest sensitivity of information in the environment. Debug: This mode provides features to identify operational end technical issues with the PAP and IEF components. It is limited to test and development operations. In this mode, the operator can override policy. Default: Provide the operator with a standard set of functions to manage and administer an IEF environment. What an administrator has access to depends on organizational policy, operating procedures, and guidelines. Disabled: The PAP operator interface is turned off or disabled. 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 adjudicating 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.

Table 72 - Command Type Enumerations	
Type Name	**Values**
SSG-RequestType	Identifies allowable types of SSG data requests. CDMReport: Request the release of a report to the CDM System. CryptographicKeys: Request a new or retrieve a cryptographic key from the user-specified key management services. IdentityInformation: Request Identity information for specified users. OperationalContext: Message conveying data about the operating/mission context in which the IEF operates. SIEMReport: Request the release of a report to the SIEM System. SOARReport: Request the release of a report to the SOAR System. Trustmarks: Request TrustMarks for specified users from the user-specified registry. UserAuthorizations: Request Attributes (/privileges/attributes) for specified users.
SSG-ResponseType	The type of response being provided by the SSG to an IEF component. CryptoGraphicKeys: Message conveying cryptographic keys and tokens for an information element. 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: A message conveying TrustMark Information for an external recipient of information elements. UserAuthorizations: Message conveying a sender's or Recipients' attributes, attributes, access rights, or privileges.

Annex A.4.2 - Component Enumerations

The following figure identifies the Component Enumerations used in the IEF RA Specification.

`

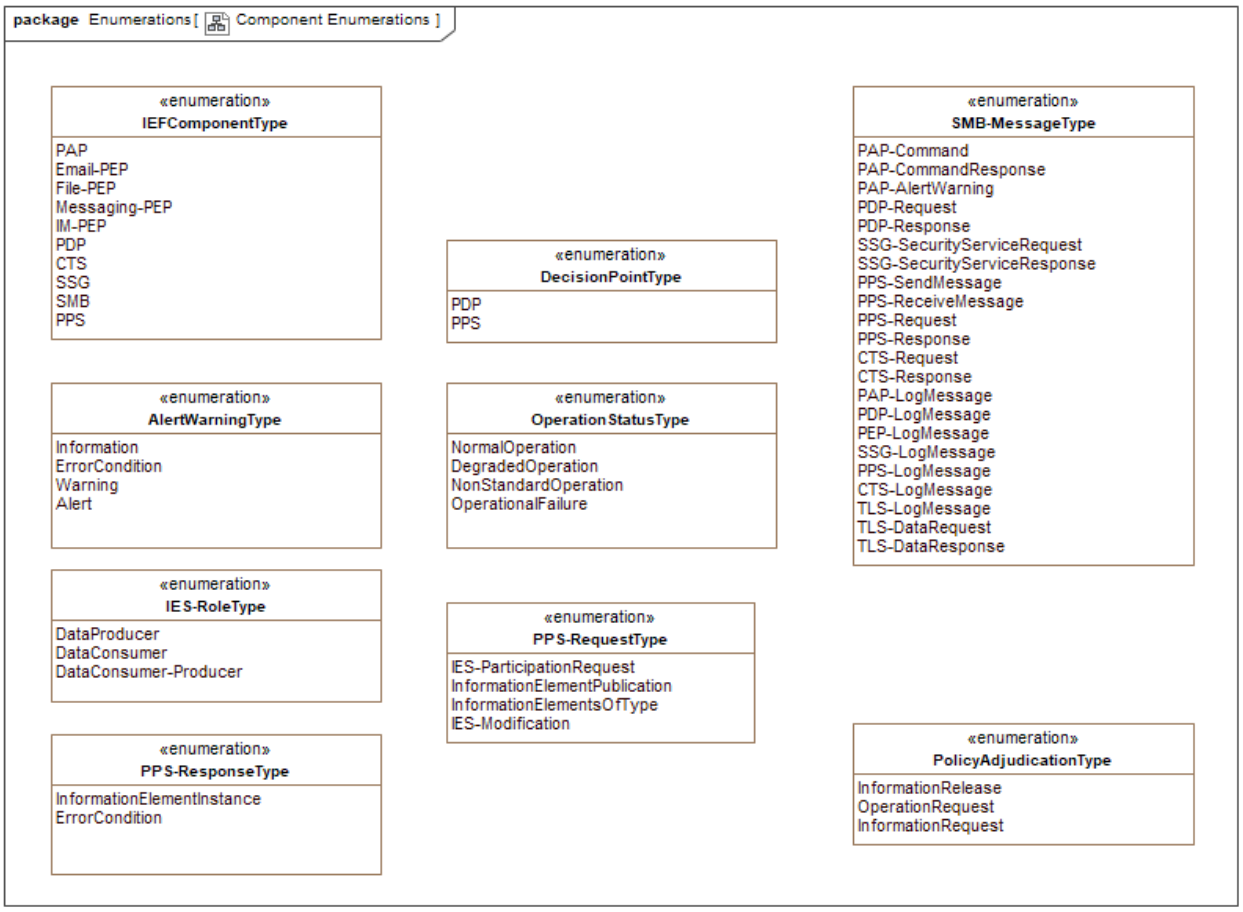


Figure 109 -Component Enumerations

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

Table 73 - Component Enumerations	
Type Name	**Values**
AlertWarningType	Identifies the types of messages passed to the PAP through the alerts and warnings channel. Alert: 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:

Table 73 - Component Enumerations	
Type Name	**Values**
	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 adjudicating 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 information element 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) encrypts and decrypts data and 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 , on an IEF-protected environment, is authorized to send the content embedded in the email. The PEP will also ensure the user appropriately marks emails sent in an IEF-protected environment. File-PEP: Policy Enforcement Point (PEP) provides a proxy that intercepts file-based operations and assures that files are handled following user-specified policy. The PEP also ensures that the users appropriately mark the files and that the content is always encrypted at rest and in transit. IM-PEP:

`

Table 73 - Component Enumerations	
Type Name	**Values**
	Policy Enforcement Point (PEP) provides a proxy that intercepts text messages between the Instant Messaging (IM) client and IM server to determine the following: • If the specified recipients are authorized to receive the content of the message and • If the sender, in an IEF-protected environment, is authorized to send the content of the message (based on user or chat room-included markings). 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, 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: • Process (parses and transforms) DataElements so they can be marshaled to the specified data store and • Package (aggregates, transforms, marks, redacts, structures, and formats) an information element tailored to the recipient's authorization. SMB: The IEF Secure Messaging Bus (SMB) provides the communication pathways between IEF Components. SSG: IEF Security Services Gateway (SSG) provides the features to integrate the IEF environment with user-specified security services and infrastructure.

Table 73 - Component Enumerations	
Type Name	**Values**
IES-RoleType	Identifies the role of a PPS within a specific Information Exchange Specification (Agreement). DataConsumer: Indicates that the connection is a receiver (e.g., consumer, recipient, or reader) of information from the community. DataConsumer-Producer: Indicates that the connection is a receiver (e.g., consumer, recipient, or reader) of information from the provider (/publisher, producer/writer) into the community. DataProducer: Indicates that the connection is a provider (e.g., publisher, producer, or writer) of information into the community.
OperationStatusType	Identifies the operating states for an IEF component or component feature. DegradedOperation: One or more specified features are not performing as expected. NonStandardOperation: An information or user operation has caused a nonstandard,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 information element 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.

`

Table 73 - Component Enumerations	
Type Name	**Values**
PPS-RequestType	Identifies the type of request being made to a PPS. IES-Modification: A user requests a change to an Information Exchange contract (SpecificationJ). IES-ParticipationRequest: A user requests participation in an existing Information Exchange contract (SpecificationJ). InformationElementPublication: Request a single instance of an information element. InformationElementsOfType: Request a list of information elements. The list is issued with the Name, Identifier, and sensitivity of the information element.
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.
SMB-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 information element be encrypted or decrypted. CTS-Response: CTS messages the PEP with the transformed information element following the encryption or decryption. PAP-AlertWarning: IEF Component message to the PAP (and the TLS) to inform the PAP (/ user /administrator) that unauthorized requests are 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.

Table 73 - Component Enumerations	
Type Name	**Values**
	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: A PDP message to an IEF Component determines whether the requested action has been authorized. PEP-LogMessage: PEP log message to the TLS. PPS-LogMessage: PPS log message to the TLS. PPS-ReceiveMessage: A PEP message to the PPS conveys information to be processed by the PEP. PPS-Request: PEP message to the PPS requesting information. PPS-Response: A PEP message was issued in response to a request for information. PPS-SendMessage: A PEP message was issued in response to a request for information. SSG-LogMessage: SSG log message to the TLS. SSG-SecurityServiceRequest: The IEF component requests information from the user's security infrastructure (e.g., User Privileges, cryptographic keys, and operational context). This request is made to the SSG, which provides a single integration point for user-specified security services. SSG-SecurityServiceResponse: SSG response to an IEF component, with the information requested from the user's security infrastructure. TLS-DataRequest: Message to the TLS Requesting Data. TLS-DataResponse: Message from the TLS to a Monitoring or logging service containing log reports for a specified time period for a specified component. TLS-LogMessage:

`

Table 73 - Component Enumerations	
Type Name	**Values**
	Message to the Trusted Logging Service.

