

MILE Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: May 10, 2013

K. Moriarty  
S. Tabet  
EMC  
D. Waltermire  
NIST  
November 6, 2012

GRC Report Exchange  
draft-ietf-mile-grc-exchange-01.txt

Abstract

Governance, risk, and compliance (GRC) programs provide oversight (governance) of risks and compliance initiatives within an organization. GRC reports are increasingly provided in an XML format. This specification defines a common method to securely transport GRC and other XML reports. The defined messaging capability provides policy options and markings in an XML schema, options for confidentiality at the document/report level, and security for the end-to-end communication. XML reports may be shared between service providers and clients, enterprises, or within enterprises. Reports may also be exchanged for official purposes such as business report filings, compliance report filings, and the handling of legal incidents (eWarrant, eDiscovery, etc.) This work is a generalized format derived from the secure exchange of incident information defined by RFC6545, Real-time Inter-network Defense (RID).

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on May 10, 2013.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	4
1.1. Normative and Informative . . . . .	5
1.2. Terminology . . . . .	5
2. Report Types . . . . .	5
3. Communication between Entities . . . . .	6
3.1. Inter-network Provider GRC Messaging . . . . .	6
3.2. GRC Report Exchange Communication Topology . . . . .	7
3.3. Message Formats . . . . .	7
3.4. GRC Report Exchange Data Types . . . . .	7
3.4.1. Boolean . . . . .	7
3.4.2. Language . . . . .	8
3.4.3. Multilingual Strings . . . . .	8
3.4.4. Uniform Resource Locator strings . . . . .	8
3.4.5. Date-Time Strings . . . . .	8
3.4.6. Timezone String . . . . .	9
3.4.7. Postal Address . . . . .	9
3.4.8. Telephone and Fax Numbers . . . . .	9
3.4.9. Email String . . . . .	10
3.5. GRC Report Exchange Message Types . . . . .	10
4. GRC-Exchange Schema . . . . .	11
4.1. GRCPolicy Class . . . . .	13
4.2. RequestStatus . . . . .	17
4.3. GRCDocument . . . . .	19
4.4. Reference Class . . . . .	24
4.5. ReportID Class . . . . .	24
4.6. Contact Class . . . . .	26
4.6.1. RegistryHandle Class . . . . .	29
4.6.2. PostalAddress Class . . . . .	30
4.6.3. Email Class . . . . .	31
4.6.4. Telephone and Fax Classes . . . . .	31
4.7. ExtensionType Class . . . . .	32
4.8. Node Class . . . . .	35

4.9. Address Class . . . . .	36
4.10. GRC-Exchange Name Spaces . . . . .	37
5. Extending the Enumerated Values of Attributes . . . . .	37
6. GRC Report Exchange Messages . . . . .	38
6.1. Acknowledgement . . . . .	38
6.2. Result . . . . .	39
6.3. Request . . . . .	40
6.4. Report . . . . .	41
6.5. Query . . . . .	42
7. GRC-Exchange Communication Flows . . . . .	43
7.1. Report Communication Flow . . . . .	43
7.1.1. GRC-Exchange Report Example . . . . .	44
7.2. Request Communication Flow . . . . .	44
7.2.1. Request Example . . . . .	45
7.2.2. Acknowledgement Message Example . . . . .	45
7.3. Query Communication Flow . . . . .	45
7.3.1. Query Example . . . . .	46
7.3.2. Acknowledgement Message Example . . . . .	46
7.3.3. Result Message Example . . . . .	47
8. Internationalization Issues . . . . .	47
9. GRC-Exchange Schema Definition . . . . .	49
10. Requirements for GRC XML Schemas for GRC-Exchange . . . . .	49
11. Security Requirements . . . . .	50
11.1. XML Digital Signatures and Encryption . . . . .	50
11.2. Public Key Infrastructure . . . . .	53
11.2.1. Authentication . . . . .	54
11.2.2. Multi-Hop Request Authentication . . . . .	55
11.3. Consortiums and Public Key Infrastructures . . . . .	56
11.4. Privacy Concerns and System Use Guidelines . . . . .	57
11.5. Sharing Profiles and Policies . . . . .	60
12. Security Considerations . . . . .	61
13. IANA Considerations . . . . .	61
14. Acknowledgements . . . . .	63
15. Summary . . . . .	64
16. References . . . . .	64
16.1. Normative References . . . . .	64
16.2. Informative References . . . . .	66
Authors' Addresses . . . . .	67

## 1. Introduction

Governance, risk, and compliance (GRC) programs provide oversight (governance) of risks and compliance initiatives within an organization. The areas typically covered by GRC include:

- o Finance and Business Operations,
- o Information Technology,
- o Security, and
- o Legal and Compliance

GRC Report Exchange provides a secure method to communicate relevant information and reports, through the automated exchange of extensible markup language (XML) documents. GRC Report Exchange considers security, policy, and privacy issues as related to the exchange of potentially sensitive information. Additionally, it enables organizations accepting GRC report filings, such as service providers or enterprises, the options to make appropriate decisions according to their policy requirements. GRC Report Exchange includes provisions for confidentiality, integrity, and authentication.

The data in GRC reports exchanged are represented in an XML [W3C.REC-xml-20081126] document using the appropriate XML schema for the included report. The XML document or formatted report is then enveloped by the GRC Report Exchange schema to set policy options and provide a common secure exchange method for such documents. By following this model, a single method for all GRC reports can be used, simplifying the integration of GRC reports across platforms.

Security and privacy considerations are of high concern since potentially sensitive information may be passed through GRC Report Exchange messages. GRC Report Exchange takes advantage of XML security and privacy policy information set in the GRC Report Exchange schema and provides standard settings for fine grain controls within GRC XML schemas. The GRC Report Exchange schema acts as an XML envelope to support the communication of GRC report documents. GRC Report Exchange messages are encapsulated for transport, which is defined in a separate document [RFC6546]. The authentication, integrity, and authorization features GRC Report Exchange and RID transport offer are used to achieve a necessary level of security.

GRC report exchange is not strictly a technical problem. There are numerous procedural, trust, and legal considerations that might prevent an organization from sharing information. GRC Report

Exchange provides information and options that can be used by organizations who must then apply their own policies for sharing information. Organizations must develop policies and procedures for the use of the GRC Report Exchange protocol and XML reports.

### 1.1. Normative and Informative

The XML schema [W3C.REC-xmlschema-1-20041028] and transport requirements contained in this document are normative; all other information provided is intended as informative. More specifically, the following sections of this document are intended as informative: Sections XXX. The following sections of this document are normative: Sections XXX.

### 1.2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

## 2. Report Types

There are many possible report types that may be exchanged using GRC Report Exchange. The reports MUST all be XML formatted reports and MAY leverage the data markings used by this specification to require security options such as encryption on the entire report (XML document) or a section of the report.

The types of reports may vary within each area of GRC. Example report types broken out by GRC focus areas include:

- o Finance and Business Operations
  - \* Finance or business Filing Report
- o Information Technology
  - \* Service Level Agreement (SLA) Reports from service providers (public cloud providers, community cloud providers, etc.)
- o Security
  - \* Security Report aligned to control requirements (ISO27002, NIST 800-53, etc.) from service providers

- o Legal and Compliance
  - \* eDiscovery Reports
  - \* eWarrant Reports
  - \* Compliance report aligned to specific regulations
  - \* Report for internal or external audit aligned to risk and control frameworks (ISO27002, NIST 800-53, COSO, COBIT, etc.)

### 3. Communication between Entities

Trust relationships. Service provider to tenant or client is the most likely scenario for the initial use cases of GRC report exchange. See Section 11.5 on profiles.

#### 3.1. Inter-network Provider GRC Messaging

GRC Report Exchange provides a standard protocol and format that is required to ensure inter-operability between vendors for the exchange and filing of GRC reports. GRC Report Exchange provides the framework necessary for communication between entities exchanging or filing GRC reports. Several message types described in Section 6 are necessary to facilitate the exchange or filing of reports. The message types include the Report, Query, Acknowledgement, Result, and the Request message.

The Report message is used when a GRC report is to be filed on a system or associated database accepting GRC Report Exchange messages, where no further action is required. A Query message is used to request information on a particular report. The Acknowledgement and Report messages are used to communicate the status and result of a Query or Request message.

Use of the communication network and the GRC Report Exchange protocol must be for pre-approved, authorized purposes only. It is the responsibility of each participating party to adhere to guidelines set forth in both a global use policy established through the peering agreements for each bilateral peer or agreed-upon consortium guidelines. The purpose of such policies is to avoid abuse of the system; the policies shall be developed by a consortium of participating entities. The global policy may be dependent on the domain it operates under; for example, a government network or a commercial network such as the Internet would adhere to different guidelines to address the individual concerns. Privacy issues must be considered in public networks such as the Internet. Privacy

issues are discussed in the Security Requirements section (Section 11), along with other requirements that must be agreed upon by participating entities.

The GRC Report Exchange system should be configurable to either require user input or automatically provide or file reports. If the trust relationship is not strong, it may not be in the peer's best interest to accept a report or respond to a request. The trust relationship may evolve over time through experience working with a peer and knowledge and review of their policies and operating procedures.

### 3.2. GRC Report Exchange Communication Topology

The most basic topology for communicating GRC Report Exchanges is a direct connection or a bilateral relationship as illustrated below.



Figure 1: Direct Peer Topology

A star topology may be desirable in instances where a peer may be a provider of GRC Reports. This requires trust relationships to be established between the provider of information and each of the consumers of that information. Examples may include clients that file compliance or business reports to an authoritative entity.

The examples provided serve as an initial baseline set of expected topologies that may change over time.

### 3.3. Message Formats

Section 6 describes the five GRC Report Exchange message types, to be used with the appropriate XML documents. The messages are expected to be generated and received on designated systems for GRC report exchanges.

### 3.4. GRC Report Exchange Data Types

#### 3.4.1. Boolean

A boolean value is represented by the BOOLEAN data type.

The BOOLEAN data type is implemented as "xs:boolean" [W3C.REC-xmlschema-1-20041028] in the schema.

### 3.4.2. Language

A language value is represented by the LANG data type.

The LANG data type is a valid language code per [RFC5646] constrained by the definition of "xs:language" [W3C.REC-xmlschema-1-20041028] inherited from [W3C.REC-xml-20081126].

### 3.4.3. Multilingual Strings

STRING data that represents multi-character attributes in a language different than the default encoding of the document is of the ML\_STRING data type.

The ML\_STRING data type is implemented as an "grc-exchange:MLStringType" in the schema.

The base definition of this type is reused from the IODEF specification [RFC5070], Section 2.4. This definition is fully included in the GRC-Exchange specification in Section 4.8 to prevent the need to use the IODEF schema.

### 3.4.4. Uniform Resource Locator strings

A uniform resource locator (URL) is represented by the URL data type. The format of the URL data type is documented in [RFC3986].

The URL data type is implemented as an "xs:anyURI" [W3C.REC-xmlschema-1-20041028] in the schema.

### 3.4.5. Date-Time Strings

Date-time strings are represented by the DATETIME data type. Each date-time string identifies a particular instant in time; ranges are not supported.

Date-time strings are formatted according to a subset of ISO 8601: 2004 [ISO.8601.2000] documented in [RFC3339].

The DATETIME data type is implemented as an "xs:dateTime" [W3C.REC-xmlschema-1-20041028] in the schema.

The base definition of this type is reused from the IODEF specification [RFC5070], Section 2.8. This definition is fully included in the GRC-Exchange specification in Section 4.8 to prevent the need to use the IODEF schema.



#### 3.4.6. Timezone String

A timezone offset from UTC is represented by the TIMEZONE data type. It is formatted according to the following regular expression:  
"Z|[\+\-](0[0-9]|1[0-4]):[0-5][0-9]".

The TIMEZONE data type is implemented as an "xs:string" [W3C.REC-xmlschema-1-20041028] with a regular expression constraint in the schema. This regular expression is identical to the timezone representation implemented in an "xs:dateTime".

The base definition of this type is reused from the IODEF specification [RFC5070], Section 2.9. This definition is fully included in the GRC-Exchange specification in Section 4.8 to prevent the need to use the IODEF schema.

#### 3.4.7. Postal Address

A postal address is represented by the POSTAL data type. This data type is an ML\_STRING whose format is documented in Section 2.23 of [RFC4519]. It defines a postal address as a free-form multi-line string separated by the "\$" character.

The POSTAL data type is implemented as an "xs:string" [W3C.REC-xmlschema-1-20041028] in the schema.

The base definition of this type is reused from the IODEF specification [RFC5070], Section 2.11. This definition is fully included in the GRC-Exchange specification in Section 4.8 to prevent the need to use the IODEF schema.

#### 3.4.8. Telephone and Fax Numbers

A telephone or fax number is represented by the PHONE data type. The format of the PHONE data type is documented in Section 2.35 of [RFC4519].

The PHONE data type is implemented as an "xs:string" [W3C.REC-xmlschema-1-20041028] in the schema.

The base definition of this type is reused from the IODEF specification [RFC5070], Section 2.13. This definition is fully included in the GRC-Exchange specification in Section 4.8 to prevent the need to use the IODEF schema.

### 3.4.9. Email String

An email address is represented by the EMAIL data type. The format of the EMAIL data type is documented in Section 3.4.1 of [RFC5322].

The EMAIL data type is implemented as an "xs:string" [W3C.REC-xmlschema-1-20041028] in the schema.

The base definition of this type is reused from the IODEF specification [RFC5070], Section 2.14. This definition is fully included in the GRC-Exchange specification in Section 4.8 to prevent the need to use the IODEF schema.

### 3.5. GRC Report Exchange Message Types

The five GRC Report Exchange message types are as follows:

1. Acknowledgement. This message is sent to the requestor of a report (Request) or in response to a Query to notify on the state of a request (approved, pending, not approved).
2. Result. This message is sent to the requestor of a GRC report (Request) or in response to a Query. The Result may contain the full report requested or a section of the report as appropriate for the request in the Query.
3. Request. This message type is used to request a specific type of GRC report. The Request MUST specify the XML schema and version for the requested report along with any other parameters required in the XML schema to generate the correct report.
4. Report. This message is used to provide a GRC Report. The message can be considered a wrapper for any approved GRC schema used to format a report for submission.
5. Query. This message is used to request information about a specific GRC report. The XML schema and version used MUST be specified along with any details required to provide the proper Report response. The response is provided through the Report message.

When an application receives a GRC Report Exchange message, it must be able to determine the type of message and parse it accordingly. The message type is specified in the GRCPolicy class. The GRCPolicy class may also be used by the transport protocol to facilitate the secure communication of the GRC Report Exchange.

#### 4. GRC-Exchange Schema

There are three classes included in the GRC Report Exchange schema required to facilitate communications. The RequestStatus class is used to indicate the approval status of a report Request or Query; the GRCDocument class identifies the XML schema to be used by the provided or requested report; and the GRCPolicy class provides information on the agreed-upon policies and specifies the type of communication message being used.

The GRC Report Exchange schema acts as an envelope for the GRC XML schema to facilitate secure GRC report communications. The intent in maintaining a separate schema is for the flexibility of sending messages between participating entities. Since GRC Report Exchange is a separate schema that includes the appropriate GRC XML schema, the GRC Report Exchange information acts as an envelope, and then the GRCPolicy class can be easily extracted for use by the transport protocol.

The security requirements of sending GRC reports and associated information on finance, IT operations, legal, compliance, and security across the network include the use of confidentiality (encryption prior to the transport level), authentication (potentially multi-hop), integrity, and non-repudiation. GRCPolicy uses labels that correspond to policy and agreements to standardize on handling requirements such as encryption and sharing limitations. The GRCPolicy information should not be encrypted, hence GRC Report Exchange is maintained separate from the GRC XML schema used to send or request a report. This segregation enables flexibility for GRC Report Exchange to be used with any GRC XML schema and removes the need for decrypting and parsing the entire GRC Report and GRC Report Exchange document to determine how it should be handled at each entity communicating via GRC Report Exchange.

The purpose of the GRCPolicy class is to specify the message type for the receiving host, facilitate the policy needs of GRC Reports, and provide routing information in the form of an IP address of the destination entity accepting GRC Report Exchange messages.

The policy information and guidelines are discussed in Section 4.1. The policy is defined between GRC-Exchange peers and within or between consortiums. The GRCPolicy is meant to be a tool to facilitate the defined policies. This MUST be used in accordance with policy set between clients, peers, consortiums, and/or regions. Security, privacy, and confidentiality MUST be considered as specified in this document.