Annex A.4.3 - Instruction Type Enumerations

The following figure identifies the Instruction Type Enumerations used in the IEF RA Specification.

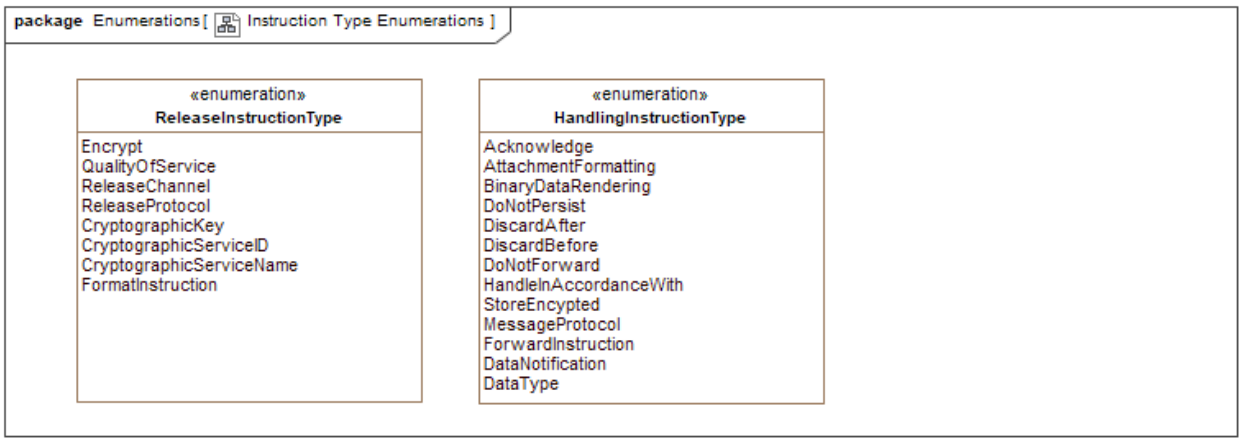


Figure 110 -Instruction Type Enumerations

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

Table 74 - Instruction Type Enumerations	
Type Name	**Values**
HandlingInstructionType	Valid *HandlingInstruction* types for the recipient of an *information element*. Acknowledge: An instruction to the recipient of an information exchange directing the issuance of an acknowledgment of the receipt of the information to the provider. 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.

Table 74 - Instruction Type Enumerations	
Type Name	**Values**
	DataNotification: An instruction to recipients specifying the type of data being received. (e.g., operational, test, experimentation, exercise, or training). DataType: An instruction to recipients specifying the type of data being received. (e.g., operational, test, experimentation, exercise, or training). DiscardAfter: Instructions to the recipient of an information exchange specify the rules for destroying or discarding data in an information package or message. This version directs the recipient to discard the information after a specified date and time. DiscardBefore: Instructions to the recipient of an information exchange specify the rules for destroying or discarding data in 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: Instructions to the recipient of an information exchange direct the recipient not to persist in any information or data in a payload or message. ForwardInstruction: Instructions to the recipient of an information exchange are to forward the information to authorized recipients by any provided list or specified information-sharing agreements. HandleInAccordanceWith: Instructions to the recipient of an information exchange direct the recipient to handle the information in the message per the instructions in a specified document. MessageProtocol: Instruction to the recipient of an information exchange that identifies the information element formatting protocol (e.g., XSD). StoreEncypted: Instructions to the recipient of an information exchange direct the recipient to encrypt the information in the message before storing it.

`

Table 74 - Instruction Type Enumerations	
Type Name	**Values**
ReleaseInstructionType	Release InstructionTypes. 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 a unique identifier. CryptographicServiceName: Specifies the name of the cryptographic service to be used. 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 before release. FormatInstruction: Specifies formatting instructions 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 releasing the information element(s). ReleaseProtocol: Specifies the messaging protocol to use when sending the information elements.

Annex A.4.4 - Policy Enumerations

The following figure identifies the Policy Enumerations used in the IEF RA Specification.

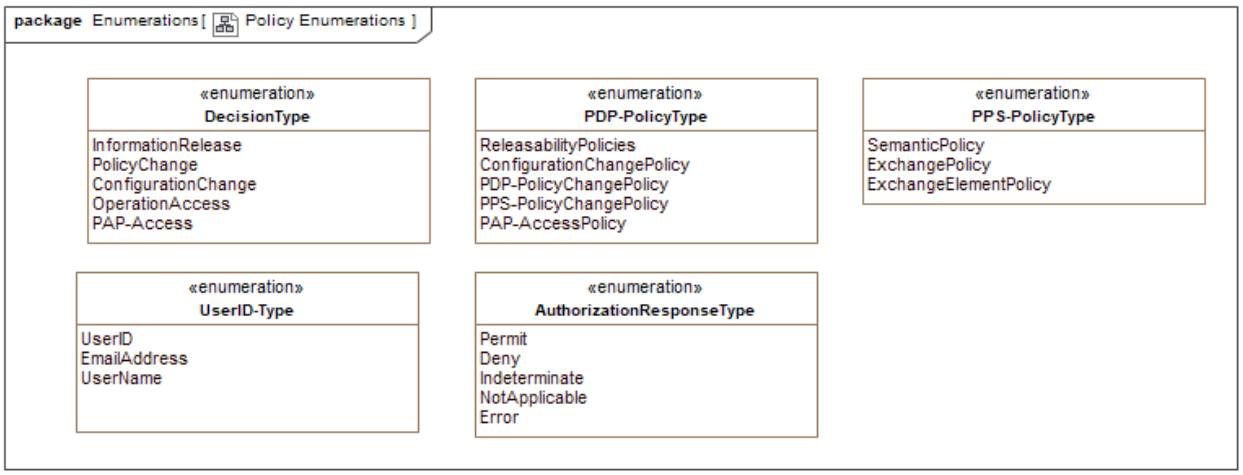


Figure 111 -Policy Enumerations

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

Table 75 - Policy Enumerations	
Type Name	**Values**
AuthorizationResponseType	Identifies the type of response being provided by the PDP. This result s from the policy adjudication process based on the user's submitted policy request 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 the PDP encounters an error (e.g., the policy database is off-line), 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: The 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 information element 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:

`

Table 75 - Policy Enumerations	
Type Name	**Values**
	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 adjudicating 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 policies 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. This includes the assembly of both transactional and semantic elements described in the IEPPV Specification.

Table 75 - Policy Enumerations	
Type Name	**Values**
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.

Annex A.4.5 - Security Enumerations

The following figure identifies the Security Enumerations used in the IEF RA Specification.

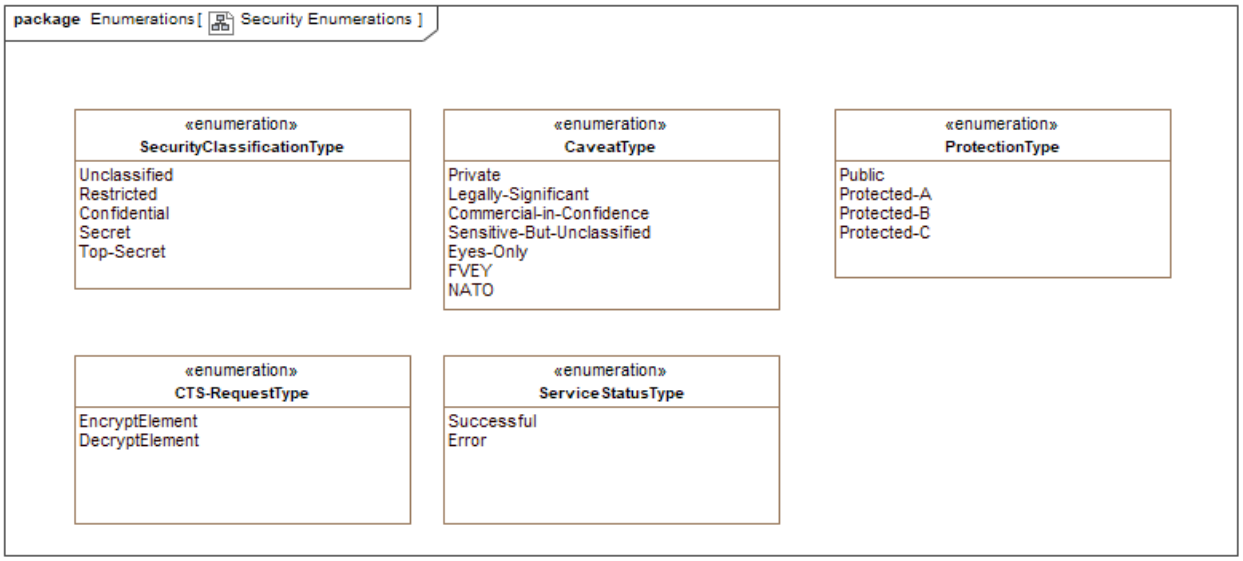


Figure 112 -Security Enumerations

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

`

Table 76 - Security Enumerations	
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 information element or Message. The identified caveats form a small subset of those used by the Military, National and Public Security, and the Legal community. This list is meant to be extended. Commercial-in-Confidence: The information contains proprietary elements (e.g., intellectual property or trade secrets) for the data owner/creator and cannot be released without their approval. Eyes-Only: Information is meant to be seen only by the person to whom it is directed. FVEY: Classified information is restricted to the Five Eyes Community. Five Eyes, often abbreviated as FVEY, is an intelligence alliance comprising Australia, Canada, New Zealand, the United Kingdom, and the United States. Legally-Significant: The information contains elements pertinent to a legal proceeding and cannot be released without special approvals. NATO: Classified information is 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 information element to its original form so authorized parties (with the decryption key) can read it. EncryptElement: Request that the CTS encrypt (encode) the specified information element so that only authorized parties (with the decryption key) can read it.

Table 76 - Security Enumerations	
Type Name	**Values**
ProtectionType	Identifies the protection levels for sensitive but unclassified (SBU) information. Protected Information is sensitive information that 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, or hurt or harm a person, corporation, or government. Examples include personal information such as pay data, medical records, and business information such as trade secrets. Protected-A: This applies to information that, if compromised, could reasonably be expected to cause injury or embarrassment outside the national interest, such as the disclosure of an exact salary figure, an individual's date of birth, or even information about to contracts and tenders. Protected-B: This 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 include Medical, criminal, or psychiatric records, trade secrets, and risk assessments. Protected-C: This 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: Information that is releasable to the public.
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: Such material's unauthorized or public release would cause "undesirable effects" on 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:

`

Table 76 - Security Enumerations	
Type Name	**Values**
	Information elements that lack the sensitivity to hold a classification listed above.
ServiceStatusType	Identifies valid service response status types. Error: The request was unsuccessful - an error message was attached. Successful: The request was successful - data attached.

Annex A.4.6 - Security Service Enumerations

The following figure identifies the Security Service Enumerations used in the IEF RA Specification.

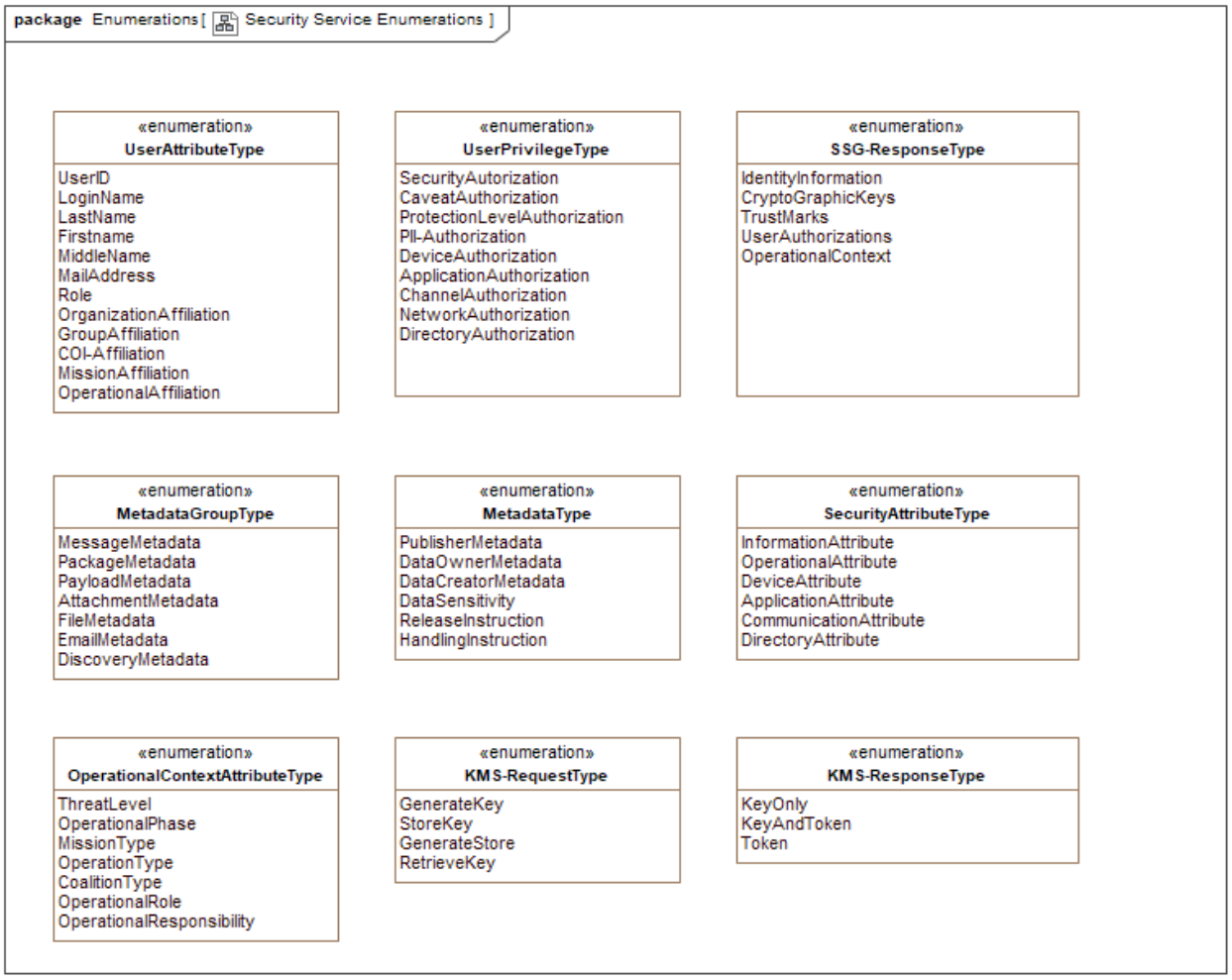


Figure 113 -Security Service Enumerations

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

Table 77 - Security Service Enumerations	
Type Name	**Values**
KMS-RequestType	The type of request being made to the user Key Management Service(s). GenerateKey: A 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. GenerateStore: A 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:

`

Table 77 - Security Service Enumerations	
Type Name	**Values**
	A request to the user-specified Key Management Services to retrieve an existing key from the key escrow system. StoreKey: A 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: A set of metadata elements is to be bound to a message attachment. DiscoveryMetadata: Set of metadata elements that aid in the discovery of an information element. EmailMetadata: A set of metadata elements is 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: A set of metadata elements is to be bound to a message package. PayloadMetadata: A set of metadata elements is to be bound to a message payload.
MetadataType	Identifies the role of metadata in terms of the information element it describes. DataCreatorMetadata: Data describing the creator of the information element. DataOwnerMetadata: Data describing the owner of the information element.

Table 77 - Security Service Enumerations	
Type Name	**Values**
	DataSensitivity: MetaData describes the information element's sensitivity (e.g., Privacy, confidentiality, classification, or legal significance). HandlingInstruction: Data describing the specific handling instructions for the information element. PublisherMetadata: Data describing the publisher of the information element. ReleaseInstruction: Data describing the specific release instructions for the information element.
OperationalContextAttributeType	Data type provides 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/attributes) being sought. ApplicationAttribute: Attributes about application access and constraints. CommunicationAttribute: Attributes about communications access and constraints. DeviceAttribute: Attributes about device access and constraints. DirectoryAttribute: Attributes about directory access and constraints.

`

Table 77 - Security Service Enumerations	
Type Name	**Values**
	InformationAttribute: Attributes about information element access. OperationalAttribute: Attributes about IEF operation/function access primarily pertain to administration functions.
SSG-ResponseType	The type of response being provided by the SSG to an IEF component. CryptoGraphicKeys: Message conveying cryptographic keys and tokens for an information element. 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: A message conveying TrustMark Information for an external recipient of information elements. UserAuthorizations: Message conveying a sender's or Recipients' attributes, attributes, access rights, or privileges.
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: The user's email address for the electronic mailbox attribute follows the syntax specified in RFC 822.

Table 77 - Security Service Enumerations	
Type Name	**Values**
	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: The user's customary function (s) for the organization or community is to connect the user to known behaviors, rights, obligations, beliefs, and norms. UserID: The user's network identification is for inter- and intra-organization 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 is required 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:

`

Table 77 - Security Service Enumerations	
Type Name	**Values**
	Authorization to access or receive information marked at that security level.