The GRC Report Exchange (GRC-Exchange) schema is defined as follows:

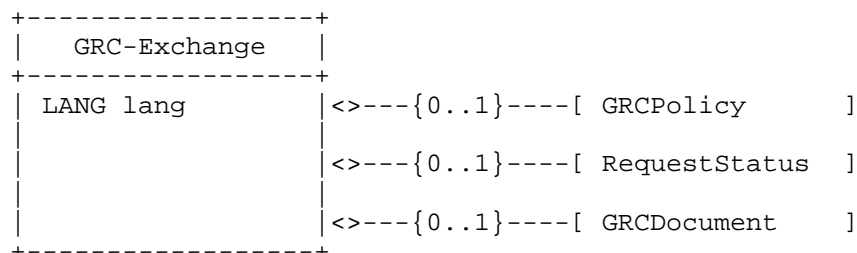


Figure 2: The GRC-Exchange Schema

The aggregate classes that constitute the GRC-Exchange schema in the grc-exchange namespace are as follows:

#### GRCPolicy

Zero or One. The GRCPolicy class is used by all message types to facilitate policy agreements between peers, consortiums, or federations, as well as to properly route messages.

#### RequestStatus

Zero or One. The RequestStatus class is used only in Acknowledgement messages to report back to the entity requesting a report or sending a report Query if the request is denied or remains in a pending state.

#### GRCDocument

Zero or One. The GRCDocument class is used in each of the message types to state the XML schema and version for the included XML report, XML report request, or response.

The GRC-Exchange class defines one attribute as follows:

#### lang

REQUIRED. ENUM. A valid language code per [RFC5646] constrained by the definition of "xs:language" [W3C.REC-xmlschema-1-20041028] inherited from [W3C.REC-xml-20081126].

Each of the three listed classes may be the only class included in the GRC-Exchange class, hence the option for zero or one. In some cases, GRCPolicy MAY be the only class in the GRC-Exchange definition when used by the transport protocol [RFC6546], as that information should be as small as possible and may not be encrypted. The Acknowledgement message using the RequestStatus class MUST be able to

stand alone without the need for an GRC XML document to facilitate the communication, limiting the data transported to the required elements per [RFC6546].

#### 4.1. GRCPolicy Class

The GRCPolicy class facilitates the delivery of GRC Report Exchange messages.

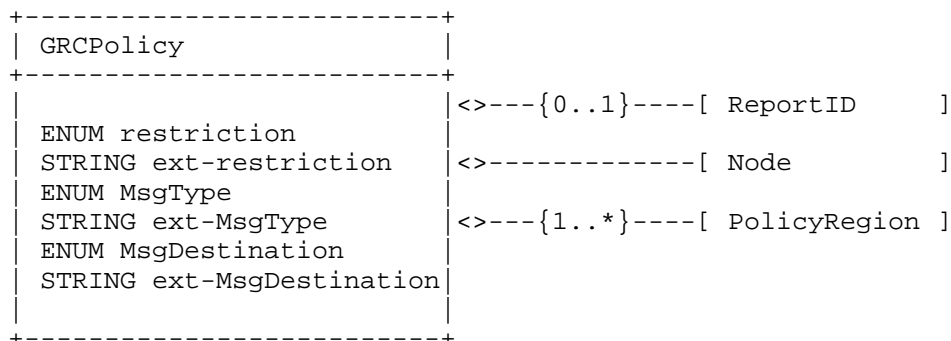


Figure 3: The GRCPolicy Class

The aggregate elements that constitute the GRCPolicy class are as follows:

##### ReportID

Zero or one. Global reference pointing back to the ReportID defined in the GRC XML data model. The ReportID includes the domain name of the entity who creates the report, a report number, and an instance of that report number. The default report number is a date, where the requested report is the most recent report on or prior to the date specified. The format for the date SHALL be YYYYMMDD, where Y is the year, M is the month, and D is the day. The instance number is appended with a dash separating the values and is used in cases for which there may be multiple reports issued in a day. The format for the instance SHALL be HHMMSS, where H is the hour as specified in a 24hour range, M is the minute, S is the second provided in GMT. An alternate ID may be specified within the GRC XML schema for the specific report. This element has been derived from IODEF [RFC5070].

##### Node

One. The Node class is used to identify a host or network device, in this case to identify the system communicating GRC-Exchange

messages. The base definition of this class is reused from the IODEF specification [RFC5070], Section 3.16. This definition is fully included in the GRC-Exchange specification in Section 4.8 to prevent the need to use the IODEF schema.

#### PolicyRegion

One or many. REQUIRED. The values for the attribute "region" are used to determine what policy area may require consideration before a trace can be approved. The PolicyRegion may include multiple selections from the attribute list in order to fit all possible policy considerations when crossing regions, consortiums, or networks.

#### region

One. ENUM.

1. ClientToSP. An enterprise initiated the request to their service provider.
2. SPToClient. An service provider passed a GRC request or report to a client or an enterprise based on requested services or service level agreements.
3. IntraConsortium. GRC report information that should have no restrictions within the boundaries of a consortium with the agreed-upon use and abuse guidelines.
4. PeerToPeer. GRC report information that should have no restrictions between two peers but may require further evaluation before continuance beyond that point with the agreed-upon use and abuse guidelines.
5. BetweenConsortiums. GRC report information that should have no restrictions between consortiums that have established agreed-upon use and abuse guidelines.
6. ext-value. An escape value used to extend this attribute. This attribute has been derived from IODEF [RFC5070], Section 5.1 and is explained in Section 5, Extending the Enumerated Values of Attributes.

Additionally, there is an extension attribute to add new enumerated values:

ext-region. An escape value used to extend this attribute. This attribute has been derived from IODEF [RFC5070] Section

5.1 and is explained in Section 5, Extending the Enumerated Values of Attributes.

The GRCPolicy class has six attributes:

restriction

Optional. ENUM. This attribute indicates the disclosure guidelines to which the sender expects the recipient to adhere for the information represented in this class and its children. This guideline provides no security since there are no specified technical means to ensure that the recipient of the document handles the information as the sender requested.

The value of this attribute is logically inherited by the children of this class. That is to say, the disclosure rules applied to this class, also apply to its children.

It is possible to set a granular disclosure policy, since all of the high-level classes (i.e., children of the Incident class) have a restriction attribute. Therefore, a child can override the guidelines of a parent class, be it to restrict or relax the disclosure rules (e.g., a child has a weaker policy than an ancestor; or an ancestor has a weak policy, and the children selectively apply more rigid controls). The implicit value of the restriction attribute for a class that did not specify one can be found in the closest ancestor that did specify a value.

This attribute is defined as an enumerated value with a default value of "private". Note that the default value of the restriction attribute is only defined in the context of the GRCPolicy class. In other classes where this attribute is used, no default is specified.

This attribute is derived from IODEF [RFC5070] and is fully included within this schema.

1. public. There are no restrictions placed in the information.
2. need-to-know. The information may be shared with other parties that are involved in the incident as determined by the recipient of this document (e.g., multiple victim sites can be informed of each other).
3. private. The information may not be shared.

4. default. The information can be shared according to an information disclosure policy pre-arranged by the communicating parties.
5. ext-value. An escape value used to extend this attribute. This attribute has been derived from IODEF [RFC5070], Section 5.1 and is explained in Section 5, Extending the Enumerated Values of Attributes.

#### ext-restriction

OPTIONAL. STRING. A means by which to extend the restriction attribute. This attribute has been derived from IODEF [RFC5070] Section 5.1 and is explained in Section 5, Extending the Enumerated Values of Attributes.

#### MsgType

REQUIRED. ENUM. The type of GRC-Exchange message sent. The five types of messages are described in Section 3.5 and can be noted as one of the five selections below.

1. Acknowledgement. This message is sent to the initiating GRC-Exchange entity if a Request or Query has been denied or is pending.
2. Result. This message provides the result of a Query.
3. Request. The purpose of the Request is to request a report from an entity.
4. Report. This message is used to provide a GRC XML report.
5. Query. This message is used to request information either about a specific report or group of reports. The actual query is specified in the GRC XML Schema and is outside the scope of this specification.
6. ext-value. An escape value used to extend this attribute. This attribute has been derived from IODEF [RFC5070], Section 5.1 and is explained in Section 5, Extending the Enumerated Values of Attributes.

#### ext-MsgType

OPTIONAL. STRING. A means by which to extend the MsgType attribute. This attribute has been derived from IODEF [RFC5070] Section 5.1 and is explained in Section 5, Extending



the Enumerated Values of Attributes.

#### MsgDestination

REQUIRED. ENUM. The destination required at this level may either be the system accepting GRC report exchange requests or reports. The Node element lists the address of the host receiving the GRC-Exchange message.

1. GRCSysyem. The address listed in the Node element of the GRCPolicy class is the system communicating via GRC-Exchange that will receive the message.
2. ext-value. An escape value used to extend this attribute. This attribute has been derived from IODEF [RFC5070] Section 5.1 and is explained in Section 5, Extending the Enumerated Values of Attributes.

#### ext-MsgDestination

OPTIONAL. STRING. A means by which to extend the MsgDestination attribute. This attribute has been derived from IODEF [RFC5070] Section 5.1 and is explained in Section 5, Extending the Enumerated Values of Attributes.

### 4.2. RequestStatus

The RequestStatus class is an aggregate class in the GRC-Exchange class.

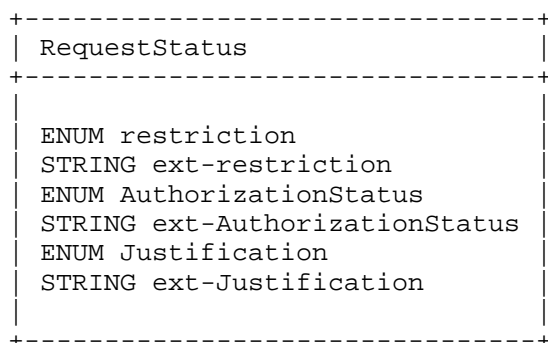


Figure 4: The RequestStatus Class

The RequestStatus class has six attributes:

#### restriction

OPTIONAL. ENUM. This attribute indicates the disclosure guidelines to which the sender expects the recipient to adhere. This guideline provides no real security since it is the choice of the recipient of the document to honor it. This attribute follows the same guidelines as "restriction" used in IODEF and is explained in the GRCPolicy Class description in Section 4.1.

#### ext-restriction

OPTIONAL. STRING. A means by which to extend the restriction attribute. This attribute has been derived from IODEF [RFC5070] Section 5.1 and is explained in Section 5, Extending the Enumerated Values of Attributes.

#### AuthorizationStatus

REQUIRED. ENUM. The listed values are used to provide a response to the requesting entity of the ReportRequest or ReportQuery.

1. Approved. The request was approved and will be provided. The approved message MAY be sent if there will be a delay in providing the report, otherwise, the Report or Result MAY be provided without sending a Acknowledgement message.
2. Denied. The request has been denied.
3. Pending. Awaiting approval; a timeout period has been reached, which resulted in this Pending status and Acknowledgement message being generated.
4. ext-value. An escape value used to extend this attribute. This attribute has been derived from IODEF [RFC5070] Section 5.1 and is explained in Section 5, Extending the Enumerated Values of Attributes.

#### ext-AuthorizationStatus

OPTIONAL. STRING. A means by which to extend the AuthorizationStatus attribute. This attribute has been derived from IODEF [RFC5070] Section 5.1 and is explained in Section 5, Extending the Enumerated Values of Attributes.

#### Justification

OPTIONAL. ENUM. Provides a reason for a Denied or Pending message.

1. SystemResource. A resource issue exists on the systems that would be involved in the request.
2. Authentication. The enveloped digital signature [RFC3275] failed to validate.
3. AuthenticationOrigin. The detached digital signature for the original requestor on the RecordItem entry failed to validate.
4. Encryption. Unable to decrypt the request.
5. UnrecognizedFormat. The format of the provided document was unrecognized.
6. CannotProcess. The document could not be processed. Reasons may include legal or policy decisions. Resolution may require communication outside of this protocol to resolve legal or policy issues. No further messages SHOULD be sent until resolved.
7. Other. There were other reasons this request could not be processed.
8. ext-value. An escape value used to extend this attribute. This attribute has been derived from IODEF [RFC5070] Section 5.1 and is explained in Section 5, Extending the Enumerated Values of Attributes.

ext-Justification-ext

OPTIONAL. STRING. A means by which to extend the Justification attribute. This attribute has been derived from IODEF [RFC5070] Section 5.1 and is explained in Section 5, Extending the Enumerated Values of Attributes.

#### 4.3. GRCDocument

The GRCDocument class is an aggregate class in the GRC-Exchange class.

GRCDocument	
ENUM Version	<>---{1..*}----[ ReportType ]
STRING ext-Version	<>---{0..1}----[ FromContact ]
ENUM XMLSchemaID	
STRING ext-XMLSchemaID	<>---{0..1}----[ URL ]
ENUM restriction	
STRING ext-restriction	<>---{1}-----[ XMLDocument ]
	<>---{0..*}----[ Signature ]

Figure 5: The GRCDocument Class

The elements that constitute the GRCDocument class are as follows:

#### ReportType

One or many. REQUIRED. The values for the attribute "type" are meant to assist in determining if the included XML report or request is appropriate for the entity receiving the request or report message. Multiple values may be selected for this element; however, where possible, it should be restricted to one value that most accurately describes the report type.

#### type

One. ENUM.

1. Filing. This ReportType is used when a GRC report is included for expected filing purposes. Examples may include the filing of a regulatory or business operations report to a regulatory body.
2. Service Level Agreement. This option specifies the report type as a report on a service level agreement. This report may be sent from a service provide (SP) to a tenant or client or from a trust authority to a requesting entity. An SLA report may be associated with any report format (XML) associated with an SLA agreement, including but not limited to an IT or security report.

3. Operational. An operational report may include any standard operating reports used within or between businesses or enterprises. This may be a routine business, IT operational, or other type of report.
4. Compliance. A compliance report is specified when there is a specific compliance report format required (as specified by the schema used for the report). This type may be used for internal or external compliance report exchanges.
5. Audit. The Audit report type is distinguished from a compliance report as the report contents may vary depending on the report or report request in the exchange. An audit report may take an approach of only providing the state of compliance or details of findings from an automated control review.
6. RiskAssessment. A RiskAssessment report differs from the Compliance and Audit reports in that the report may prioritize risks as specified in the XML schema and may include GRC-XML risk ratings. A RiskAssessment may be provided for any GRC area or on the GRC program as specified by the GRCDocument.
7. OfficialBusiness. This option MUST be used if the GRC information is requested by or affiliated with any government or other official business request. This could be used during an investigation for an eDiscovery, eWarrant, or other use case.
8. Other. If this option is selected, a description of the request MUST be provided so that policy decisions can be made to proceed with the request or act upon the report. The information should be provided in the GRC-Exchange class meaning attribute.
9. ext-value. An escape value used to extend this type. This value has been derived from IODEF [RFC5070], Section 5.1 and is explained in Section 5, Extending the Enumerated Values of Attributes.

ext-type

OPTIONAL. One. STRING. A means by which to extend the type attribute. This attribute has been derived from IODEF [RFC5070] Section 5.1 and is explained in Section 5, Extending the Enumerated Values of Attributes.

#### FromContact

Zero or more. Contact. Provides contact information for the parties responsible for a report provided in the GRC Report Exchange as defined by the Contact class in Section 4.6.

#### URL

Zero or One. URL. A reference to the XML schema included. The URL data type is defined in Section 3.4.4. The schemaLocation for IODEF is already included in the RID schema, so this is not necessary to include a URL for IODEF documents.

#### XMLDocument

One. ExtensionType. The XMLDocument is a complete XML document defined by the ExtensionType class in Section 4.7. This class follows the guidelines in [RFC5070] Section 5 where the data type is set to "xml" and meaning is set to "xml" to include an xml document.

#### Signature

Zero to many. ExtensionType. The Signature includes a complete XML document with the type specified by the ExtensionType class in Section 4.7. This class follows the guidelines in [RFC5070] Section 5 where the data type is set to "xml" and meaning is set to "xml" to include an xml document. The usage of this element is similar to RID [RFC6545] and is used to encapsulate the detached signature based on a specific class within the XML document to verify the originator of the message. If a detached signature is used, guidance for the encapsulated document MUST be provided as to which class should be used to create the signature. Alternatively, if no guidance is provided, the digital signature MUST be an enveloped signature of the entire XML document that is encapsulated. This attribute has been derived from RID [RFC6545], Section 5.1.1.

The GRCDocument class has six attributes:

#### Version

OPTIONAL. One. The Version attribute is the version number of the specified XML schema. That schema must be an approved version of a schema registered with IANA for use with GRC-Exchange. The IANA registry for managing schemas used with GRC-Exchange is specified in Section 13. This attribute has

been derived from RID [RFC6545], Section 5.1.1.

ext-value. An escape value used to extend this attribute. This attribute has been derived from IODEF [RFC5070], Section 5.1 and is explained in Section 5, Extending the Enumerated Values of Attributes.

#### ext-Version

OPTIONAL. One. STRING. The ext-Version attribute is the version number of the included XML schema. This attribute is used if a schema other than an IANA registered schema that has been added to the enumerated list for Version is included. This attribute has been derived from RID [RFC6545], Section 5.1.1.

#### XMLSchemaID

OPTIONAL. One. URL. The XMLSchemaID attribute is the identifier, the defined namespace [W3C.REC-xml-names-20091208], of the XML schema of the XML document included. The XMLSchemaID and Version specify the format of the XMLDocument element. The only permitted values, include any namespace listed in the IANA managed list of registered schemas for use with GRC-Exchange. The IANA registry for managing schemas is specified in Section 13. This attribute has been derived from RID [RFC6545], Section 5.1.1.

ext-value. An escape value used to extend this attribute. This attribute has been derived from IODEF [RFC5070], Section 5.1 and is explained in Section 5, Extending the Enumerated Values of Attributes.

#### ext-XMLSchemaID

OPTIONAL. One. The ext-XMLSchemaID attribute is the identifier (defined namespace) of the XML schema of the XML document included. The ext-XMLSchemaID and ext-Version specify the format of the XMLDocument element and are used if the included schema is not an IANA registered schema that has been added to the enumerated list for XMLSchemaID. This attribute has been derived from RID [RFC6545], Section 5.1.1.

#### restriction

OPTIONAL. ENUM. This attribute indicates the disclosure guidelines to which the sender expects the recipient to adhere. This guideline provides no real security since it is the choice

of the recipient of the document to honor it. This attribute follows the same guidelines as "restriction" used in IODEF and is explained in the GRCPolicy Class description in Section 4.1.

#### ext-restriction

OPTIONAL. STRING. A means by which to extend the restriction attribute. This attribute has been derived from IODEF [RFC5070] Section 5.1 and is explained in Section 5, Extending the Enumerated Values of Attributes.

### 4.4. Reference Class

The Reference class is a reference to the GRC Schema used for the exchange. A reference consists of a name, a URL to this reference, and an optional description.

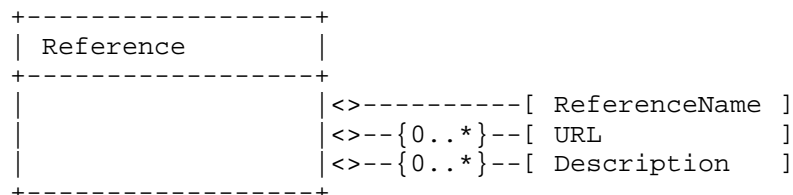


Figure 6: The Reference Class

The aggregate classes that constitute Reference:

#### ReferenceName

One. ML\_STRING. Name of the reference.

#### URL

Zero or many. URL. A URL associated with the reference.

#### Description

Zero or many. ML\_STRING. A free-form text description of this reference.

### 4.5. ReportID Class

The ReportID class represents a report tracking number that is unique in the context of the reporting organization and identifies the activity characterized in a GRCDocument. This identifier would serve as an index into the organizational reporting system. The



combination of the name attribute and the string in the element content MUST be a globally unique identifier describing the activity. Documents generated by a given organization MUST NOT reuse the same value unless they are referencing the same report instance. The ReportID class is derived from IODEF [RFC5070], Section 3.3.

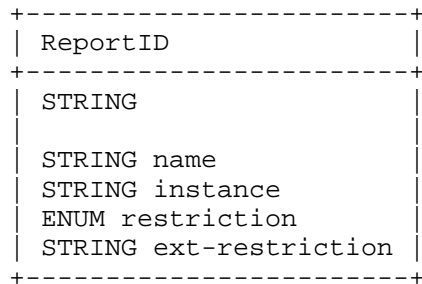


Figure 7: The ReportID Class

The ReportID class has four attributes:

#### name

Required. STRING. An identifier describing the organization that created the report. In order to have a globally unique organization name, the fully qualified domain name associated with the organization MUST be used.

#### instance

Optional. STRING. An identifier referencing a subset of the named report.

#### restriction

Optional. ENUM. This attribute follows the same guidelines as "restriction" explained in the GRCPolicy Class description in Section 4.1.

#### ext-restriction

OPTIONAL. STRING. A means by which to extend the restriction attribute. This attribute has been derived from IODEF [RFC5070] Section 5.1 and is explained in Section 5, Extending the Enumerated Values of Attributes.

#### 4.6. Contact Class

The Contact class describes contact information for organizations and personnel involved in the report exchange. This class allows for the naming of the involved party, specifying contact information for them, and identifying their role in the XML Report. The Contact class is derived from IODEF [RFC5070], Section 3.7.

People and organizations are treated interchangeably as contacts; one can be associated with the other using the recursive definition of the class (the Contact class is aggregated into the Contact class). The 'type' attribute disambiguates the type of contact information being provided.

The inheriting definition of Contact provides a way to relate information without requiring the explicit use of identifiers in the classes or duplication of data. A complete point of contact is derived by a particular traversal from the root Contact class to the leaf Contact class. As such, multiple points of contact might be specified in a single instance of a Contact class. Each child Contact class logically inherits contact information from its ancestors.

+-----+   Contact   +-----+	
ENUM role	<>-{0..1}-[ ContactName ]
STRING ext-role	<>-{0..*}-[ Description ]
ENUM type	<>-{0..*}-[ RegistryHandle ]
STRING ext-type	<>-{0..1}-[ PostalAddress ]
ENUM restriction	<>-{0..*}-[ Email ]
STRING ext-restriction	<>-{0..*}-[ Telephone ]
	<>-{0..1}-[ Fax ]
	<>-{0..1}-[ Timezone ]
	<>-{0..*}-[ AdditionalContact ]
	<>-{0..*}-[ AdditionalData ]
+-----+	

Figure 8: The Contact Class

The aggregate classes that constitute the Contact class are:

##### ContactName

Zero or one. ML\_STRING. The name of the contact. The contact may either be an organization or a person. The type attribute disambiguates the semantics.

#### Description

Zero or many. ML\_STRING. A free-form description of this contact. In the case of a person, this is often the organizational title of the individual.

#### RegistryHandle

Zero or many. A handle name into the registry of the contact.

#### PostalAddress

Zero or one. The postal address of the contact.

#### Email

Zero or many. The email address of the contact.

#### Telephone

Zero or many. The telephone number of the contact.

#### Fax

Zero or one. The facsimile telephone number of the contact.

#### Timezone

Zero or one. TIMEZONE. The timezone in which the contact resides formatted according to Section 3.4.6.

#### AdditionalContact

Zero or many. A Contact instance contained within another Contact instance inherits the values of the parent(s). This recursive definition can be used to group common data pertaining to multiple points of contact and is especially useful when listing multiple contacts at the same organization.

#### AdditionalData

Zero or many. A mechanism by which to extend the data model.

At least one of the aggregate classes MUST be present in an instance of the Contact class. This is not enforced in the GRC-Exchange schema as there is no simple way to accomplish it.

The Contact class has six attributes:

## role

Required. ENUM. Indicates the role the contact fulfills. This attribute is defined as an enumerated list:

1. creator. The entity that generate the document.
2. admin. An administrative contact for a host, network, or entity.
3. tech. A technical contact for a host or network.
4. cc. (also known as carbon-copy) An entity that is to be kept informed about the report.
5. ext-value. An escape value used to extend this attribute. This attribute has been derived from IODEF [RFC5070], Section 5.1 and is explained in Section 5, Extending the Enumerated Values of Attributes.

## ext-role

Optional. STRING. A means by which to extend the role attribute. This attribute has been derived from IODEF [RFC5070] Section 5.1 and is explained in Section 5, Extending the Enumerated Values of Attributes.

## type

Required. ENUM. Indicates the type of contact being described. This attribute is defined as an enumerated list:

1. person. The information for this contact references an individual.
2. organization. The information for this contact references an organization.
3. ext-value. An escape value used to extend this attribute. This attribute has been derived from IODEF [RFC5070], Section 5.1 and is explained in Section 5, Extending the Enumerated Values of Attributes.

## ext-type

Optional. STRING. A means by which to extend the type attribute. This attribute has been derived from IODEF [RFC5070] Section 5.1 and is explained in Section 5, Extending

the Enumerated Values of Attributes.

#### restriction

Optional. ENUM. This attribute follows the same guidelines as "restriction" used in IODEF and is explained in the GRCPolicy Class description in Section 4.1.

#### ext-restriction

OPTIONAL. STRING. A means by which to extend the restriction attribute. This attribute has been derived from IODEF [RFC5070] Section 5.1 and is explained in Section 5, Extending the Enumerated Values of Attributes.

This definition is from the IODEF specification [RFC5070], Section 3.7. This definition is fully included in the GRC-Exchange specification to prevent the need to use the IODEF schema.

#### 4.6.1. RegistryHandle Class

The RegistryHandle class represents a handle into an Internet registry or community-specific database. The handle is specified in the element content and the type attribute specifies the database. The RegistryHandle class is derived from IODEF [RFC5070], Section 3.7.1.

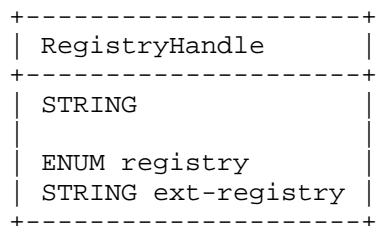


Figure 9: The RegistryHandle Class

The RegistryHandle class has two attributes:

#### registry

Required. ENUM. The database to which the handle belongs. The default value is 'local'. The possible values are:

1. internic. Internet Network Information Center

2. apnic. Asia Pacific Network Information Center
3. arin. American Registry for Internet Numbers
4. lacnic. Latin-American and Caribbean IP Address Registry
5. ripe. Reseaux IP Europeens
6. afrinic. African Internet Numbers Registry
7. local. A database local to the CSIRT
8. ext-value. An escape value used to extend this attribute. This attribute has been derived from IODEF [RFC5070], Section 5.1 and is explained in Section 5, Extending the Enumerated Values of Attributes.

#### ext-registry

Optional. STRING. A means by which to extend the registry attribute. This attribute has been derived from IODEF [RFC5070] Section 5.1 and is explained in Section 5, Extending the Enumerated Values of Attributes.

This definition is from the IODEF specification [RFC5070], Section 3.7.1. This definition is fully included in the GRC-Exchange specification to prevent the need to use the IODEF schema.

#### 4.6.2. PostalAddress Class

The PostalAddress class specifies a postal address formatted according to the POSTAL data type (Section 3.4.7).

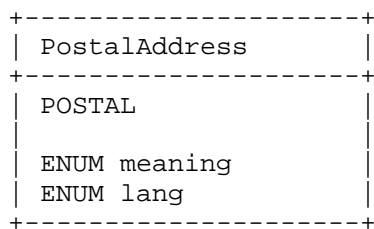


Figure 10: The PostalAddress Class

The PostalAddress class has two attributes:

meaning

Optional. ENUM. A free-form description of the element content.

lang

Required. ENUM. A valid language code per [RFC5646] constrained by the definition of "xs:language" [W3C.REC-xmlschema-1-20041028] inherited from [W3C.REC-xml-20081126].

This definition is from the IODEF specification [RFC5070], Section 3.7.2. This definition is fully included in the GRC-Exchange specification to prevent the need to use the IODEF schema.

#### 4.6.3. Email Class

The Email class specifies an email address formatted according to EMAIL data type (Section 3.4.9).

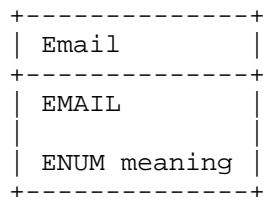


Figure 11: The Email Class

The Email class has one attribute:

meaning

Optional. ENUM. A free-form description of the element content.

This definition is from the IODEF specification [RFC5070], Section 3.7.3. This definition is fully included in the GRC-Exchange specification to prevent the need to use the IODEF schema.

#### 4.6.4. Telephone and Fax Classes

The Telephone and Fax classes specify a voice or fax telephone number respectively, and are formatted according to PHONE data type (Section 3.4.8).

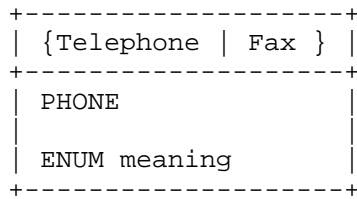


Figure 12: The Telephone and Fax Classes

The Telephone class has one attribute:

meaning

Optional. ENUM. A free-form description of the element content (e.g., hours of coverage for a given number).

This definition is from the IODEF specification [RFC5070], Section 3.7.4. This definition is fully included in the GRC-Exchange specification to prevent the need to use the IODEF schema.

#### 4.7. ExtensionType Class

The ExtensionType class serves as an extension mechanism for information not otherwise represented in the data model. For relatively simple information, atomic data types (e.g., integers, strings) are provided with a mechanism to annotate their meaning. The class can encapsulate entire XML documents conforming to an IANA registered Schema. This class is also used to provide a consistent location for the inclusion of digital signatures.

Unlike XML, which is self-describing, atomic data must be documented to convey its meaning. This information is described in the 'meaning' attribute. Since these descriptions are outside the scope of the specification, some additional coordination may be required to ensure that a recipient of a document using the ExtensionType classes can make sense of the custom extensions.



AdditionalData
ANY
ENUM dtype
STRING ext-dtype
STRING meaning
STRING formatid
ENUM restriction

Figure 13: The ExtensionType Class

The ExtensionType class has five attributes:

#### dtype

Required. ENUM. The data type of the element content. The permitted values for this attribute are shown below. The default value is "string".

1. boolean. The element content is of type BOOLEAN.
2. byte. The element content is of type BYTE.
3. character. The element content is of type CHARACTER.
4. date-time. The element content is of type DATETIME.
5. integer. The element content is of type INTEGER.
6. portlist. The element content is of type PORTLIST.
7. real. The element content is of type REAL.
8. string. The element content is of type STRING.
9. file. The element content is a base64 encoded binary file encoded as a BYTE[] type.
10. frame. The element content is a layer-2 frame encoded as a HEXBIN type.
11. packet. The element content is a layer-3 packet encoded as a HEXBIN type.

12. `ipv4-packet`. The element content is an IPv4 packet encoded as a HEXBIN type.
13. `ipv6-packet`. The element content is an IPv6 packet encoded as a HEXBIN type.
14. `path`. The element content is a file-system path encoded as a STRING type.
15. `url`. The element content is of type URL.
16. `csv`. The element content is a common separated value (CSV) list per Section 2 of [RFC4180] encoded as a STRING type.
17. `winreg`. The element content is a Windows registry key encoded as a STRING type.
18. `xml`. The element content is XML (see Section 5).
19. `ext-value`. An escape value used to extend this attribute. This attribute has been derived from IODEF [RFC5070], Section 5.1 and is explained in Section 5, Extending the Enumerated Values of Attributes.

#### `ext-dtype`

Optional. STRING. A means by which to extend the dtype attribute. This attribute has been derived from IODEF [RFC5070] Section 5.1 and is explained in Section 5, Extending the Enumerated Values of Attributes.

#### `meaning`

Optional. STRING. A free-form description of the element content.

#### `formatid`

Optional. STRING. An identifier referencing the format and semantics of the element content.

#### `restriction`

Optional. ENUM. This attribute follows the same guidelines as "restriction" explained in the GRCPolicy Class description in Section 4.1.

This definition is from the IODEF specification [RFC5070], Section 3.6. This definition is fully included in the GRC-Exchange specification to prevent the need to use the IODEF schema.

#### 4.8. Node Class

The Node class names a system (e.g., PC, router) or network.

This class was derived from IODEF [RFC5070] and is partially included in this specification. The original IODEF definition was derived from IDMEF [RFC4765].

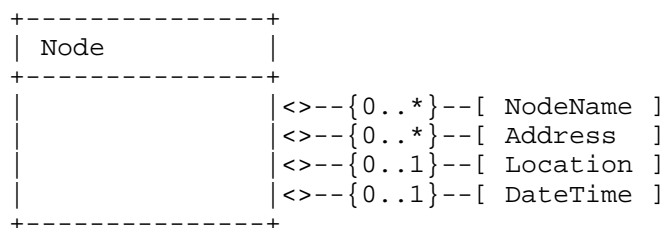


Figure 14: The Node Class

The aggregate classes that constitute Node are:

##### NodeName

Zero or more. ML\_STRING. The name of the Node (e.g., fully qualified domain name). This information **MUST** be provided if no Address information is given.