Annex A4 - Security Service Enumerations

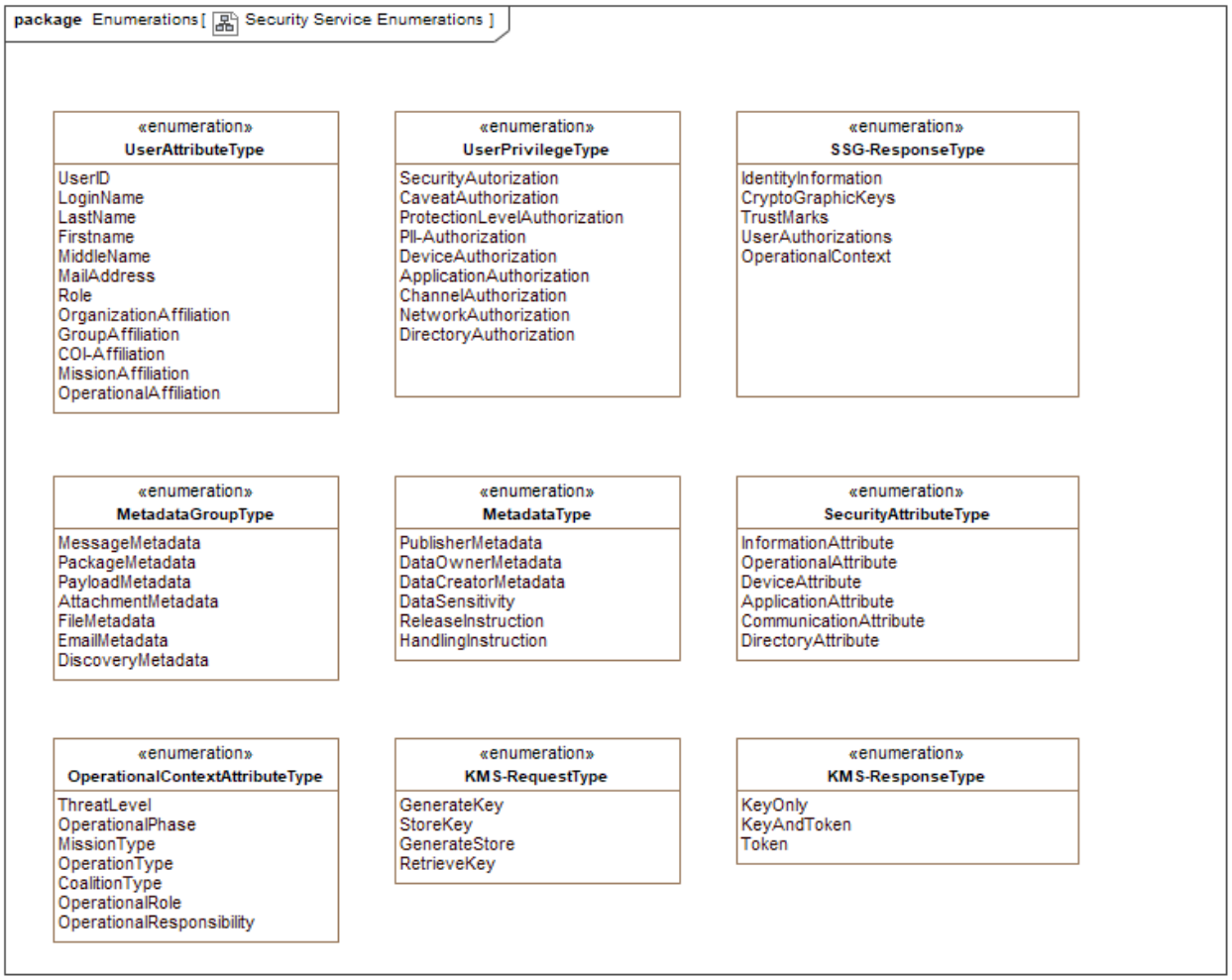


Figure 114 -Security Service Enumerations

The following table describes the message elements illustrated in the previous figure - Security Service Enumerations.

Table 78 - Security Service Enumerations Classes	
Element Name	**Attributes**

`

This page intentionally left blank.

Annex B – Minimum Metadata (informational)

The following table describes the minimum metadata needed to deliver secure data interoperability. This metadata is bound to the data objects within the message header, metadata element, or manifest – see Annex C. This set of metadata elements were required during 2023 - 2024 testing.

#	Metadata Name	Description
1	Security:	
	ConfidentialityLabel	Security or confidentiality labels for data objects.
	Security Level	Security Level for the data content.
	Releasability	Released data is authorized to (List of Recipients).
	Policy	The national or organizational SPIF.
	MetadataConfidentiality Label	Security or confidentiality labels for data object metadata.
	Security Level	Security Level for the metadata content.
	Releasability	Released metadata is authorized to (List of Recipients).
	Policy	The national or organizational SPIF.
2	Provenance	
	Creator	Creator of the Data.
	Creation Date/Time	The Date that data was created.
	Publisher	The source of the data object.
3	Identification	
	Title	Title of the data object.
	Description	Brief description of the data object.
	Identifier	Universally Unique Identifier of the data object.
4	Distribution	
	ResourceURL	URL of the data object at rest.
	SourceURL	URL to the native data object at rest.
	SourceFormat	Free text describing the format of Native Data.
5	Search	
	Object Type	The type data
	Geospatial	Geographical metadata (country, region, place, and/or coordinate(s)) & encoding
6	Handling	
	HandlingInstruction	Free text instructing release processors or recipients

`

Annex C – Metadata Bindings (Informational)

The following sections outline several methods for binding metadata to data objects. The binding pattern comes from existing or evolving standards (e.g., ADatP-4778, ZTDF, and ACP-240). See these references for additional information. This Annex is identified as information as the reference standards supersede the information provided in this Annex.

Annex C.1 - Message Header

The NATO Standardization Agreement (STANAG) is a document that specifies the agreement of member countries to implement a standard.

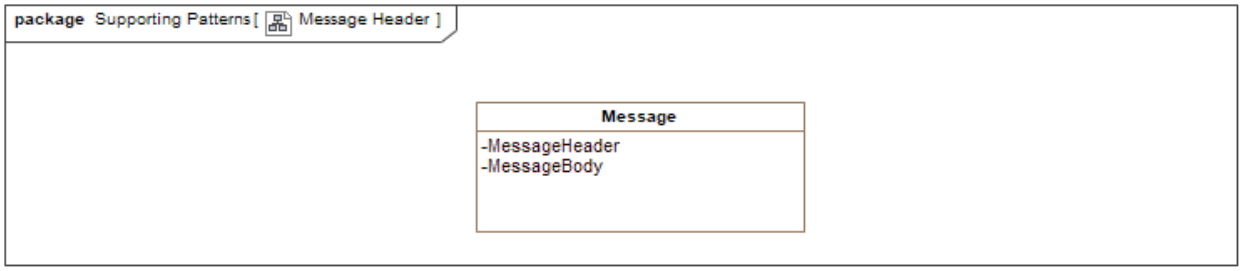


Figure 115 -Message Header

The following table describes the message elements illustrated in the previous figure - Message Header.

Table 79 - Message Header Classes	
Element Name	**Attributes**
Message	A message is a semi-structured data object composed of data attributes or attribute groups ordered in a specified sequence, each attribute or group characterized by an identifier (name) and data values. It is designed to facilitate automated handling and processing. MessageBody (type:) [1..1]: The message body contains data attributes or attribute groups ordered in a specified sequence of data elements and attributes. MessageHeader (type:) [1..1]: The message header contains data attributes or attribute groups ordered in a specified sequence to provide a list of technical or operational details about the message, such as who sent it, the standard and version applied, the confidentiality or sensitivity about the content, the software used to compose it, and the systems and servers that it passed through on its way to the recipient. See Annex C for a minimum set of metadata attributes.

`

Annex C.2 - SecureAssetContainer

The NATO Standardization Agreement (STANAG) is a document that specifies the agreement of member countries to implement a standard.