##### Address

Zero or more. The hardware, network, or application address of the Node. If a NodeName is not provided, at least one Address **MUST** be specified. This class is defined in Section 4.9.

##### Location

Zero or one. ML\_STRING. A free-form description of the physical location of the equipment.

##### DateTime

Zero or one. DATETIME. A timestamp of when the resolution between the name and address was performed. This information **SHOULD** be provided if both an Address and NodeName are specified.

#### 4.9. Address Class

The Address class represents a hardware (layer-2), network (layer-3), or application (layer-7) address.

This class was derived from IODEF [RFC5070] and is fully included in this specification. The original IODEF definition was derived from IDMEF [RFC4765].

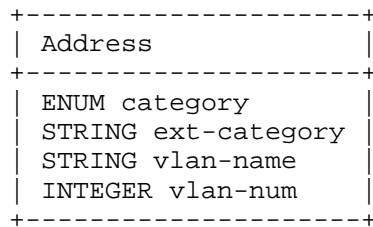


Figure 15: The Address Class

The Address class has four attributes:

category

Required. ENUM. The type of address represented. The permitted values for this attribute are shown below. The default value is "ipv4-addr".

asn. Autonomous System Number

atm. Asynchronous Transfer Mode (ATM) address

e-mail. Electronic mail address (RFC 822)

ipv4-addr. IPv4 host address in dotted-decimal notation (a.b.c.d)

ipv4-net. IPv4 network address in dotted-decimal notation, slash, significant bits (a.b.c.d/nn)

ipv4-net-mask. IPv4 network address in dotted-decimal notation, slash, network mask in dotted-decimal notation (a.b.c.d/w.x.y.z)

ipv6-addr. IPv6 host address

ipv6-net. IPv6 network address, slash, significant bits

ipv6-net-mask. IPv6 network address, slash, network mask

mac. Media Access Control (MAC) address

ext-value. An escape value used to extend this attribute. This attribute has been derived from IODEF [RFC5070], Section 5.1 and is explained in Section 5, Extending the Enumerated Values of Attributes.

ext-category

Optional. STRING. A means by which to extend the category attribute. This attribute has been derived from IODEF [RFC5070] Section 5.1 and is explained in Section 5, Extending the Enumerated Values of Attributes.

vlan-name

Optional. STRING. The name of the Virtual LAN to which the address belongs.

vlan-num

Optional. STRING. The number of the Virtual LAN to which the address belongs.

#### 4.10. GRC-Exchange Name Spaces

The GRC-Exchange schema declares a namespace of "grc-exchange-1.0" and registers it per [W3C.REC-xml-names-20091208]. Any XML instance incorporating GRC-Exchange MUST use the element GRC-Exchange in the "urn:ietf:params:xml:ns:grc-exchange-1.0" namespace. It can be referenced as follows:

```
<GRC-Exchange version="1.00" lang="en-US"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-Instance"
  xmlns:grc-exchange="urn:ietf:params:xml:ns:grc-exchange-1.0"
  xsi:schemaLocation="http://www.iana.org/assignments/xml-registry/schema
/grc-exchange-1.0.xsd">
```

#### 5. Extending the Enumerated Values of Attributes

In order to support the evolving needs of XML Schema exchanges, some extensibility is built into the GRC Report Exchange protocol. This section discusses how new attributes that have no current representation in the data model can be incorporated into GRC-Exchange. These techniques are designed so that adding new data will

not require a change to the schema. With proven value, well-documented additions can be incorporated into future versions of the specification. However, this approach also supports private additions relevant only to a closed consortium.

The data model supports a means by which to add new enumerated values to an attribute, following the method used in IODEF [RFC5070] for the same purpose. For each attribute that supports this extension technique, there is a corresponding attribute in the same element whose name is identical, less a prefix of "ext-". This special attribute is referred to as the extension attribute, and the attribute being extended is referred to as an extensible attribute. For example, an extensible attribute named "foo" will have a corresponding extension attribute named "ext-foo". An element may have many extensible, and therefore many extension, attributes. In addition to a corresponding extension attribute, each extensible attribute has "ext-value" as one its possible values. This particular value serves as an escape sequence and has no valid meaning.

In order to add a new enumerated value to an extensible attribute, the value of this attribute MUST be set to "ext-value", and the new desired value MUST be set in the corresponding extension attribute. For example, an extended instance of the type attribute of the Impact class would look as follows:

```
<Impact type="ext-value" ext-type="new-attack-type">
```

A given extension attribute MUST NOT be set unless the corresponding extensible attribute has been set to "ext-value".

## 6. GRC Report Exchange Messages

The GRC-Exchange schema is used in combination with GRC XML documents to facilitate GRC Report Exchange communications. Each message type varies slightly in format and purpose; hence, the requirements vary and are specified for each.

Note: The implementation of GRC-Exchange may automate the ability to fill in the content required for each message type from the GRC management systems involved in the message exchange.

### 6.1. Acknowledgement

Description: This message is sent in response to a Request or a Query message to provide status as to the approval of a request.

The following information is required for Acknowledgement messages and is provided through:

GRC-Exchange Information:

GRCPolicy

GRC-Exchange message type, ReportID, and destination policy information

RequestStatus class:

AuthorizationStatus of request

Standards for encryption and digital signatures [RFC3275], [W3C.REC-xmlsig-core-20080610]:

Digital signature of responding entity authenticity of GRC-Exchange Message, from the entity creating this message using an enveloped XML digital signature.

XML encryption as required by policy, agreements, and standard data markers.

A pending status is automatically generated after a 5-minute timeout without system predefined or administrator action taken to approve or deny the request. If a request is left in a pending state for more than a configurable period of time (default of 5 minutes), a response is sent to the requestor with the enumeration value set to pending. If a request is denied, the response sets the enumeration value to denied. If the request is approved, but the response will be delayed, a response MAY be sent with the enumerated value set to approved. The approved message is not mandatory, however the pending and denied message types MUST be sent if the conditions are reached.

## 6.2. Result

Description: This message provides the result of an approved Query. The Query may be used when a query is made on a group of reports or a request is made for specific details within a report. If a standard report is requested based on a specific XML schema, Request MUST be used. The details of the Query will vary depending on the included GRC XML schema. The XML schema may provide specific guidance on how queries are conducted as this specification is intended to provide a generalized structure for many types of GRC information exchanges.

The following information is required for Result messages and will be provided through:

#### GRC-Exchange Information:

##### GRCPolicy

GRC-Exchange message type, ReportID, and destination policy information

##### GRCDocument

The GRCDocument class specifies the specific GRC-Exchange XML schema that is required per the Query. The Result will include the necessary information to appropriately respond to the request.

#### GRC XML Information:

GRC XML schema elements and attributes as appropriate for the Query.

#### Standards for encryption and digital signatures [RFC3275]:

Digital signature of sending entity for authenticity of Result message, from the entity creating this message using an enveloped XML digital signature.

XML encryption as required by policy, agreements, and standard data markers.

A Result message is sent back to the requesting entity of a Query. This will include the results of the query using the appropriate XML schema named in the request. Details of what standard queries are automated in addition to the standard responses are to be detailed by the appropriate GRC communities (GRC-XML, LI-XML, etc.) in guidance documents associated with each of the relevant schemas.

### 6.3. Request

Description: The Request is used to request a report in a standardized format using the referenced XML schema in the GRCDocument class. The report requested will be the most recent report to the date and time requested.

The following information is required for Request messages and is provided through:

#### GRC-Exchange Information:



## GRCPolicy

GRC-Exchange message type, ReportID, and destination policy information

### GRC XML Information:

GRC XML schema elements and attributes as appropriate for the Request.

### Standards for encryption and digital signatures [RFC3275]:

Digital signature from initiating entity sending the GRC-Exchange message using a detached XML digital signature on the GRC-Exchange information.

Digital signature of requesting entity for authenticity of the GRC-Exchange message, from the entity sending this message using an enveloped XML digital signature on the included GRC-XML document document.

XML encryption as required by policy, agreements, and data markers.

Security requirements include the ability to encrypt [W3C.REC-xmlenc-core-20021210] the contents of the ReportRequest message using the public key of the destination entity communicating via GCR-Exchange. If no report is available for the exact date and time in the request, the most recent report details prior to the date requested will be provided. If there is no report to provide per the specified date and time, the Acknowledgement message will be sent instead setting the AuthorizationStatus to denied and providing the appropriate reason for the deny.

## 6.4. Report

Description: This message is used to provide a report using a specified GRC XML schema. This message does not require any actions to be taken, except to file the report on the receiving system or associated database. This message may be in response to a Request or sent as a regularly scheduled report.

The following information is required for Report messages and will be provided through:

### GRC-Exchange Information:

GRCPolicy GRC-Exchange message type, ReportID, and destination policy information

The following data is recommended if available and can be provided through:

GRC XML Information:

GRC XML schema elements and attributes as appropriate for the Request.

Standards for encryption and digital signatures [RFC3275]:

Digital signature from initiating entity, passed to all systems receiving the report using an enveloped XML digital signature.

XML encryption as required by policy, agreements, and standard data markers.

Security requirements include the ability to encrypt [W3C.REC-xmlenc-core-20021210] the contents of the Report message using the public key of the destination entity. Senders of a Report message should note that the information may be used to correlate information for the purpose of trending, pattern detection, etc., and may be shared with other parties unless otherwise agreed upon with the receiving entity in an established contract or agreement. Therefore, sending parties of a Report message may obfuscate or remove sensitive information before sending a Report message. A Report message may be sent either to file a report or in response to an ReportRequest, and data sensitivity must be considered in both cases.

## 6.5. Query

Description: The report Query message is used to request information from a trusted entity participating in GRC-Exchanges. The request can include the ReportID number, if known, or detailed information about the report or group of reports applicable to the query.

The following information must be used for a report Query message and is provided through:

GRC-Exchange Information:

GRCPolicy

GRC-Exchange message type, ReportID, and destination policy information

GRC XML information (optional):

GRC XML schema elements and attributes as appropriate for the report Query.

Standards for encryption and digital signatures [RFC3275]:

Digital signature from the entity initiating the GRC-Exchange message, passed to all systems receiving the report Query using an enveloped XML digital signature.

XML encryption as required by policy, agreements, and standard data markers.

The proper response to the Query message is a Result message. Security requirements include the ability to encrypt [W3C.REC-xmlenc-core-20021210] the contents of the report Request message using the public key of the destination entity communicating via GCR-Exchange. If no report is available for the exact date and time in the request, the most recent report details prior to the date requested will be provided. If there is no report to provide per the specified date and time, the Acknowledgement message will be sent instead setting the AuthorizationStatus to denied and providing the appropriate reason for the deny.

## 7. GRC-Exchange Communication Flows

The following section outlines the communication flows for GRC-Exchange and also provides examples of messages.

### 7.1. Report Communication Flow

The diagram below outlines the communication flow for a GRC-Exchange Report message sent from one entity to another. This communication flow is the simplest as no response is required. The Report may be a regularly scheduled report filing.

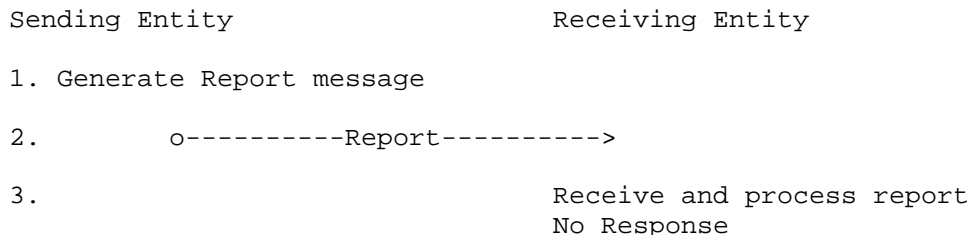


Figure 16: GRC-Exchange Report Communication Flow

The Report message MAY be encrypted [W3C.REC-xmlenc-core-20021210] for the recipient of the report depending upon the markers included in the restriction class either in the GRC-Exchange schema or in the GRC XML schema used for the report. When a report is received, the receiving entity must verify that the report has not already been filed. The ReportID and other distinguishing information in the specific report type can be used to compare with existing database entries. The Report message typically does not have a response, but the use of an Acknowledgement message is sometimes required to communicate status or error handling information.

#### 7.1.1. GRC-Exchange Report Example

The example listed is of a Report based on ...

In the following example, use of [W3C.REC-xmlsig-core-20080610] to generate digital signatures follows the guidance of XMLDsig 1.0 [W3C.REC-xmlsig-core-20080610]. XMLDsig version 1.1 [W3C.CR-xmlsig-core1-20110303] supports additional digest algorithms. Reference [RFC4051] for URIs intended for use with XML digital signatures, encryption, and canonicalization. SHA-1 SHOULD NOT be used, see [RFC6194] for further details.

Example to be provided in an updated version of this document.

#### 7.2. Request Communication Flow

The diagram below outlines the GRC-Exchange report request communication flow between participating entities. The proper response to a report Request is a Report message. If there is a problem with the request, such as a failure to validate the digital signature or decrypt the request, a Acknowledgement message is sent to the requestor. The Acknowledgement message should provide the reason why the message could not be processed.

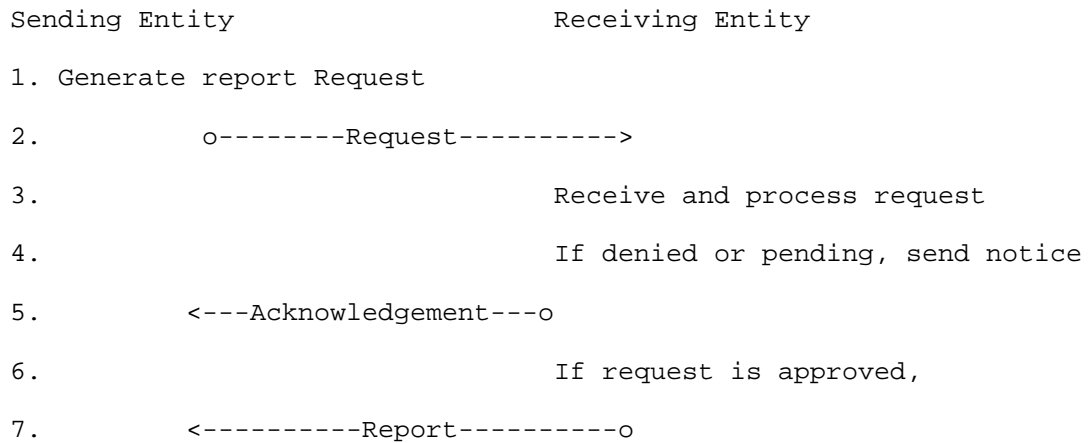


Figure 17: Request Communication Flow

#### 7.2.1. Request Example

The following example of the report Request is based on the ReportID time-based identifier tied to the specified GRC XML GRCDocument.

Example to be provided in an updated version of this document.

#### 7.2.2. Acknowledgement Message Example

The example Acknowledgement message is in response to the report Request listed above. The entity that received the request was unable to validate the digital signature used to authenticate the sending RID system.

Example to be provided in an updated version of this document.

#### 7.3. Query Communication Flow

The diagram below outlines the GRC-Exchange report Query communication flow between participating entities.

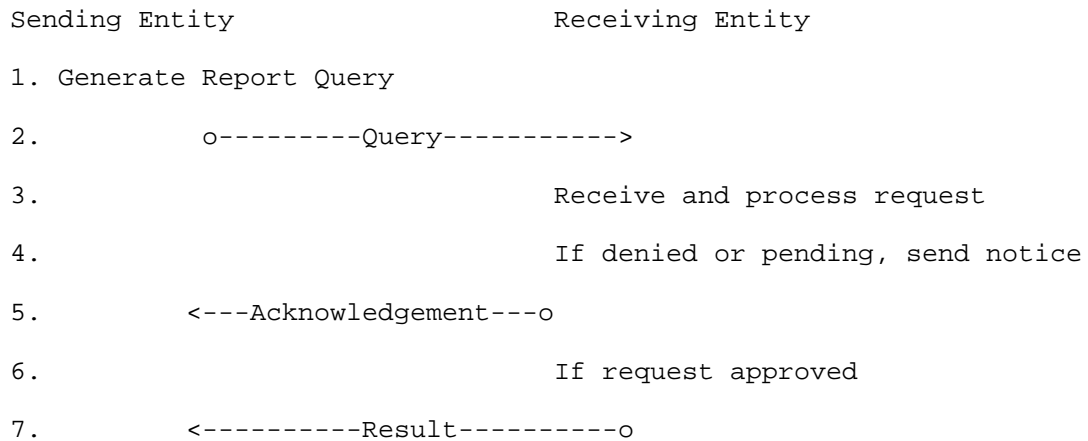


Figure 18: Query Communication Flow

The report Query communication flow is used to request specific information about a GRC report or group of reports. Information may be shared between participating entities using this format.