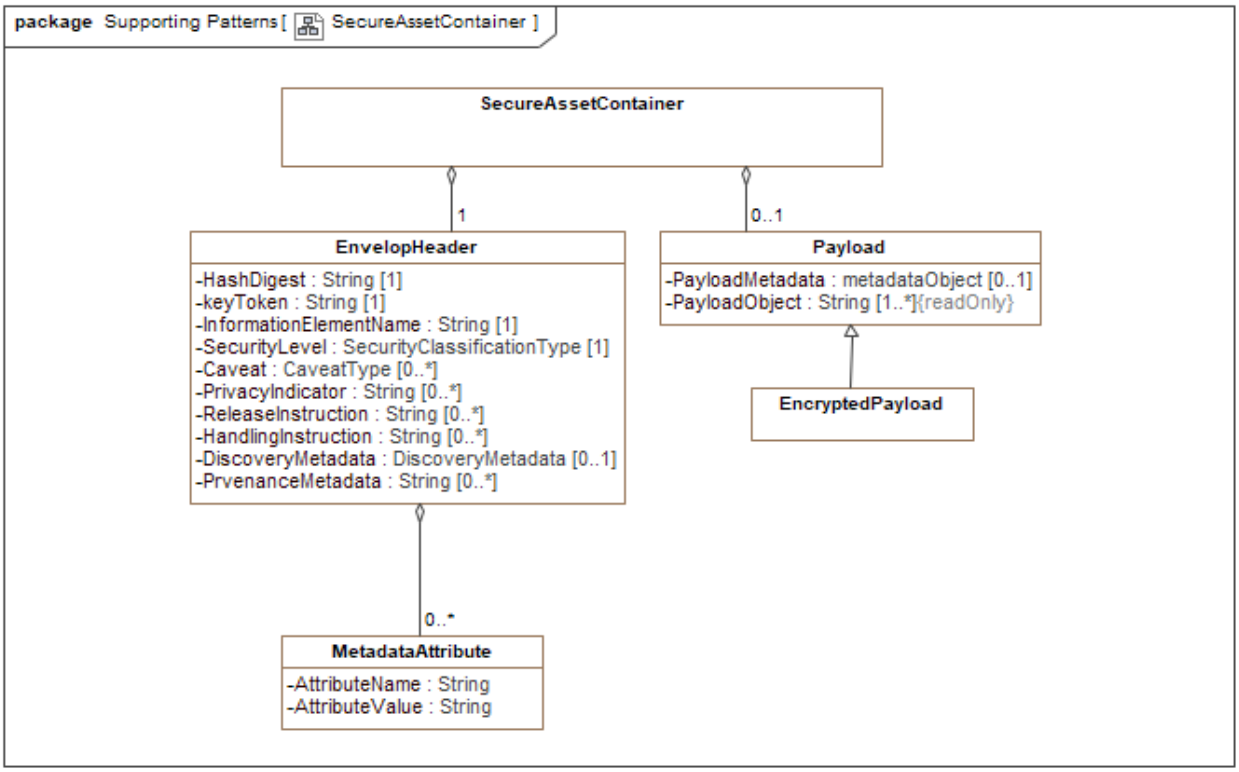


Figure 116 -SecureAssetContainer

The following table describes the message elements illustrated in the previous figure - SecureAssetContainer.

Table 80 - SecureAssetContainer Classes	
Element Name	**Attributes**
EncryptedPayload	Identifies that the container element holds an encrypted information element or that the information element must be encrypted within the container.
EnvelopHeader	This file (e.g., text file) holds the metadata describing the data payload in the SAC. Caveat (type: CaveatType) [0..*]: List field populated with a comma-separated list of Caveat markings for the information element. DiscoveryMetadata (type: DiscoveryMetadata) [0..1]: Metadata elements that facilitate the discovery of information. HandlingInstruction (type: String) [0..*]:

Table 80 - SecureAssetContainer Classes	
Element Name	**Attributes**
	A string describing an instruction for using, processing, or storing the information element. HashDigest (type: String) [1..1]: All Secure Asset Containers (SAC) include a hash digest calculated at creation time to detect container tampering. 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 SACs, the digest is calculated as follows: • The contents of the encrypted information element; • The SecurityLevel for the information element; • The caveats for the information element; • PrivacyIndicators for the information element; • Special Handling instructions for the information element; • 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 left to the user. All Secure Asset Containers (SAC) include a digest calculated at creation time to detect container tampering. 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 key token is extracted. This is then used to retrieve the key from the KMS. PrivacyIndicator (type: String) [0..*]: This list field is populated with a comma-separated list of privacy markings (e.g., PII) for the information element. PrvenanceMetadata (type: String) [0..*]: ReleaseInstruction (type: String) [0..*]: This string describes a constraint on the release of the information element. SecurityLevel (type: SecurityClassificationType) [1..1]: The list field is populated with a comma-separated list of security markings for the information element.
MetadataAttribute	Name-value pair of the metadata attribute. See minimum metadata elements in Annex C.

`

Table 80 - SecureAssetContainer Classes	
Element Name	**Attributes**
	AttributeName (type: String) [1..1]: Name of the metadata attribute. AttributeValue (type: String) [1..1]: Value of the attribute.
Payload	Data or information objects are formatted according to the specified exchange protocol (e.g., XSD) and bound to the required metadata. Payload formatting, metadata requirements, and binding profiles are derived from the user-specified data exchange or storage standards. PayloadMetadata (type: metadataObject) [0..1]: PayloadObject (type: String) [1..*]:
SecureAssetContainer	The Secure Access container (SAC) is an information element containing the required set of metadata elements in a data file (e.g., text file) and either an encrypted or non-encrypted payload. The SAC is a container (e.g., ZIP File) that binds the metadata to the data, enabling the PEP to access the metadata whether or not the payload is encrypted. The SAC contains one of the payloads or encrypted payloads.

Annex C.3 - Trusted Data Object

The NATO Standardization Agreement (STANAG) is a document that specifies the agreement of member countries to implement a standard.

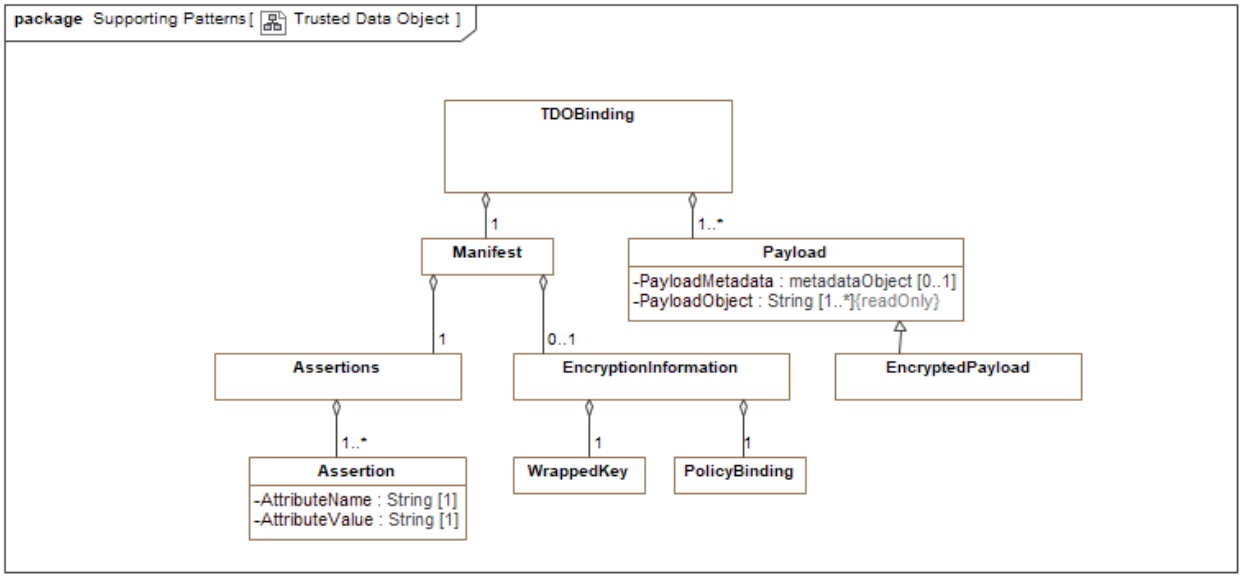


Figure 117 -Trusted Data Object

The following table describes the message elements illustrated in the previous figure - Trusted Data Object.

Table 81 - Trusted Data Object Classes	
Element Name	**Attributes**
Assertion	Name-value pair for an assertion. AttributeName (type: String) [1..1]: Name of the assertion attribute. AttributeValue (type: String) [1..1]: Value of the assertion attribute.
Assertions	A set of metadata attributes and values associated with the data object (/payload) or payloads.
EncryptedPayload	Identifies that the container element holds an encrypted information element or that the information element must be encrypted within the container.
EncryptionInformation	Data is needed by an authorized system or user application to decrypt the data payloads.
Manifest	The manifest includes the data object metadata or assertions, encryption information, iHash-based Message Authentication Code (HMAC), 'wrapped' encryption keys, and a signed policy.
Payload	Data or information objects are formatted according to the specified exchange protocol (e.g., XSD) and bound to the required metadata. Payload formatting, metadata requirements, and binding profiles are derived from the user-specified data exchange or storage standards. PayloadMetadata (type: metadataObject) [0..1]: PayloadObject (type: String) [1..*]:
PolicyBinding	Identifies the crypto-binding used to bind metadata within the DCS (/data) objects. For additional information about the binding policy, see references ACP240 or ZTDF.
TDOBinding	A data encoding that enables data tagging and cryptographic security features to be applied to one or more data payloads. See ZTDF and ACP240 references for additional information.
WrappedKey	This decryption key can only be decrypted if the receiving system or user application has the private key. Utilizing asymmetric cryptography requires a Public Key Infrastructure (PKI); currently, within the Allied community, there is no central PKI root. For additional information about wrapped keys, see references to ACP240 or ZTDF.

This page intentionally left blank.

Annex D – Glossary (Informational)

The following elements define the meaning of terms and acronyms used in this specification. The definitions reflect the meaning of the terms within the contexts of information sharing and safeguarding and Data-Centric Security.

Table 82 - Terms and Definitions

Term	Definition
ABAC	Attribute-Based Access Control
Accurate	Free from error or defect; consistent with a standard, rule, or model; precise; exact.
Acknowledge	An instruction directing the recipient of an information exchange to acknowledge the receipt of the information.
ActionInstruction	An instruction directing the producer or receiver of information to take a specific action: 1. Rules (e.g., encrypt payload) governing the release of the information or 2. Rules (e.g., do not forward and do not store) that the receiver must enforce upon receipt of the information.
Active Policy	The set of policies (/rules) that are instantiated and set "active" for one or more environment policy enforcement and decision points.
Adaptive Information Sharing	The ability to selectively share data with recipients based on their needs, attributes, and existing operational or business context (e.g., roles, relationships, risks, threats, severity, scale, and trust).
Aggregation	Defines how data elements are combined to referentially and semantically complete data sets.
API	Application Program Interface
Attachment	A computer or electronic file (typically unstructured) sent with or included in a message, email, or instant message.
AttachmentElement	A binary file (e.g., PDF file, image, or video) or document, and information about the binary or document, such as the size, type, and description.
AttachmentSpecification	A specification of the rules governing the attachment of binary information elements to an information exchange or message.
AttachmentSummary	A summary or list of attachments for a specific data package.
Attribute	A defined property of an entity, object, triple, or schema Source: A Dictionary of Computing. Oxford University Press, 2008. Oxford Reference Online. Oxford University Press
bcc	Blind Copy
C2	Command and Control
C3	Consultation, Command and Control

`

Table 82 - Terms and Definitions	
Term	Definition
C4I	Command, Control, Computers, Communications and Intelligence
C4ISR	Command, Control, Computers, Communications Intelligence Surveillance and Reconnaissance
Caveat	Markings (e.g., meta-data, tags, or labels) specifying a restriction or warning order on a specific data/information element or operating environment.
Caveat Separation	The process for selectively exchanging data/information is based on the decisions made on the recipient's attributes of the recipient(s), security policy, and the sensitivity of the information.
CDM	Continuous Diagnostics and Mitigation
Challenged Networks or Communication	Under operational conditions, most front-line communications employ radio (e.g., HF, VHF, or HCDR). These forms of communication are inherently less robust than the Wi-Fi and wired networks realized by most organizations. Challenged refers to the reality that data communication over these networks: • Have limited bandwidth capability (as low as 1Kb/Sec); • Sustain periodic outages due to range limitations, jamming, and voice overrides; • Have Large node counts and • Have significant Packet loss.
Classified Information	Sensitive information or digital assets that, if compromised, could reasonably be expected to cause injury to the national interest, defense, and maintenance of a nation's social and economic stability.
Common Operating Picture	A collection of processes, tools, and technologies that provide the users with a shared understanding of the operational environment. The COP integrates perspectives, delivers actionable knowledge, and structures data/information for specific user needs. Information presented may include threats, opportunities, objectives, resources, organization structure, activities, situational awareness, and other relevant information.
Common Representational Operating Picture	The CROP is equivalent to the COP but limits access to the information required to exercise a specific role or function for the user role or function.
Communication Channel	A means of communication, access, or data exchange.
Community	A community of interest or community of practice.
Community of Interest	A collaborative group of users that share or exchange information to pursue its shared goals, interests, missions, or business processes must have a shared vocabulary for the information exchanges.
Community of Practice	An informal, self-organized network of peers with diverse skills and experience in an area of practice or profession.
Conceptual Interoperability	The assumptions and constraints of the meaningful abstraction of reality are aligned to reach the highest level of interoperability. This level of interoperability requires that conceptual models are documented based on engineering methods, enabling their interpretation and evaluation by other engineers. In essence, this requires a "fully specified, but implementation independent model," as

Table 82 - Terms and Definitions

Term	Definition
	requested by Davis and Anderson; this does not simply describe the conceptual idea. This level ensures that the underlying levels follow the same theory.
Confidential Information	This applies to information or digital assets that, if compromised, could harm the national interest, national security, public safety, defense, and maintenance of the nation's social, political, and economic stability.
Content-Centric	Refers to both information-centric and data-centric.
Continuous Diagnostics and Mitigation (CDM)	The Continuous Diagnostics and Mitigation (CDM) Program provides a dynamic approach to fortifying the cybersecurity of government networks and systems. The CDM Program delivers cybersecurity tools, integration services, and dashboards that help participating agencies improve their security posture by: • Reducing agency threat surface; • Increasing visibility into the federal cybersecurity posture; • Improving federal cybersecurity response capabilities and • Streamlining Federal Information Security Modernization Act (FISMA) reporting. (https://www.cisa.gov/resources-tools/programs/continuous-diagnostics-and-mitigation-cdm-program)
Contract	(Source: SOPES and UPDM) A contract represents Semantic construction rules and information flow controls that 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 are taken to diffuse crises, prevent their escalation into armed conflict, and contain the 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
CSP	Cloud Service Provider
CTS	The Cryptographic Transformation Service (CTS) is the IEF component that encrypts and decrypts InformationElements as authorized by policy.
Data	Facts or factoids used to calculate, analyze, or plan.
Data Composite	A data set resulting from the aggregation of data elements.

`

Table 82 - Terms and Definitions	
Term	Definition
Data Element	Representation of information (data) in a formalized manner suitable for communication, interpretation, or processing by humans or by automated means.
Data Integration	Combining two or more data elements from separate sources into a semantically and referentially complete information (or business) object.
Data Integrity	Compliance with the allowable types, ranges, or domain values for each data element (or attribute).
Data Ownership	Identification that the entity controls the data or information so 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 Payload	Actual data in a packet, message, or file minus all headers attached for transport and minus all descriptive meta-data.
Data Stewardship	Accountable for integrity and quality of data.
Data-Centric	Enforce policies/rules to govern individual data assets/holdings/content, often referring to metadata or tags included within an information asset.
Data-Centric Security	Data-centric security (DCS) is the application of practices, processes, procedures, standards, and technology that focus on protecting actual data (at rest, in use, and in transit) based on the sensitivity of its content versus focusing primarily on software, infrastructure, networks, and devices. DCS comprises layers of services designed to understand, govern, and secure sensitive data, whether on-premises or in the cloud.
DataCreatorMetadata	Metadata tags and markings that identify the creator of data or information elements.
DataOwnerMetadata	Tags and markings that identify the owner or steward of the data or information elements.
DataSensitivity	MetaData describes the sensitivity (Privacy, confidentiality, classification, or legal significance) of the information element.
DCS	Data-Centric Security
DDS	Data Distribution Services for Real-Time Systems
Deadline	A QoS attribute describing the latest acceptable time for the occurrence of certain events.
Decision Advantage	Enable decision-makers, based upon information advantage and situational understanding, to make effective and informed decisions more rapidly than their adversary, thereby allowing one to increase operations' pace, coherence, and effectiveness dramatically.
Decrypt	To convert encrypted text into equivalent plain text employing a crypto-system and its key or password.
Defense-in-Depth	(1) The coordinated application of security services to protect the integrity of enterprise assets and resources. The strategy employs the principle that it is more difficult for an adversary to defeat a complex and multi-layered defense than to penetrate a single barrier.