If there is a problem with the Query message, such as a failure to validate the digital signature [RFC3275] or decrypt the request, an Acknowledgement message is sent to the requestor. The Acknowledgement message should provide the reason why the message could not be processed.

#### 7.3.1. Query Example

The following example includes the GRC-Exchange information and an example query using an included XML schema, which is also referenced in the GRCDocument class.

Example to be provided in an updated version of this document.

#### 7.3.2. Acknowledgement Message Example

The example Acknowledgement message is in response to the Query message listed above. The entity that received the request is responding with an answer to the Query. The Result in this instance will be delayed for more than the 5-minute default time period, hence a Acknowledgement message is sent to notify of the approval status.

Example to be provided in an updated version of this document.

### 7.3.3. Result Message Example

The example Result message is in response to the Query request. This message type may be preceded by a Acknowledgement within the report Query flow of messages. It may be a direct response to a report Query request if the request is approved prior to the timeout period. This message provides a response to the request in the Query.

Example to be provided in an updated version of this document.

## 8. Internationalization Issues

Internationalization and localization is of specific concern to the GRC-Exchange, since information will often need to be exchanged across language barriers. The GRC-Exchange supports this goal by depending on XML constructs, and through explicit design choices in the data model.

GRC-Exchange documents are limited to the use of UTF-8 as it adequately provides the necessary support for internationalization. Additionally, each included document **MUST** specify the language in which their contents are encoded. The language can be specified with the attribute "xml:lang" (per Section 2.12 of [W3C.REC-xml-20081126]) in the top-level element (i.e., GRC-Exchange-Document@lang) and letting all other elements inherit that definition. All GRC-Exchange classes with a free-form text definition (i.e., all those defined of type grc-exchange:MLStringType) can also specify a language different from the rest of the document. The valid language codes for the "xml:lang" attribute are described in [RFC5646].

The data model supports multiple translations of free-form text. In the places where free-text is used for descriptive purposes, the given class always has a one-to-many cardinality to its parent (e.g., Description class). The intent is to allow the identical text to be encoded in different instances of the same class, but each being in a different language. This approach allows a GRC-Exchange document author to send recipients speaking different languages an identical document. The GRC-Exchange parser **SHOULD** extract the appropriate language relevant to the recipient.

While the intent of the data model is to provide internationalization and localization, the intent is not to do so at the detriment of

interoperability. While the GRC-Exchange does support different languages, the data model also relies heavily on standardized enumerated attributes that can crudely approximate the contents of the document. With this approach, an organization should be able to make some sense of an GRC-Exchange document it receives even if the text based data elements are written in a language unfamiliar to the consumer.

The Node class identifies a host or network device. This document re-uses the definition of Node from the IODEF specification [RFC5070], Section 3.16. However, that document did not clearly specify whether a NodeName could be an Internationalized Domain Name (IDN). GRC-Exchange systems MUST treat the NodeName class as a domain name slot [RFC5890]. GRC-Exchange systems SHOULD support IDNs in the NodeName class; if they do so, the UTF-8 representation of the domain name MUST be used, i.e., all of the domain name's labels MUST be U-labels expressed in UTF-8 or NR-LDH labels [RFC5890]; A-labels MUST NOT be used. An application communicating via GRC-Exchange can convert between A-labels and U-labels by using the Punycode encoding [RFC3492] for A-labels as described in the protocol specification for Internationalized Domain Names in Applications [RFC5891].



## 9. GRC-Exchange Schema Definition

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:grc-xml="urn:ietf:params:xml:ns:grc-xml-1.0"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
  targetNamespace="urn:ietf:params:xml:ns:grc-xml-1.0"
  elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xs:import namespace="http://www.w3.org/2000/09/xmldsig#"
    schemaLocation=
      "http://www.w3.org/TR/xmldsig-core/xmldsig-core-schema.xsd"/>

  <!-- *****
  *****
  ***      GRC Report Exchange - GRC-Exchange      ***
  ***      Namespace - grc-exchange, October 2011   ***
  ***      The namespace is defined to support transport of XML      ***
  ***      documents for exchanging GRC information.      ***
  *****
  -->
  <!--GRC-Exchange acts as an envelope for XML documents to support the
    exchange of messages-->
  <!--
  =====      GRC Report Exchange      =====
  === Suggested definition for GRC messaging =====
  -->