Table 82 - Terms and Definitions

Term	Definition
	(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 attributes of the publisher and each recipient.
DEM	Data Exchange Mechanism
DHS	Department of Homeland Security
Digest	A digest is an information structure, format, and syntax common to all communities. It allows systems to handle heterogeneous data without understanding the source's context and semantics as long as the entities relevant to the packaged data items are represented in the digest. The digest enables users to discover, link, map, etc., the information within the message.
DistributionSpecification	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 coalition that forms spontaneously, members may join and leave without warning. The nature of the relationships between participants may vary dramatically across the coalition and throughout its lifetime.
Dynamic Interoperability	As a system operates on data over time, its state will change, including the assumptions and constraints that affect its data interchange. If systems have attained Dynamic Interoperability, they can comprehend the state changes in the assumptions and constraints each makes over time.
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)
eISA	Electronic Information Sharing Agreement

`

Table 82 - Terms and Definitions

Term	Definition
Electronic Information Sharing Agreement	A machine-readable version of an information-sharing agreement.
Encrypt	The process for altering (encoded) data uses a mathematical algorithm to make it 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.
ERMA	ERMAs are real-time bills of materials (BOMs) designed to supply SRPNetOS real-time ER operational data for quality of service and security operations.
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 and type; for example, a user-created document, program data, image, video, or software program.
Filter	A profile or script containing the rules to restrict the assembly of data or information elements.
Gateway	An IEF component that connects the SMB and the users' network and infrastructure, the gateway provides a single integration point for security services hosted elsewhere in the user environment. It allows passing message traffic, security redaction, filtering, proxies, or protocol translations at various network layers.
HashDigest	All Secure Asset Containers (SAC) include a digest calculated at creation time to detect tampering with 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 SACs, 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)

Table 82 - Terms and Definitions	
Term	Definition
IEPPS	Information Exchange Policy-based Packaging & Processing Service
IEPPV	Information Exchange Packaging & Processing Policy Vocabulary
IES	Information Exchange Specification
IES Policy	Rules and constraints governing the release of data to specified users.
IM	Information Management
indeterminant	It does not lead to a definite end or result.
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 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.
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 Element	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.
Information Exchange Policy	Run-time serialization of an IEPPV InformationExchangeSpecification.
Information Exchange Policy Set	A general term identifying a group of InformationExchangePolicies 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 InformationExchangePolicy represents a serialization of the PolicyModels defined using the Information Exchange Packaging Policy Vocabulary (IEPPV)
Information Exchange Specification	An information supplier and consumer agreement is needed 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. Depending on the established agreements, packages may contain metadata, a Digest, a Structured Payload, Rendering Instructions, and optional linkages.

`

Table 82 - Terms and Definitions	
Term	Definition
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 Payload	A formatted dataset without protocols and metadata that is required for information exchange.
Information Processing	The parsing, transformation, and marshaling of information and data elements to information or data store(s).
Information Producer	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: • Trustworthy: Information quality and content can be trusted by stakeholders, decision-makers, and users; • Relevant: Information content tailored to the 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 standard 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 provided in a form that is brief and concise, yet including all necessary information; • Trusted: Information that is accepted as authoritative by stakeholders, decision-makers, and users and • Secure: Information is protected from accidental 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 Sharing Agreement	A written documentation outlines the legal authorities for collection, use, and disclosure when sharing information to ensure compliance.
Information Supplier	Any user, application, or system supplying information for distribution or dissemination.
Instrument	See Policy Instrument
ISA	Information Sharing Agreement
ISE	Information Sharing Environment
ISS	Information Sharing and Safeguarding
ISS Policy	Rules and constraints governing the sharing and safeguarding of user data.
KMS	Key Management Services
LDAP	Lightweight Directory Access Protocol

Table 82 - Terms and Definitions

Term	Definition
Legally Significance	It is important to a legal proceeding or action.
Legally Significant Information	Information must be captured, maintained, and protected 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 with no interoperability or integration. • Level 1: Technical Interoperability: a communication protocol exists for exchanging data between participating systems. • Level 2: Syntactic Interoperability: a standard data structure and format exists for data exchange between participating systems; • Level 3: Semantic Interoperability: A standard unambiguous semantic reference model exists for data exchange between participating systems. • Level 4: Pragmatic Interoperability: Participating systems are aware of each system's methods and procedures. • Level 5: Dynamic Interoperability: Systems operate on data over time and comprehend state changes in the operating environment. Manual or automated processes adjust to these changes using assumptions, rules, and constraints written in user-defined policies. • Level 6: Conceptual Interoperability: AI and Machine learning govern the ISS and DCS.
LEXS	Logical Entity eXchange Specification
LEXS(2)	LEISP Exchange Specification
Local Policy Store	Data, object, or file storage that maintains the current set of access control and ISS policy.
Local Services	Services that are defined and used by the user community to perform a specific set of functions.
Marshaling	This defines processes through which data sets are divided and transformed into the data elements described by the underlying data store(s).
MDA	Model Driven Architecture
MEM	Message Exchange Mechanism
Memorandum of Understanding	A bilateral or multilateral agreement between parties.
Message	A formatted information element transferred by a message switching or exchange system, middleware of web service
MessageElement	An identifiable part of a message structure containing contextually relevant data or metadata, including metadata, data payload, digest, links, and attachments.

`

Table 82 - Terms and Definitions

Term	Definition
Messaging Protocol	The rules, formats, and functions for exchanging messages between the components of a messaging system.
Metadata	Data (labels, tags, and markings) that describe other data.
Middleware	Software that serves as an intermediary between systems software and an application.
MILS	Multi-Independent Levels of Security
MIM	MIP Information Model
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	MILS is a high-assurance security architecture based on separating and controlling information flow. It integrates mechanisms that separate untrusted and trustworthy components, ensuring the total security solution is non-bypassable, evaluative, always invoked, and tamperproof. MILS employs one or more separation mechanisms (e.g., Separation kernel, Partitioning Communication System, physical separation) to maintain assured data and process separation. MILS supports enforcing one or more system-specific security policies by authorizing information flow between components in the same security domain or through trustworthy security monitors (e.g., access control guards, downgraders, crypto devices, etc.).
Multilevel Security	Information systems and networks that can process information with incompatible classifications (i.e., at different security levels and caveats). These systems and networks permit authorized users to access information elements (users holding the appropriate security clearances and needs-to-know) and prevent access to users without authorization.
NAF	NATO Architecture Framework
NATO	North Atlantic Treaty Organization
NGO	Non-Government Organization
NIEM	National Information Exchange Model
OCL	Object Constraint Language
Octet	An octet is a unit of computing and telecommunications digital information consisting of eight bits.
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 MetaModel

Table 82 - Terms and Definitions

Term	Definition
OMG	Object Management Group
Ontology	Ontology is a description of the concepts and relationships that can exist for an agent or a community of agents. In the context of knowledge sharing, ontology means a specification of a conceptualization. Ontology is a description of the concepts and relationships that exist for an agent or a community of agents. In the context of knowledge sharing, ontology means a specification of a conceptualization.
OODB	Object Oriented Database
OODBMS	Object Oriented Database Management System
OpenTDF	OpenTDF is an open-source system for implementing data-centric security. It provides the essential services required to enable the definition, application, and enforcement of attribute-based policies using the Zero Trust Data Format (ZTDF).
Operating Concept	It describes the operating characteristics of a system, typically from the viewpoint of the individuals who will use that system. It also describes the set of systems capabilities (e.g., services, decision and enforcement points, and interfaces) employed to achieve a desired effect, objective, or end state. Ideally, it offers a straightforward methodology to realize the goals and objectives for the system while not intending to be an implementation or transition plan itself. An operating concept may include: 1. Statement of the goals and objectives; 2. Operational conditions/contexts affecting the system; 3. Organizations, activities, processes, and interactions among participants using the system; 4. Specific operational concepts and processes for fielding the system and 5. Processes for initiating, developing, maintaining, and adapting the system.
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: 1. Role and Responsibility; 2. Operational Phase; 3. Operational Threat; 4. Operational Risk; 5. Command Intent; 6. Physical location; 7. Available communications links and 8. Access device.
Operational Domain	The sphere of knowledge, influence, or activity for a specific mission or operation.
ORDBMS	Object-Relational Database Management System

`

Table 82 - Terms and Definitions	
Term	Definition
OWL	Web Ontology Language
Package	For this specification, "Package" refers to 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 for managing and administering the configuration and policy environments for IEF Components in their designated operating environment.
Participant	A list of entities to produce or receive the information or message.
PBAC	Policy-Based Access Control
PDP	The Policy Decision Point adjudicates receipt of, access to, or release of data to specified users.
PDU	Protocol Data Unit
PEP	A Policy Enforcement Point (PEP) intercepts each information element transiting between a user client application and the server (email, instant messaging, 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 included in standard operating procedures to mitigate or recover from the impact of that incident.
Planned Threat	A threat for which standard operating procedures or safeguards exist to prevent or mitigate the impact of the threat.
Policy	A defined course of action or method of action selected from among alternatives and in light of given conditions to guide and determine present and future decisions.
Policy automation	The use of software services to enforce user-specified ISS, DCS, IES, and Semantic rules and constraints.
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 directing and describing methods or processes to be used or applied, including legislation, regulation, agency policy, memorandum of understanding (MoU), and service level agreements (SLA).
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.