  *** Schema to be included here ***
```

## 10. Requirements for GRC XML Schemas for GRC-Exchange

GRC Report Exchange is a generalized version of the Real-time Inter-network Defense (RID) [RFC6545] protocol. RID leverages certain aspects of the Incident Object Description Exchange Format (IODEF) [RFC5070] schema to provide the necessary security features such as confidentiality and integrity required for the exchange of potentially sensitive information. In generalizing RID into a schema and set of message exchange flows for GRC reports, the GRC XML schemas MUST include the following: classes, elements, and attributes with enumerated values to facilitate the automated security and confidentiality concerns for GRC Report Exchange. A GRC XML schema within this document may refer to any type of XML schema used for Governance, Risk, and Compliance information or reporting. Examples

include, but are not limited to GRC-XML, LI-XML, and security automation XML schemas.

The restriction attribute, reused from IODEF [RFC5070] into GRC-Exchange, MUST be included in any individual class of a GRC XML schema that could require XML encryption [W3C.REC-xmlenc-core-20021210] just on the data contained in that class. If encryption is only required at the full document level based on the sensitivity and sharing requirements, the restriction attribute in GRC-Exchange may be sufficient.

## 11. Security Requirements

The content in this section is derived from RID [RFC6545].

### 11.1. XML Digital Signatures and Encryption

GRC-Exchange leverages existing security standards and data markings in GRCPolicy to achieve the required levels of security for the exchange of GRC information. The use of standards include TLS and the XML security features of encryption [W3C.REC-xmlenc-core-20021210] and digital signatures [RFC3275], [W3C.REC-xmlsig-core-20080610]. The standards provide clear methods to ensure that messages are secure, authenticated, and authorized, and that the messages meet policy and privacy guidelines and maintain integrity.

As specified in the relevant sections of this document, the XML digital signature [RFC3275] and XML encryption [W3C.REC-xmlenc-core-20021210] are used in the following cases:

#### XML Digital Signature

- o For all message types, the full GRC-Exchange document MUST be signed using an enveloped signature by the sending peer to provide authentication and integrity to the receiving GRC-Exchange system. The signature is placed in an instance of the Signature element.
- o XML Signature Best Practices [W3C.WD-xmlsig-bestpractices-20110809] guidance SHOULD be followed to prevent or mitigate security risks. Examples include the recommendation to authenticate a signature prior to processing (executing potentially dangerous operations) and limiting the use of URIs since they may enable cross-site scripting attacks or access to local information.

- o XML Path Language (XPath) 2.0 [W3C.REC-xpath20-20101214] MUST be followed to specify the portion of the XML document to be signed. XPath is used to specify a location within an XML document. Best practice recommendations for using XPath [W3C.WD-xmldsig-bestpractices-20110809] SHOULD be referenced to reduce the risk of denial of service attacks. The use of XSLT transforms MUST be restricted according to security guidance in [W3C.WD-xmldsig-bestpractices-20110809].

#### XML Encryption

- o The document included in GRC-Exchange messages MAY be encrypted to provide an extra layer of security between peers so that the message is not only encrypted for transport. This behavior would be agreed upon between peers or a consortium, or determined on a per-message basis, depending on security requirements. It should be noted that there are cases for transport where the GRCPolicy class needs to be presented in clear text, as detailed in the transport document [RFC6546].
- o A Request, or any other message type that may be relayed through GRC-Exchange systems before reaching the intended destination as a result of trust relationships, MAY be encrypted specifically for the intended recipient. This may be necessary if the GRC-Exchange network is being used for message transfer, the intermediate parties do not need to have knowledge of the request contents, and a direct communication path does not exist. In that case, the GRCPolicy class is used by intermediate parties and as such, GRCPolicy is maintained in clear text.
- o A message may be encrypted using the key of the request originator, while leaving the GRC-Exchange contents in clear text. In that case, the intermediate parties can view the GRCPolicy information and know a response has been provided without seeing the contents of the response. If the use of encryption were limited to sections of the message, the History class information would be encrypted. Otherwise, it is RECOMMENDED to encrypt the entire included schema plus GRC-Exchange document and use an enveloped signature, for the originator of the request. The existence of the Result message for an incident would tell any intermediate parties used in the path of the incident investigation that the incident handling has been completed.
- o The restriction attribute sets expectations for the privacy of an incident and is defined in Section 4.1. Following the guidance for XML encryption in the Security Requirements Section, the restriction attribute can be set in any of the GRC-Exchange classes to define restrictions and encryption requirements for the

exchange of GRC information. The restriction options enable encryption capabilities for the complete exchange of an XML document (including any extensions), within specific classes of a schema that embeds the restriction attribute where more limited restrictions are desired. The restriction attribute is contained in each of the GRC-Exchange classes and MUST be used in accordance with confidentiality expectations for either sections of the included XML document or the complete included XML document. Consortia and organizations should consider this guidance when creating exchange policies.

- o Expectations based on restriction setting:
  - \* If restriction is set to "private", the class or document MUST be encrypted for the recipient using XML encryption and the public key of the recipient. See Section 11.2 for a discussion on public key infrastructure (PKI) and other security requirements.
  - \* If restriction is set to "need-to-know", the class or document MUST be encrypted to ensure only those with need-to-know access can decrypt the data. The document can either be encrypted for each individual for which access is intended or a single group key may be used. The method used SHOULD adhere to any certificate policy and practices agreements between entities for the use of GRC-Exchange. A group key in this instance refers to a single key (symmetric) that is used to encrypt the block of data. The users with need-to-know access privileges may be given access to the shared key via a secure distribution method, for example, providing access to the symmetric key encrypted with each of users public keys.
  - \* If restriction is set to "public", the class or document MUST be sent in clear text. This setting can be critical if certain sections of a document or an entire document are to be shared without restrictions. This provides flexibility within an exchange to share out certain information freely where appropriate.
  - \* If restriction is set to "default", The information can be shared according to an information disclosure policy pre-arranged by the communicating parties.
- o Expectations based on placement of the restriction setting:
  - \* If restriction is set within one of the GRC-Exchange classes, the restriction applies to the entire included XML document.

- \* If restriction is set within individual classes of the included XML document, the restriction applies to the specific class and the children of that class.

The formation of policies is a very important aspect of using a messaging system like GRC-Exchange to exchange potentially sensitive information. Many considerations should be involved for peering parties, and some guidelines to protect the data, systems, and transport are covered in this section. Policies established should provide guidelines for communication methods, security, and fall-back procedures. See Sections 11.3 and Section 11.4 for additional information on consortiums and PKI considerations.

The security considerations for the storage and exchange of information in GRC-Exchange messaging may include adherence to local, regional, or national regulations in addition to the obligations to protect information. GRC-Exchange Policy is a necessary tool for listing the requirements of messages to provide a method to categorize data elements for proper handling. Controls are also provided for the sending entity to protect messages from third parties through XML encryption.

GRC-Exchange provides a method to exchange GRC request and Report messages between entities. Administrators have the ability to base decisions on the available resources and other factors of their enterprise and maintain control of GRC exchanges. Thus, GRC-Exchange provides the ability for participating networks to manage their own security controls, leveraging the information listed in GRCPolicy.

GRC-Exchange is used to transfer or exchange XML documents in an IANA registered format. Implementations SHOULD NOT download schemas at runtime due to the security implications, and included documents MUST NOT be required to provide a resolvable location of their schema.

## 11.2. Public Key Infrastructure

It is RECOMMENDED that GRC-Exchange, the XML security functions, and transport protocols properly integrate with a PKI managed by the consortium, federate PKIs within a consortium, or use a PKI managed by a trusted third party. Entities MAY use shared keys as an alternate solution, although this may limit the ability to validate certificates and could introduce risk. For the Internet, a few of examples of existing efforts that could be leveraged to provide the supporting PKI include the Regional Internet Registry's (RIR's) PKI hierarchy, vendor issued certificates, or approved issuers of Extended Validation (EV) Certificates. Security and privacy considerations related to consortiums are discussed in Sections 11.3 and Section 11.4.

The use of PKI between entities or by a consortium SHOULD adhere to any applicable certificate policy and practices agreements for the use of GRC-Exchange. [RFC3647] specifies a commonly used format for certificate policy (CP) and certification practices statements (CPS). Systems with predefined relationships for GRC-Exchange include those who peer directly or through a consortium with agreed-upon appropriate use agreements. The agreements to trust other entities may be based on assurance levels that could be determined by a comparison of the CP, CPS, and/or GRC-Exchange operating procedures. The initial comparison of policies and ability to audit controls provides a baseline assurance level for entities to form and maintain trust relationships. Trust relationships may also be defined through a bridged or hierarchical PKI in which both peers belong. If shared keys or keys issued from a common CA are used, the verification of controls to determine the assurance level to trust other entities may be limited to the GRC-Exchange policies and operating procedures.

XML security functions utilized in GRC-Exchange require a trust center such as a PKI for the distribution of credentials to provide the necessary level of security for this protocol. Layered transport protocols also utilize encryption and rely on a trust center. Public key certificate pairs issued by a trusted Certification Authority (CA) MAY be used to provide the necessary level of authentication and encryption for the GRC-Exchange protocol. The CA used for GRC-Exchange messaging must be trusted by all involved parties and may take advantage of similar efforts, such as the Internet2 federated PKI or the ARIN/RIR effort to provide a PKI to service providers. The PKI used for authentication also provides the necessary certificates needed for encryption used for the GRC-Exchange transport protocol [RFC6546].

#### 11.2.1. Authentication

Hosts receiving a GRC-Exchange message MUST be able to verify that the sender of the request is valid and trusted. Using digital signatures on a hash of the GRC-Exchange message with an X.509 version 3 certificate issued by a trusted party MUST be used to authenticate the request. The X.509 version 3 specifications as well as the digital signature specifications and path validation standards set forth in [RFC5280] MUST be followed in order to interoperate with a PKI designed for similar purposes. Full path validation verifies the chaining relationship to a trusted root and also performs a certificate revocation check. The use of digital signatures in GRC-Exchange XML messages MUST follow the World Wide Web Consortium (W3C) recommendations for signature syntax and processing when either the XML encryption [W3C.REC-xmlenc-core-20021210] or digital signature [W3C.REC-xmldsig-core-20080610], [RFC3275] is used within a document.

It might be helpful to define an extension to the authentication scheme that uses attribute certificates [RFC5755] in such a way that an application could automatically determine whether human intervention is needed to authorize a request; however, the specification of such an extension is out of scope for this document.

The use of pre-shared keys may be considered for authentication at the transport layer. If this option is selected, the specifications set forth in "Pre-Shared Key Ciphersuites for Transport Layer Security (TLS)" [RFC4279] MUST be followed. Transport specifications are detailed in a separate document [RFC6546].

#### 11.2.2. Multi-Hop Request Authentication

The use of multi-hop authentication in a Request is used when a Request is sent to multiple entities in an iterative manner. Multi-hop authentication is REQUIRED in Requests that involve multiple entities where Requests are forwarded iteratively through peers. Bilateral trust relationships MAY be used between peers, then Multi-hop authentication MUST be used for cases where the originator of a message is authenticated several hops into the message flow.

For practical reasons, entities may want to prioritize incident handling events based upon the immediate peer for a Request, the originator of a request, and other relevant information provided in metadata. In order to provide a higher assurance level of the authenticity of a Request, the originating GRC-Exchange system is included in the Request along with contact information and the information of all GRC-Exchange systems in the path the Request has taken. This information is provided through the GRC-Exchange From-Contact class nesting the list of systems and contacts involved in a request.

To provide multi-hop authentication, the originating GRC-Exchange system MUST include a digital signature in the Request sent to all systems in the upstream path. The signature MUST be passed to all parties that receive a Request, and each party MUST be able to perform full path validation on the digital signature [RFC5280]. In order to accommodate that requirement, the signed data MUST remain unchanged as a request is passed along between providers and may be restricted to one element for which the signature is applied. A second benefit to this requirement is that the integrity of the filter used is ensured as it is passed to subsequent entities in the upstream trace of the incident. The trusted PKI also provides the keys used to digitally sign the selected data element for a Request to meet the requirement of authenticating the original request. Any host in the path of the trace should be able to verify the digital signature using the trusted PKI.

In the case in which an enterprise using GRC-Exchange sends a Request to its provider, the signature from the enterprise MUST be included in the initial request. The provider may generate a new request to send upstream to members of the provider's consortium to continue the request. If the original request is sent, the originating provider, acting on behalf of the enterprise network with a request, MUST also digitally sign, with an enveloped signature, the full included XML document to assure the authenticity of the Request. A provider that offers GRC-Exchange as a service may be using its own PKI to secure GRC-Exchange communications between its GRC-Exchange system and the attached enterprise networks. Providers participating in the trace MUST be able to determine the authenticity of GRC-Exchange requests.

### 11.3. Consortia and Public Key Infrastructures

Consortia are an ideal way to establish a communication web of trust for GRC-Exchange messaging. It should be noted that direct relationships may be ideal for some communications, such as those between a provider of incident information and a subscriber of the incident reports. The consortium could provide centralized resources, such as a PKI, and established guidelines and control requirements for use of GRC-Exchange. The consortium may assist in establishing trust relationships between the participating providers to achieve the necessary level of cooperation and experience-sharing among the consortium entities. This may be established through PKI certificate policy [RFC3647] reviews to determine the appropriate trust levels between organizations or entities. The consortium may also be used for other purposes to better facilitate communication among providers in a common area (Internet, region, government, education, private networks, etc.).

Using a PKI to distribute certificates used by GRC-Exchange systems provides an already established method to link trust relationships between consortia that peer with SPs belonging to a separate consortium. In other words, consortia could peer with other consortia to enable communication of GRC-Exchange messages between the participating providers. The PKI along with Memorandums of Agreement could be used to link border directories to share public key information in a bridge, a hierarchy, or a single cross-certification relationship.

Consortia also need to establish guidelines for each participating provider to adhere. The RECOMMENDED guidelines include:

- o Physical and logical practices to protect GRC-Exchange systems;
- o Network and application layer protection for GRC-Exchange systems and communications;



- o Proper use guidelines for GRC-Exchange systems, messages, and requests; and
- o A PKI, certificate policy, and certification practices statement to provide authentication, integrity, and privacy.

The functions described for a consortium's role parallel that of a PKI federation. The PKI federations that currently exist are responsible for establishing security guidelines and PKI trust models. The trust models are used to support applications to share information using trusted methods and protocols.

A PKI can also provide the same level of security for communication between an end entity (enterprise, educational, or government customer network) and the provider.

#### 11.4. Privacy Concerns and System Use Guidelines

Information sharing typically raises many concerns especially when privacy related information may be exchanged. The GRCPolicy class is used to automate the enforcement of the privacy concerns listed within this document. The privacy and system use concerns for the system communicating GRC-Exchange messages and other integrated components include the following:

##### Service Provider Concerns:

- o Privacy information contained in Human Resources, legal, compliance and other reports.

##### Customer Attached Networks Participating in GRC-Exchange with Provider:

- o Customer networks may include an enterprise, educational, government, or other attached networks to a provider participating in GRC-Exchange. Customers should review data handling policies to understand how data will be protected by a service provider. This information will enable customers to decide what types of data at what sensitivity level can be shared with service providers. This information could be used at the application layer to establish sharing profiles for entities and groups, see Section 11.5.
- o Customers should request information on the security and privacy considerations in place by their provider and the consortium of which the provider is a member. Customers should understand if their data were to be forwarded, how might it be sanitized and how will it be protected. Customers should also understand if

limitations can be placed on how any data they share with their provider will be used in advance of sharing that data.

- o Customers should be aware that their data can and will be sent to other providers in order to complete a request unless an agreement stating otherwise is made in the service level agreements between the customer and provider. Customers considering privacy options may limit the use of this feature if they do not want the data forwarded.

#### Parties Involved in Exchanges:

- o Privacy of information such as the source and destination used for communication purposes over the monitored or GRC-Exchange connected network(s).
- o Protection of data from being viewed by intermediate parties in the path of an Request request should be considered.
- o Privacy of information exchanged in reports.

#### Consortium Considerations:

- o System use restrictions for information sharing within the local region's definitions of appropriate traffic. When participating in a consortium, appropriate use guidelines should be agreed upon and entered into contracts.
- o System use prohibiting the consortium's participating providers from inappropriately requesting information unlawfully within the jurisdiction or region.

#### Inter-Consortium Considerations:

- o System use between peering consortiums should consider any government communication regulations that apply between those two regions, such as encryption export and import restrictions.
- o System use between consortiums SHOULD NOT request information and actions beyond the scope intended and permitted by law or inter-consortium agreements.
- o System use between consortiums should consider national boundary issues and request limits in their appropriate system use agreements. Appropriate use should include restrictions to prevent the use of the protocol to limit or restrict traffic that is otherwise permitted within the country in which the peering consortium resides.

The security and privacy considerations listed above are for the consortiums, providers, and enterprises to agree upon. The agreed-upon policies may be facilitated through use of the GRCPolicy class and application layer options. Some privacy considerations are addressed through the GRC-Exchange guidelines for encryption and digital signatures as described in Section 11.1.

GRC-Exchange messaging privacy concerns should be elaborated on here...

Information shared through through GRC-Exchange could be sensitive. Such data in GRC-Exchange messages can be protected through the use of encryption [W3C.REC-xmlenc-core-20021210] enveloping the XML and GRC-Exchange document, using the public encryption key of the originating entity.

The decision is left to the system users and consortiums to determine appropriate data to be shared given that the goal of the specification is to provide the appropriate technical options to remain compliant. Local, state, or national laws may dictate the appropriate reporting requirements for specific exchange types.

Privacy becomes an issue whenever sensitive data traverses a network.

In the case of a Request or Report, where the originating provider is aware of the entity that will receive the request for processing, the free-form text areas of the document could be encrypted [W3C.REC-xmlenc-core-20021210] using the public key of the destination entity to ensure that no other entity in the path can read the contents. The encryption is accomplished through the W3C [W3C.REC-xmlenc-core-20021210] specification for encrypting an element.

GRC Report Exchanges must be legitimate incidents and not used for purposes such as sabotage or censorship. An example of such abuse of the system includes a report containing information about a competitor's compliance that may have been falsified to hurt their business.

Intra-consortium GRC-Exchange communications raise additional issues, especially when the peering consortiums reside in different regions or nations.

The GRC Report Exchange messages may be a valid use of the system within the confines of that country's network border; however, it may not be permitted to continue across network boundaries where such content is permitted under law. A continued Request, Query, or Report into a second country may break the laws and regulations of

that nation. Any such messages MUST cease at the country's border.

The privacy concerns listed in this section address issues among the trusted parties involved in a trace within an provider, a GRC-Exchange consortium, and peering GRC-Exchange consortiums. Data used for GRC-Exchange communications must also be protected from parties that are not trusted. This protection is provided through the authentication and encryption of documents as they traverse the path of trusted servers and the local security controls in place for the GRC Report Exchange systems. Each GRC-Exchange system MUST perform a bi-directional authentication when sending a GRC-Exchange message and use the public encryption key of the upstream or downstream peer to send a message or document over the network. This means that the document is decrypted and re-encrypted at each GRC-Exchange system via TLS over a transport protocol such as [RFC6546]. The GRC-Exchange messages may be decrypted at each GRC-Exchange system in order to properly process the request or relay the information. Today's processing power is more than sufficient to handle the minimal burden of encrypting and decrypting relatively small typical GRC-Exchange messages.

#### 11.5. Sharing Profiles and Policies

The application layer can be used to establish workflows and rulesets specific to sharing profiles for entities or consortiums. The profiles can leverage sharing agreements to restrict data types or classifications of data that are shared. The level of information or classification of data shared with any entity may be based on protection levels offered by the receiving entity and periodic validation of those controls. The profile may also indicate how far information can be shared according to the entity and data type. The profile can also support if requests to share data from an entity must go directly to that entity.

In some cases, pre-defined sharing profiles will be possible. These include any use case where an agreement is in place in advance of sharing. Examples may be between clients and providers, entities such as partners, or consortiums. There may be other cases when sharing profiles may not be established in advance. An organization may want to establish sharing profiles specific to possible user groups to prepare for possible incident scenarios. The user groups could include business partners, industry peers, service providers, experts not part of a service provider, law enforcement, or regulatory reporting bodies.

Workflows to approve transactions may be specific to sharing profiles and data types. Application developers should include capabilities to enable these decision points for users of the system.

Any expectations between entities to preserve the weight and admissibility of evidence should be handled at the policy and agreement level. A sharing profile may include notes or an indicator for approvers in workflows to reflect if such agreements exist.

## 12. Security Considerations

GRC Report Exchange has many security requirements and considerations built into the design of the protocol, several of which are described in the Security Requirements section. For a complete view of security, considerations include the availability, confidentiality, and integrity concerns for the transport, storage, and exchange of information.

Authenticated encrypted tunnels between systems accepting GRC-Exchange communications are used to provide confidentiality, integrity, authenticity, and privacy for the data at the transport layer. Encryption and digital signatures are also used at the GRC XML document level through GRC-Exchange options to provide confidentiality, integrity, authenticity, privacy and traceability of the document contents. Trust relationships may be through direct peers or consortiums using established trust relationships of public key infrastructure (PKI) via cross-certifications. Trust levels can be established in cross-certification processes where entities compare PKI policies that include the specific management and handling of an entity's PKI and certificates issued under that policy. [RFC3647] defines an Internet X.509 Public Key Infrastructure Certificate Policy and Certification Practices Framework that may be used in the comparison of policies to establish trust levels and agreements between entities, an entity and a consortium, and consortia. The agreements SHOULD consider key management practices including the ability to perform path validation on certificates [RFC5280], key distribution techniques [RFC2585], Certificate Authority and Registration Authority management practices.

The agreements between entities SHOULD also include a common understanding of the usage of GRC-Exchange security, policy, and privacy options discussed in this section. The formality, requirements, and complexity of the agreements for the certificate policy, practices, and the use of GRC-Exchange options SHOULD be decided by the entities or consortiums creating those agreements.

## 13. IANA Considerations

This document uses URNs to describe XML namespaces

[W3C.REC-xml-names-20091208] and XML schemas  
[W3C.REC-xmlschema-1-20041028] conforming to a registry mechanism  
described in [RFC3688].

Registration request for the grc-exchange namespace:

URI: urn:ietf:params:xml:ns:grc-exchange-1.0

Registrant Contact: See the "Author's Address" section of this document.

XML: None. Namespace URIs do not represent an XML specification.

Registration request for the grc-exchange XML schema:

URI: urn:ietf:params:xml:schema:grc-exchange-1.0

Registrant Contact: See the "Author's Address" section of this document.

XML: See Section 4, "GRC-Exchange Schema", of this document.

Request for the specified registry to be created and managed by IANA:

Name of the registry: "XML Schemas Exchanged via GRC-Exchange"

Namespace details: A registry entry for an XML Schema Transferred via GRC-Exchange consists of:

Schema Name: A short string that represents the schema referenced. This value is for reference only in the table. The version of the schema MUST be included in this string to allow for multiple versions of the same specification to be in the registry.

Version: The version of the registered XML schema. The version is a string that SHOULD be formatted as numbers separated by a '.' (period) character.

Namespace: The namespace of the referenced XML schema. This is represented in the GRC-Exchange GRCDocument class in the XMLSchemaID attribute as an enumerated value is represented by a URN or URI.

Specification URI: A URI [RFC3986] from which the registered specification can be obtained. The specification MUST be publicly available from this URI.

Information that must be provided to assign a new value: The above list of information.

Fields to record in the registry: Schema Name/Version/Namespace/Specification URI

Initial registry contents: See section Section 13

Allocation Policy: Expert Review [RFC5226] and Specification Required [RFC5226].

The Designated Expert is expected to consult with the MILE (Managed Incident Lightweight Exchange) working group or its successor if any such WG exists (e.g., via email to the working group's mailing list). The Designated Expert is expected to retrieve the XML schema specification from the provided URI in order to check the public availability of the specification and verify the correctness of the URI. An important responsibility of the Designated Expert is to ensure that the XML schema is appropriate for use in GRC-Exchange.

Request for the specified registry to be created and managed by IANA:

Name of the registry:"GRC-Exchange Enumeration List"

The registry is intended to enable enumeration value additions to attributes in the grc-exchange XML schema.

Fields to record in the registry: Attribute Name/Attribute Value/Description

Initial registry content: none.

Allocation Policy: Expert Review [RFC5226]

The Designated Expert is expected to consult with the mile (Managed Incident Lightweight Exchange) working group or its successor if any such WG exists (e.g., via email to the working group's mailing list). The Designated Expert is expected to review the request and validate the appropriateness of the enumeration for the attribute. If a draft specification is associated with the request, it MUST be reviewed by the Designated Expert.

#### 14. Acknowledgements

Many thanks to colleagues and the Internet community for reviewing and commenting on the document.

## 15. Summary

Governance, Risk, and Compliance reports may contain some of the most sensitive information for a business. Reports may contain the prioritized risks for the effective management of Business Operations, IT, Security, Compliance, and Legal departments of an enterprise. There may be a regulatory or legal requirement to share information or formatted reports with a regulatory body or other entities in a legal review. Outsourcing of computer infrastructure has necessitated the need for service providers to share reports with tenants or clients to ensure SLAs and agreements on security requirements are met. Each of these use cases require a secure method to exchange reports. GRC Report Exchange provides a standardized method to exchange reports while considering the security, privacy and policy requirements without relying on the transport layer for security. Security is provided at the document level to provide methods to share a report where policy requirements can be implemented by mapping to technical options and data markers in the GRC-Exchange protocol.

## 16. References

### 16.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2585] Housley, R. and P. Hoffman, "Internet X.509 Public Key Infrastructure Operational Protocols: FTP and HTTP", RFC 2585, May 1999.
- [RFC3275] Eastlake, D., Reagle, J., and D. Solo, "(Extensible Markup Language) XML-Signature Syntax and Processing", RFC 3275, March 2002.
- [RFC3339] Klyne, G., Ed. and C. Newman, "Date and Time on the Internet: Timestamps", RFC 3339, July 2002.
- [RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, January 2004.
- [RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, January 2005.
- [RFC4051] Eastlake, D., "Additional XML Security Uniform Resource Identifiers (URIs)", RFC 4051, April 2005.



- [RFC4279] Eronen, P. and H. Tschofenig, "Pre-Shared Key Ciphersuites for Transport Layer Security (TLS)", RFC 4279, December 2005.
- [RFC4519] Sciberras, A., "Lightweight Directory Access Protocol (LDAP): Schema for User Applications", RFC 4519, June 2006.
- [RFC5070] Danyliw, R., Meijer, J., and Y. Demchenko, "The Incident Object Description Exchange Format", RFC 5070, December 2007.
- [RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, May 2008.
- [RFC5322] Resnick, P., Ed., "Internet Message Format", RFC 5322, October 2008.
- [RFC5646] Phillips, A. and M. Davis, "Tags for Identifying Languages", BCP 47, RFC 5646, September 2009.
- [RFC5755] Farrell, S., Housley, R., and S. Turner, "An Internet Attribute Certificate Profile for Authorization", RFC 5755, January 2010.
- [RFC5890] Klensin, J., "Internationalized Domain Names for Applications (IDNA): Definitions and Document Framework", RFC 5890, August 2010.
- [RFC5891] Klensin, J., "Internationalized Domain Names in Applications (IDNA): Protocol", RFC 5891, August 2010.
- [RFC6545] Moriarty, K., "Real-time Inter-network Defense (RID)", RFC 6545, February 2012.
- [W3C.REC-xml-20081126]  
Sperberg-McQueen, C., Yergeau, F., Maler, E., Bray, T., and J. Paoli, "Extensible Markup Language (XML) 1.0 (Fifth Edition)", World Wide Web Consortium Recommendation REC-xml-20081126, November 2008,  
<<http://www.w3.org/TR/2008/REC-xml-20081126>>.
- [W3C.REC-xml-names-20091208]  
Hollander, D., Layman, A., Thompson, H., Tobin, R., and T. Bray, "Namespaces in XML 1.0 (Third Edition)", World Wide Web Consortium Recommendation REC-xml-names-20091208,

December 2009,  
<<http://www.w3.org/TR/2009/REC-xml-names-20091208>>.

[W3C.REC-xmlenc-core-20021210]  
Eastlake, D. and J. Reagle, "XML Encryption Syntax and Processing", World Wide Web Consortium Recommendation REC-xmlenc-core-20021210, December 2002,  
<<http://www.w3.org/TR/2002/REC-xmlenc-core-20021210>>.

[W3C.REC-xmlschema-1-20041028]  
Thompson, H., Beech, D., Mendelsohn, N., and M. Maloney, "XML Schema Part 1: Structures Second Edition", World Wide Web Consortium Recommendation REC-xmlschema-1-20041028, October 2004,  
<<http://www.w3.org/TR/2004/REC-xmlschema-1-20041028>>.

[W3C.REC-xmlsig-core-20080610]  
Solo, D., Roessler, T., Reagle, J., Eastlake, D., and F. Hirsch, "XML Signature Syntax and Processing (Second Edition)", World Wide Web Consortium Recommendation REC-xmlsig-core-20080610, June 2008,  
<<http://www.w3.org/TR/2008/REC-xmlsig-core-20080610>>.

[W3C.CR-xmlsig-core1-20110303]  
Reagle, J., Nystroem, M., Yiu, K., Hirsch, F., Eastlake, D., Roessler, T., and D. Solo, "XML Signature Syntax and Processing Version 1.1", World Wide Web Consortium CR-xmlsig-core1-20110303, March 2011,  
<<http://www.w3.org/TR/2011/CR-xmlsig-core1-20110303>>.

[W3C.WD-xmlsig-bestpractices-20110809]  
Datta, P. and F. Hirsch, "XML Signature Best Practices", World Wide Web Consortium WD-xmlsig-bestpractices-20110809, August 2011, <<http://www.w3.org/TR/2011/WD-xmlsig-bestpractices-20110809>>.

[W3C.REC-xpath20-20101214]  
Boag, S., Berglund, A., Kay, M., Simeon, J., Robie, J., Chamberlin, D., and M. Fernandez, "XML Path Language (XPath) 2.0 (Second Edition)", World Wide Web Consortium Recommendation REC-xpath20-20101214, December 2010,  
<<http://www.w3.org/TR/2010/REC-xpath20-20101214>>.

## 16.2. Informative References

[RFC4180] Shafranovich, Y., "Common Format and MIME Type for Comma-Separated Values (CSV) Files", RFC 4180, October 2005.

- [RFC6194] Polk, T., Chen, L., Turner, S., and P. Hoffman, "Security Considerations for the SHA-0 and SHA-1 Message-Digest Algorithms", RFC 6194, March 2011.
- [RFC3492] Costello, A., "Punycode: A Bootstring encoding of Unicode for Internationalized Domain Names in Applications (IDNA)", RFC 3492, March 2003.
- [RFC4765] Debar, H., Curry, D., and B. Feinstein, "The Intrusion Detection Message Exchange Format (IDMEF)", RFC 4765, March 2007.
- [RFC3647] Chokhani, S., Ford, W., Sabett, R., Merrill, C., and S. Wu, "Internet X.509 Public Key Infrastructure Certificate Policy and Certification Practices Framework", RFC 3647, November 2003.
- [RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.
- [ISO.8601.2000] International Organization for Standardization, "Data elements and interchange formats -- Information interchange -- Representation of dates and times", ISO Standard 8601, December 2000.

#### Authors' Addresses

Kathleen M. Moriarty  
EMC Corporation  
176 South Street  
Hopkinton, MA  
United States

Phone:  
Email: Kathleen.Moriarty@emc.com

Said Tabet  
EMC Corporation  
176 South Street  
Hopkinton, MA  
United States

Phone:  
Email: Said.Tabet@emc.com

David Waltermire  
National Institute of Standards and Technology  
100 Bureau Drive  
Gaithersburg, MD  
United States

Phone:  
Email: david.waltermire@nist.gov



MILE Working Group  
Internet-Draft  
Updates: 5070 (if approved)  
Intended status: Standards Track  
Expires: November 10, 2012

B. Trammell  
ETH Zurich  
May 9, 2012

Expert Review for IODEF Extensions in IANA XML Registry  
draft-ietf-mile-iodef-xmlreg-01.txt

Abstract

This document specifies restrictions on additions to the subset of the IANA XML Namespace and Schema registries, to require Expert Review for extensions to IODEF.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on November 10, 2012.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## 1. Introduction

IODEF extensions via class extension through `AdditionalData` and `RecordItem` elements, as per section 5.2 of [RFC5070], generally register their namespaces and schemas with the IANA XML Namespace registry at <http://www.iana.org/assignments/xml-registry/ns.html> and the IANA XML Schema registry at <http://www.iana.org/assignments/xml-registry/schema.html>, respectively [RFC3688].

In addition to schema reviews required by IANA, these registry requests should be accompanied by a review by IODEF experts to ensure the specified `AdditionalData` and/or `RecordItem` contents are compatible with IODEF and with other existing IODEF extensions. This document specifies that review.

## 2. Expert Review of IODEF-related XML Registry Entries

Changes to the XML Schema registry for schema names beginning with "urn:ietf:params:xml:schema:iodef" are subject to an additional IODEF Expert Review [RFC5226] for IODEF-correctness and -appropriateness.

The IODEF expert(s) for these reviews will be designated by the IETF Security Area Directors.

## 3. Security Considerations

This document has no security considerations.

## 4. IANA Considerations

[IANA NOTE: The authors request that IANA list an IODEF Expert be designated by the IETF Security Area Directors on the XML Schema registry, responsible for performing an IODEF Expert Review [RFC5226] on schemas with names beginning with 'urn:ietf:params:xml:schema:iodef', to be noted in the most appropriate way. This should appear in the same place as other expert designations, and note that the review is IODEF-specific. In addition, the References column of urn:ietf:params:xml:schema:iodef-1.0 at <https://www.iana.org/assignments/xml-registry/schema.html> should be made to reference this document as well as RFC5070.]

This document specifies additional expert reviews for IODEF extensions, on the XML Schema registry

(<http://www.iana.org/assignments/xml-registry/schema.html>), in Section 2.

## 5. Normative References

- [RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, January 2004.
- [RFC5070] Danyliw, R., Meijer, J., and Y. Demchenko, "The Incident Object Description Exchange Format", RFC 5070, December 2007.
- [RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.

## Author's Address

Brian Trammell  
Swiss Federal Institute of Technology Zurich  
Gloriastrasse 35  
8092 Zurich  
Switzerland

Phone: +41 44 632 70 13  
Email: [trammell@tik.ee.ethz.ch](mailto:trammell@tik.ee.ethz.ch)





MILE Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: July 18, 2014

T. Takahashi  
NICT  
K. Landfield  
McAfee  
T. Millar  
USCERT  
Y. Kadobayashi  
NAIST  
Jan 14, 2014

IODEF-extension for structured cybersecurity information  
draft-ietf-mile-sci-13.txt

Abstract

This document extends the Incident Object Description Exchange Format (IODEF) defined in RFC 5070 [RFC5070] to exchange enriched cybersecurity information among security experts at organizations and facilitates their operations. It provides a well-defined pattern to consistently embed structured information, such as identifier- and XML-based information.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on July 18, 2014.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of

publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	3
2. Terminology . . . . .	3
3. Applicability . . . . .	4
4. Extension Definition . . . . .	5
4.1. IANA Table for Structured Cybersecurity Information . . . . .	5
4.2. Extended Data Type: XMLDATA . . . . .	6
4.3. Extending IODEF . . . . .	6
4.4. Basic Structure of the Extension Classes . . . . .	7
4.5. Defining Extension Classes . . . . .	9
4.5.1. AttackPattern . . . . .	9
4.5.2. Platform . . . . .	10
4.5.3. Vulnerability . . . . .	10
4.5.4. Scoring . . . . .	11
4.5.5. Weakness . . . . .	12
4.5.6. EventReport . . . . .	13
4.5.7. Verification . . . . .	14
4.5.8. Remediation . . . . .	15
5. Mandatory to Implement features . . . . .	15
5.1. An Example XML . . . . .	16
5.2. An XML Schema for the Extension . . . . .	18
6. Security Considerations . . . . .	22
6.1. Transport-Specific Concerns . . . . .	22
6.2. Protection of Sensitive and Private Information . . . . .	23
6.3. Application and Server Security . . . . .	24
7. IANA Considerations . . . . .	24
8. Acknowledgment . . . . .	26
9. References . . . . .	26
9.1. Normative References . . . . .	26
9.2. Informative References . . . . .	27
Authors' Addresses . . . . .	29

## 1. Introduction

The number of incidents in cyber society is growing day by day. Incident information needs to be reported, exchanged, and shared among organizations in order to cope with the situation. IODEF is one of the tools already in use that enables such an exchange.

To more efficiently run security operations, information exchanged between organizations needs to be machine readable. IODEF provides a means to describe the incident information, but it often needs to include various non-structured types of incident-related data in order to convey more specific details about what is occurring. Further structure within IODEF increases the machine-readability of the document thus providing a means for better automating certain security operations.

Within the security community there exist various means for specifying structured descriptions of cybersecurity information such as [CAPEC][CCE][CCSS][CEE][CPE][CVE][CVRF][CVSS][CWE][CWSS][MAEC][OCIL][OVAL][SCAP][XCCDF]. In this context, cybersecurity information encompasses a broad range of structured data representation types that may be used to assess or report on the security posture of an asset or set of assets. Such structured descriptions facilitates a better understanding of an incident while enabling more streamlined automated security operations. Because of this, it would be beneficial to embed and convey these types of information inside IODEF documents.

This document extends IODEF to embed and convey various types of structured information. Since IODEF defines a flexible and extensible format and supports a granular level of specificity, this document defines an extension to IODEF instead of defining a new report format. For clarity, and to eliminate duplication, only the additional structures necessary for describing the exchange of such structured information are provided.

## 2. Terminology

The terminology used in this document follows the one defined in RFC 5070 [RFC5070].

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

### 3. Applicability

To maintain awareness of the continually changing security threat landscape, organization needs to exchange cybersecurity information, which includes the following information: attack pattern, platform information, vulnerability and weakness, countermeasure instruction, computer event logs, and severity assessments. IODEF provides a scheme to describe and exchange such information among interested parties. However, it does not define the detailed formats to specify such information.

There already exists structured and detailed formats for describing these types of information that can be used in facilitating such an exchange. They include [CAPEC][CCE][CCSS][CEE][CPE][CVE][CVRP][CVSS][CWE][CWSS][MAEC][OCIL][OVAL][SCAP][XCCDF]. By embedding them into the IODEF document, the document can convey more detailed context information to the receivers, and the document can be easily reused.

The use of structured information formats facilitates more advanced security operations on the receiver side. Since the information is machine readable, the data can be processed by computers thus allowing better automation of security operations.

For instance, an organization wishing to report a security incident wants to describe what vulnerability was exploited. In this case the sender can simply use IODEF, where an XML-based [XML1.0] attack pattern record that follows the syntax and vocabulary defined by an industry specification is embedded, instead of describing everything in free form text. The receiver can identify the needed details of the attack pattern by looking up some of the XML tags defined by the specification. The receiver can accumulate the attack pattern record in its database and could distribute it to the interested parties as needed, all without requiring human interventions.

In another example, an administrator is investigating an incident and detected a configuration problem that he wishes to share with a partner organization to prevent the same event from occurring. He accesses configuration information in an internal repository that was gathered prior to the initial attack specific to a new vulnerability alert to confirm the configuration was in fact vulnerable. He uses this information to automatically generate an XML-based software configuration description, embed it in an IODEF document, and send the resulting IODEF document to the partner organization.

#### 4. Extension Definition

This document extends IODEF to embed structured information by introducing new classes that can be embedded consistently inside an IODEF document as element contents of the `AdditionalData` and `RecordItem` classes.

##### 4.1. IANA Table for Structured Cybersecurity Information

This extension embeds structured cybersecurity information defined by other specifications. The list of supported specifications is managed by IANA, and this document defines the needed fields for the list's entry.

Each entry has namespace [XMLNames], specification name, version, reference URI, and applicable classes for each specification. Arbitrary URIs that may help readers to understand the specification could be embedded inside the Reference URI field, but it is recommended that standard/informational URI describing the specification is prepared and is embedded here.

The initial IANA table has only one entry, as below.

Namespace:	urn:ietf:params:xml:ns:mile:mmdef:1.2
Specification Name:	Malware Metadata Exchange Format
Version:	1.2
Reference URI:	<a href="http://standards.ieee.org/develop/indconn/icsg/mmdef.html">http://standards.ieee.org/develop/indconn/icsg/mmdef.html</a> , <a href="http://grouper.ieee.org/groups/malware/malwg/Schema1.2/">http://grouper.ieee.org/groups/malware/malwg/Schema1.2/</a>
Applicable Classes:	AttackPattern

Note that the specification was developed by The Institute of Electrical and Electronics Engineers, Incorporated (IEEE), through the Industry Connections Security Group (ICSG) of its Standards Association.

The table is to be managed by IANA following the allocation policy specified in Section 7.

The SpecID attributes of extension classes (Section 4.5) must allow the values of the specifications' namespace fields, but otherwise, implementations are not required to support all specifications of the IANA table and may choose which specifications to support, though the specification listed in the initial table needs to be minimally supported, as described in Section 5. In case an implementation

received a data it does not support, it may expand its functionality by looking up the IANA table or notify the sender of its inability to parse the data. Note that the look-up could be done manually or automatically, but automatic download of data from IANA's website is not recommended since it is not designed for mass retrieval of data by multiple devices.

#### 4.2. Extended Data Type: XMLDATA

This extension inherits all of the data types defined in the IODEF data model. One data type is added: XMLDATA. An embedded XML data is represented by the XMLDATA data type. This type is defined as the extension to the iodef:ExtensionType [RFC5070], whose dtype attribute is set to "xml".

#### 4.3. Extending IODEF

This document defines eight extension classes, namely AttackPattern, Platform, Vulnerability, Scoring, Weakness, EventReport, Verification and Remediation. Figure 1 describes the relationships between the IODEF Incident class [RFC5070] and the newly defined classes. It is expressed in Unified Modeling Language (UML) syntax as with the RFC 5070 [RFC5070]. The UML representation is for illustrative purposes only; elements are specified in XML as defined in Section 5.2.

+-----+	
Incident	
+-----+	
ENUM purpose	<>-----[IncidentID]
STRING	<>--{0..1}-[AlternativeID]
ext-purpose	<>--{0..1}-[RelatedActivity]
ENUM lang	<>--{0..1}-[DetectTime]
ENUM	<>--{0..1}-[StartTime]
restriction	<>--{0..1}-[EndTime]
	<>-----[ReportTime]
	<>--{0..*}-[Description]
	<>--{1..*}-[Assessment]
	<>--{0..*}-[Method]
	<>--{0..*}-[AdditionalData]
	<>--{0..*}-[AttackPattern]
	<>--{0..*}-[Vulnerability]
	<>--{0..*}-[Weakness]
	<>--{1..*}-[Contact]
	<>--{0..*}-[EventData]
	<>--{0..*}-[Flow]
	<>--{1..*}-[System]
	<>--{0..*}-[AdditionalData]
	<>--{0..*}-[Platform]
	<>--{0..*}-[Expectation]
	<>--{0..1}-[Record]
	<>--{1..*}-[RecordData]
	<>--{1..*}-[RecordItem]
	<>--{0..*}-[EventReport]
	<>--{0..1}-[History]
	<>--{0..*}-[AdditionalData]
	<>--{0..*}-[Verification]
	<>--{0..*}-[Remediation]
+-----+	

Figure 1: Incident class

#### 4.4. Basic Structure of the Extension Classes

Figure 2 shows the basic structure of the extension classes. Some of the extension classes have extra elements as defined in Section 4.5, but the basic structure is the same.



+-----+	
New Class Name	
+-----+	
ENUM SpecID	<>--(0..*)-[ RawData ]
STRING ext-SpecID	<>--(0..*)-[ Reference ]
STRING ContentID	
+-----+	

Figure 2: Basic Structure

Three attributes are defined as below.

**SpecID:** REQUIRED. ENUM. A specification's identifier that specifies the format of a structured information. The value should be chosen from the namespaces [XMLNames] listed in the IANA table (Section 4.1) or "private". The value "private" is prepared for conveying structured information based on a format that is not listed in the table. This is usually used for conveying data formatted according to an organization's private schema. When the value "private" is used, ext-SpecID element MUST be used.

**ext-SpecID:** OPTIONAL. STRING. A specification's identifier that specifies the format of a structured information. This is usually used to support private schema that is not listed in the IANA table (Section 4.1). This attribute MUST be used only when the value of SpecID element is "private."

**ContentID:** OPTIONAL. STRING. An identifier of a structured information. Depending on the extension classes, the content of the structured information differs. This attribute enables IODEF documents to convey the identifier of a structured information instead of conveying the information itself.

Likewise, three elements are defined as below.

**RawData:** Zero or more. XMLDATA. An XML of a structured information. This is a complete document that is formatted according to the specification and its version identified by the SpecID/ext-SpecID. When this element is used, writers/senders MUST ensure that the namespace specified by SpecID/ext-SpecID and the schema of the XML are consistent; if not, the namespace identified by SpecID SHOULD be preferred, and the inconsistency SHOULD be logged so a human can correct the problem.

**Reference:** Zero or more of iodef:Reference [RFC5070]. A reference to a structured information. This element allows an IODEF document to include a link to a structured information instead of directly embedding it into a RawData element.

Though ContentID, RawData, and Reference are optional attribute and elements, one of them MUST be used to convey structured information. Note that only one of them SHOULD be used to avoid confusing the receiver.

#### 4.5. Defining Extension Classes

This document defines the following seven extension classes.

##### 4.5.1. AttackPattern

An AttackPattern is an extension class to the Incident.Method.AdditionalData element with a dtype of "xml". It describes attack patterns of incidents or events. It is RECOMMENDED that Method class contain the extension elements whenever available. An AttackPattern class is structured as follows.