Table 82 - Terms and Definitions	
Term	Definition
Policy Transformation	The single or multi-stage transformation of policy instruments into machine-readable and enforceable rules.
Policy-based Packaging and Processing Service	Software services that enforce user-defined policies governing the processing, packaging, and exchange of user data. PPS policies are developed using the Information Exchange Packaging Policy Vocabulary (IEPPV).
Policy-Right	Permission is granted through the application of policy.
PPS	Policy-based Packaging and Processing Service
PrivacyMetadata	Tags, labels, or markings that support the enforcement of privacy policy.
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 provided for specific purposes by an individual, which the individual can reasonably expect, will not be made public, including such items as: • person's Social Security number, • driver's license number, • employee identification number, • biometric identifiers, • passwords or other access codes, • medical records, • financial record, • home or personal telephone numbers and • personal email addresses.
Proprietary Information	Privately owned knowledge, information, or data, such as that protected by a registered patent, copyright, or trademark.
Protected Information	Sensitive information or digital assets that, if compromised, could reasonably be expected to cause injury to a non-national interest, e.g., an individual interest such as a person or an organization.
Protected-A	Sensitive information or digital assets that, if compromised, could cause injury to an individual, organization, or government.
Protected-B	Sensitive information or digital assets that could cause serious injury to an individual, organization, or government.
Protected-C	Sensitive information or digital assets that could cause extremely grave injury to an individual, organization, or government.
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 publishing sharable information to a data registry, repository, or publication-subscription middleware infrastructure. This metadata provides the structures required to represent the data and that is associated with the publishing and

`

Table 82 - Terms and Definitions

Term	Definition
	storage of data. The data registry, repository, or middleware receives and records the published metadata so users and systems can discover the associated information elements.
PVO	Private Volunteer Organization.
QoS	Quality of Service.
QoS History	A data record that is generated by the systems to synchronize late joining systems to the network.
Quality Information	See Information Quality.
Quality of Service	A set of attributes that define a middleware's ability to meet the requirements of a service level agreement for data delivery or management, such as reliability, ownership policy, history size, time-to-keep, etc.
RBAC	Role-Based Access Control
RDF	Resource Description Framework.
Real-time	A system's ability to respond to dynamic real-time events. In an ISS or DCS, this references the system's ability to deliver event-triggered (e.g., data change) global information updates across all nodes, systems, and applications requiring access to the information.
Redact	To obscure or remove (text or data) from a document before publication or release. Data filters typically provide this feature.
Reference Architecture	An authoritative source of information about a specific subject area that guides and constrains the instantiations of multiple architectures and solutions. It defines abstract architectural elements within the domain in terms that are independent of specific technologies, platforms, and tools.
Reference Model	It illustrates the interaction between IEF components when processing a message received for a PPS operating within the IEF's environment.
Releasability	The authorization to release a data object or message based on the sensitivity of its data content and the authorization of the recipient(s).
Releasable Dataset	A collection of data (/information) elements that can be provided (/released) to the recipient(s) based on the sensitivity of data content and the authorization of the recipient(s) governed by policy.
Releasable Message	A message (formatted data set) that can be provided (/released) to the recipient(s) based on the sensitivity of data content and the authorization of the recipient(s) governed by policy.
Reliability	A QoS attribute that describes the guarantees and feedback provided to the application regarding delivering the information supplied to the middleware.
Responsible Information Sharing	(1)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 and oversight - maximize the quantity and quality of information that is discoverable and accessible to authorized users and partners.

Table 82 - Terms and Definitions

Term	Definition
	(2)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 a remote intrusion.
SA	Situational Awareness
SAC	Secure Asset Container
Safeguard	Policies, rules, services, and technologies that guard or protect data and information elements from malicious or accidental release of sensitive or protected information.
SDS	Secure Data Service
Secret Information	Sensitive information or digital assets that, if compromised, could cause serious injury to the national interest, national security, public safety, defense, and maintenance of the social, political, and economic stability of the nation.
Secure Asset Container	An envelope allows some unprotected metadata to travel with a protected (encrypted) payload.
Secure Data Service	A configuration of IEF services and virtual security features that wraps a data store with Data-Centric Security.
Secure Relationship Protocol	An access-based access control protocol for defining, managing, and securing connections to any enterprise resource.
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, labels, and markings that assist in enforcing security policy and malicious or accidental release of classified information to unauthorized recipients.
Semantic Integrity	Compliance with the structure, format, and content (mandatory or optional) for information sets (or business objects).
Semantic Interoperability	A standard semantic reference model is employed, which defines data meaning unambiguously.
Semantic Pattern	A plan, diagram, or model to aggregate Transactional patterns that conform to an information-sharing and safeguarding agreement or Data patterns that describe a data set conforming to an information-sharing specification. It comprises a SemanticElement enclosing a

`

Table 82 - Terms and Definitions	
Term	Definition
	specified set of TransactionalElements and WrapperElements described in the PPS policy conforming to the IEPPV.
Semantic Policy	Rules and constraints governing the packaging of data for release to specified users.
Sensitive Information	Information (/data) elements are identified as classified, private, confidential, or legally significant.
Sensitivity Markings	A general reference to an information element's Security Level [1], caveats [0..*], Privacy Markings [0..*], and Legal Significance Marking [0..*].
Sensor 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.
Service Level Agreement	An agreement between two or more parties where the level of service is formally defined.
SIEM	Security Information and Event Management
SLA	Service Level Agreement
SMB	IEF Secure Messaging Bus. Also, see ISMB.
SOAR	Security Orchestration, Automation, and Response—This system provides integrated threat management and automated response to streamline security operations.
SOPES	Shared Operational Picture Exchange Services
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	A detailed and 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.
SRP	Secure Relationship Protocol
SSG	The Security Services Gateway (SSG) provides a single integration point between IEF components and users' security services and infrastructure. NOTE: The SSG also acts as a PEP between the IEF Components and users' security services (e.g., ICAM, Key Management, and Monitoring services).
Stage	To gather and prepare information for release to a community per established policy, memorandum of understanding, or service-level agreements.
Stakeholder	A person with an interest or concern in the practical application of ISS Policy.
STANAG	A Standardization Agreement (STANAG) is a NATO standardization document that specifies the agreement of member countries to implement a standard.

Table 82 - Terms and Definitions	
Term	Definition
StaticFilter	A filter created at design-time that cannot be modified at run-time.
Syntactic Interoperability	A standard structure and format are used to exchange information.
TDO	Trusted Data Object.
Tearline	A physical line on a message or document separates categories of information approved for disclosure and release.
Technical Interoperability	A communication protocol exists for exchanging data between participating systems. Communication infrastructure is established on this level, allowing systems to exchange bits and bytes, and the underlying networks and protocols are unambiguously defined. This level ensures a common understanding of signals.
TLS	The Trusted Logging Service (TLS) securely records IEF components and acts as a transactional history of policy decisions and access control enforcement.
Top Secret Information	It applies to information or digital assets that, if compromised, could cause exceptionally grave injury to the national interest, national security, public safety, defense, and maintenance of the nation's social, political, and economic stability.
Trust	Within the scope of this Specification, Trust refers to the level of confidence an information supplier has regarding the release of selected information to a specific consumer of that information.
Trustmark	A type of certification mark or symbol representing due compliance with established standards or regulatory requirements within privacy, security, or quality.
UBAC	User-Based Access Control.
UML	Unified Modeling Language.
Unplanned Incidents	An occurrence of an event 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 who requests access to a resource (e.g., IEF Component, information element, or data element) that has a specified set of privileges (e.g., policy rights, attributes, and attributes) that permit or deny access to that participant to services and resources (e.g., data elements, information elements, system devices, applications, and 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, Session, and Network. Each member of the identified categories must have credentials, attributes, attributes, and policy rights that specify their rights to access the resource.
UUID	Universally Unique Identifier
Validate	To give official sanction, confirmation, or approval to a specified element.

`

Table 82 - Terms and Definitions

Term	Definition
Verify	To ascertain the truth or correctness of a specified element by examination, research, or comparison.
VHF	Very High Frequency
Vocabulary	A representation of a set of concepts by formal, descriptive statements that differentiate those concepts from related concepts within a given domain or area of expertise.
WatchPoint	An element within a policy model triggers a mechanism to commence assembling semantic and transactional elements.
WrapperElement	A logical construct that wraps or encapsulates the definition of a data set, table entity, triple, file, etc.
XACML	XML Access Control Markup Language
XMI	XML Metadata Interchange
XML	Extensible Markup Language
Zero Trust Computing	The implementation and operation of a Zero-Trust architecture.
Zero-Trust Architecture	Zero Trust (ZT) is a strategic cyber security approach that secures and eliminates implicit trust and continuously validates every digital interaction. ZT enforces the principle of “never trust, always verify.” by employing robust authentication methods, leveraging network segmentation, preventing lateral movement, Layer 7 threat prevention, and granular “least access” policies.
ZTA	Zero-Trust Architecture
ZTDF	Zero Trust Data format

`

Report Generation Scripts

#PrintClassDescriptions