```
+-----+
| AttackPattern |
+-----+
| ENUM SpecID   | <>--(0..*)-[ RawData ]
| STRING ext-SpecID | <>--(0..*)-[ Reference ]
| STRING ContentID | <>--(0..*)-[ Platform ]
+-----+
```

Figure 3: AttackPattern class

This class has the following attributes.

SpecID: REQUIRED. ENUM. See Section 4.4.

ext-SpecID: OPTIONAL. STRING. See Section 4.4.

ContentID: OPTIONAL. STRING. An identifier of an attack pattern information. See Section 4.4.

Likewise, this class has the following elements.

RawData: Zero or more. XMLDATA. An XML of an attack pattern information. See Section 4.4.

Reference: Zero or more. A reference to an attack pattern information. See Section 4.4.

Platform: Zero or more. An identifier of software platform involved in the specific attack pattern. See Section 4.5.2.

#### 4.5.2. Platform

A Platform is an extension class that identifies a software platform. It is RECOMMENDED that AttackPattern, Vulnerability, Weakness, and System classes contain the extension elements whenever available. A Platform element is structured as follows.

```
+-----+
| Platform |
+-----+
| ENUM SpecID | <>--(0..*)-[ RawData ]
| STRING ext-SpecID | <>--(0..*)-[ Reference ]
| STRING ContentID |
+-----+
```

Figure 4: Platform class

This class has the following attributes.

SpecID: REQUIRED. ENUM. See Section 4.4.

ext-SpecID: OPTIONAL. STRING. See Section 4.4.

ContentID: OPTIONAL. STRING. An identifier of a platform information. See Section 4.4.

Likewise, this class has the following elements.

RawData: Zero or more. XMLDATA. An XML of a platform information. See Section 4.4.

Reference: Zero or more. A reference to a platform information. See Section 4.4.

#### 4.5.3. Vulnerability

A Vulnerability is an extension class to the Incident.Method.AdditionalData element with a dtype of "xml". The extension describes the vulnerabilities that are exposed or were exploited in incidents. It is RECOMMENDED that Method class contain the extension elements whenever available. A Vulnerability element is structured as follows.

```

+-----+
| Vulnerability |
+-----+
| ENUM SpecID   | <>--(0..*)-[ RawData ]
| STRING ext-SpecID | <>--(0..*)-[ Reference ]
| STRING ContentID | <>--(0..*)-[ Platform ]
|               | <>--(0..*)-[ Scoring ]
+-----+

```

Figure 5: Vulnerability class

This class has the following attributes.

SpecID: REQUIRED. ENUM. See Section 4.4.

ext-SpecID: OPTIONAL. STRING. See Section 4.4.

ContentID: OPTIONAL. STRING. An identifier of a vulnerability information. See Section 4.4.

Likewise, this class has the following elements.

RawData: Zero or more. XMLDATA. An XML of a vulnerability information. See Section 4.4.

Reference: Zero or more. A reference to a vulnerability information. See Section 4.4.

Platform: Zero or more. An identifier of software platform affected by the vulnerability. See Section 4.5.2.

Scoring: Zero or more. An indicator of the severity of the vulnerability. See Section 4.5.4.

#### 4.5.4. Scoring

A Scoring is an extension class that describes the severity scores in terms of security. It is RECOMMENDED that Vulnerability and Weakness classes contain the extension elements whenever available. A Scoring class is structured as follows.

```

+-----+
| Scoring |
+-----+
| ENUM SpecID | <>--(0..*)-[ RawData ]
| STRING ext-SpecID | <>--(0..*)-[ Reference ]
| STRING ContentID |
+-----+

```

Figure 6: Scoring class

This class has two attributes.

SpecID: REQUIRED. ENUM. See Section 4.4.

ext-SpecID: OPTIONAL. STRING. See Section 4.4.

ContentID: OPTIONAL. STRING. An identifier of a score set. See Section 4.4.

Likewise, this class has the following elements.

RawData: Zero or more. XMLDATA. An XML of a score set. See Section 4.4.

Reference: Zero or more. A reference to a score set. See Section 4.4.

#### 4.5.5. Weakness

A Weakness is an extension class to the Incident.Method.AdditionalData element with a dtype of "xml". The extension describes the weakness types that are exposed or were exploited in incidents. It is RECOMMENDED that Method class contain the extension elements whenever available. A Weakness element is structured as follows.

```

+-----+
| Weakness |
+-----+
| ENUM SpecID | <>--(0..*)-[ RawData ]
| STRING ext-SpecID | <>--(0..*)-[ Reference ]
| STRING ContentID | <>--(0..*)-[ Platform ]
| | <>--(0..*)-[ Scoring ]
+-----+

```

Figure 7: Weakness class

This class has the following attributes.

SpecID: REQUIRED. ENUM. See Section 4.4.

ext-SpecID: OPTIONAL. STRING. See Section 4.4.

ContentID: OPTIONAL. STRING. An identifier of a weakness information. See Section 4.4.

Likewise, this class has the following elements.

RawData: Zero or more. XMLDATA. An XML of a weakness information. See Section 4.4.

Reference: Zero or more. A reference to a weakness information. See Section 4.4.

Platform: Zero or more. An identifier of software platform affected by the weakness. See Section 4.5.2.

Scoring: Zero or more. An indicator of the severity of the weakness. See Section 4.5.4.

#### 4.5.6. EventReport

An EventReport is an extension class to the Incident.EventData.Record.RecordData.RecordItem element with a dtype of "xml". The extension embeds structured event reports. It is RECOMMENDED that RecordItem class contain the extension elements whenever available. An EventReport element is structured as follows.

```
+-----+
| EventReport |
+-----+
| ENUM SpecID | <>--(0..*)-[ RawData ]
| STRING ext-SpecID | <>--(0..*)-[ Reference ]
| STRING ContentID |
+-----+
```

Figure 8: EventReport class

This class has the following attributes.

SpecID: REQUIRED. ENUM. See Section 4.4.

ext-SpecID: OPTIONAL. STRING. See Section 4.4.

ContentID: OPTIONAL. STRING. An identifier of an event report.  
See Section 4.4.

Likewise, this class has the following elements.

RawData: Zero or more. XMLDATA. An XML of an event report. See  
Section 4.4.

Reference: Zero or more. A reference to an event report. See  
Section 4.4.

#### 4.5.7. Verification

A Verification is an extension class to the Incident.AdditionalData element with a dtype of "xml". The extension elements describes information on verifying security, e.g., checklist, to cope with incidents. It is RECOMMENDED that Incident class contain the extension elements whenever available. A Verification class is structured as follows.

```
+-----+
| Verification |
+-----+
| ENUM SpecID   | <>--(0..*)-[ RawData ]
| STRING ext-SpecID | <>--(0..*)-[ Reference ]
| STRING ContentID |
+-----+
```

Figure 9: Verification class

This class has the following attributes.

SpecID: REQUIRED. ENUM. See Section 4.4.

ext-SpecID: OPTIONAL. STRING. See Section 4.4.

ContentID: OPTIONAL. STRING. An identifier of a verification  
information. See Section 4.4.

Likewise, this class has the following elements.

RawData: Zero or more. XMLDATA. An XML of a verification  
information. See Section 4.4.

Reference: Zero or more. A reference to a verification information.  
See Section 4.4.

#### 4.5.8. Remediation

A Remediation is an extension class to the Incident.AdditionalData element with a dtype of "xml". The extension elements describes incident remediation information including instructions. It is RECOMMENDED that Incident class contain the extension elements whenever available. A Remediation class is structured as follows.

```
+-----+
| Remediation |
+-----+
| ENUM SpecID | <!--(0..*)-[ RawData ]
| STRING ext-SpecID | <!--(0..*)-[ Reference ]
| String ContentID |
+-----+
```

Figure 10: Remediation class

This class has the following attributes.

SpecID: REQUIRED. ENUM. See Section 4.4.

ext-SpecID: OPTIONAL. STRING. See Section 4.4.

ContentID: OPTIONAL. STRING. An identifier of a remediation information. See Section 4.4.

Likewise, this class has the following elements.

RawData: Zero or more. XMLDATA. An XML of a remediation information. See Section 4.4.

Reference: Zero or more. A reference to a remediation information.  
See Section 4.4.

### 5. Mandatory to Implement features

The implementation of this document MUST be capable of sending and receiving the XML conforming to the specification listed in the initial IANA table described in Section 4.1 without error. An SCI document is an XML document that MUST be well-formed and MUST be valid according to schemata, including extension schemata, available to the validator and applicable to the XML document. Note that the receiver can look up the namespace in the IANA table to understand



what specifications the embedded XML documents follows.

For the purpose of facilitating the understanding of mandatory to implement features, the following subsections provide an XML conformant to this document, and a schema for that.

### 5.1. An Example XML

An example IODEF document for checking implementation's MTI conformity is provided here. The document carries MMDEF metadata. Note that the metadata is generated by genMMDEF [MMDEF] with EICAR [EICAR] files.

```
<?xml version="1.0" encoding="UTF-8"?>
<IODEF-Document version="1.00" lang="en"
  xmlns="urn:ietf:params:xml:ns:iodef-1.0"
  xmlns:iodef="urn:ietf:params:xml:ns:iodef-1.0"
  xmlns:iodef-sci="urn:ietf:params:xml:ns:iodef-sci-1.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <Incident purpose="reporting">
    <IncidentID name="iodef-sci.example.com">189493</IncidentID>
    <ReportTime>2013-06-18T23:19:24+00:00</ReportTime>
    <Description>a candidate security incident</Description>
    <Assessment>
      <Impact completion="failed" type="admin" />
    </Assessment>
    <Method>
      <Description>A candidate attack event</Description>
      <AdditionalData dtype="xml">
        <iodef-sci:AttackPattern
          SpecID="http://xml/metadataSharing.xsd">
          <iodef-sci:RawData dtype="xml">
            <malwareMetaData xmlns="http://xml/metadataSharing.xsd"
              xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
              xsi:schemaLocation="http://xml/metadataSharing.xsd
                file:metadataSharing.xsd" version="1.200000" id="10000">
              <company>N/A</company>
              <author>MMDEF Generation Script</author>
              <comment>Test MMDEF v1.2 file generated using genMMDEF
                </comment>
              <timestamp>2013-03-23T15:12:50.726000</timestamp>
              <objects>
                <file id="6ce6f415d8475545be5ba114f208b0ff">
                  <md5>6ce6f415d8475545be5ba114f208b0ff</md5>
                  <sha1>da39a3ee5e6b4b0d3255bfef95601890afd80709</sha1>
                  <sha256>e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca4
                    95991b7852b855</sha256>
                  <sha512>cf83e1357eefb8bdf1542850d66d8007d620e4050b5715dc83
```

```
f4a921d36ce9ce47d0d13c5d85f2b0ff8318d2877eec2f63b9
31bd47417a81a538327af927da3e</sha512>
<size>184</size>
<filename>eicar_com.zip</filename>
<MIMEType>application/zip</MIMEType>
</file>
<file id="44d88612fea8a8f36de82e1278abb02f">
  <md5>44d88612fea8a8f36de82e1278abb02f</md5>
  <sha1>3395856ce81f2b7382dee72602f798b642f14140</sha1>
  <sha256>275a021bbfb6489e54d471899f7db9d1663fc695ec2fe2a2c4
    538aabf651fd0f</sha256>
  <sha512>cc805d5fab1fd71a4ab352a9c533e65fb2d5b885518f4e565e
    68847223b8e6b85cb48f3afad842726d99239c9e36505c64b0
    dc9a061d9e507d833277ada336ab</sha512>
  <size>68</size>
  <crc32>1750191932</crc32>
  <filename>eicar.com</filename>
  <filenameWithinInstaller>eicar.com
  </filenameWithinInstaller>
</file>
</objects>
<relationships>
  <relationship type="createdBy" id="1">
    <source>
      <ref>file[@id="6ce6f415d8475545be5ba114f208b0ff"]</ref>
    </source>
    <target>
      <ref>file[@id="44d88612fea8a8f36de82e1278abb02f"]</ref>
    </target>
    <timestamp>2013-03-23T15:12:50.744000</timestamp>
  </relationship>
</relationships>
</malwareMetaData>
</iodef-sci:RawData>
</iodef-sci:AttackPattern>
</AdditionalData>
</Method>
<Contact role="creator" type="organization">
  <ContactName>iodef-sci.example.com</ContactName>
  <RegistryHandle registry="arin">iodef-sci.example-com
  </RegistryHandle>
  <Email>contact@csirt.example.com</Email>
</Contact>
<EventData>
  <Flow>
    <System category="source">
      <Node>
        <Address category="ipv4-addr">192.0.2.200</Address>
```

```

        <Counter type="event">57</Counter>
      </Node>
    </System>
    <System category="target">
      <Node>
        <Address category="ipv4-net">192.0.2.16/28</Address>
      </Node>
      <Service ip_protocol="4">
        <Port>80</Port>
      </Service>
    </System>
  </Flow>
  <Expectation action="block-host" />
  <Expectation action="other" />
</EventData>
</Incident>
</IODEF-Document>

```

## 5.2. An XML Schema for the Extension

An XML schema describing the elements defined in this document is given here.

```

<?xml version="1.0" encoding="UTF-8"?>

<xsd:schema targetNamespace="urn:ietf:params:xml:ns:iodef-sci-1.0"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:iodef="urn:ietf:params:xml:ns:iodef-1.0"
  xmlns:iodef-sci="urn:ietf:params:xml:ns:iodef-sci-1.0"
  elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:import namespace="urn:ietf:params:xml:ns:iodef-1.0"
    schemaLocation="urn:ietf:params:xml:schema:iodef-1.0"/>

  <xsd:complexType name="XMLDATA">
    <xsd:complexContent>
      <xsd:restriction base="iodef:ExtensionType">
        <xsd:sequence>
          <xsd:any namespace="##any" processContents="lax" minOccurs="0"
            maxOccurs="unbounded"/>
        </xsd:sequence>
        <xsd:attribute name="dtype" type="iodef:dtype-type"
          use="required" fixed="xml"/>
        <xsd:attribute name="ext-dtype" type="xsd:string" use="optional"/>
        <xsd:attribute name="meaning" type="xsd:string"/>
        <xsd:attribute name="formatid" type="xsd:string"/>
        <xsd:attribute name="restriction" type="iodef:restriction-type"/>
      </xsd:restriction>
    </complexContent>
  </xsd:complexType>

```

```
</xsd:complexContent>
</xsd:complexType>

<xsd:element name="Scoring">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:choice>
        <xsd:element name="ScoreSet" type="iodef-sci:XMLDATA"
          minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="iodef:Reference" minOccurs="0"
          maxOccurs="unbounded"/>
      </xsd:choice>
    </xsd:sequence>
    <xsd:attribute name="SpecID" type="xsd:string" use="required"/>
    <xsd:attribute name="ext-SpecID" type="xsd:string"
      use="optional"/>
    <xsd:attribute name="ContentID" type="xsd:string"
      use="optional"/>
  </xsd:complexType>
</xsd:element>

<xsd:element name="AttackPattern">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:choice>
        <xsd:element name="RawData" type="iodef-sci:XMLDATA"
          minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="iodef:Reference" minOccurs="0"
          maxOccurs="unbounded"/>
      </xsd:choice>
      <xsd:element ref="iodef-sci:Platform" minOccurs="0"
        maxOccurs="unbounded"/>
    </xsd:sequence>
    <xsd:attribute name="SpecID" type="xsd:string" use="required"/>
    <xsd:attribute name="ext-SpecID" type="xsd:string"
      use="optional"/>
    <xsd:attribute name="ContentID" type="xsd:string"
      use="optional"/>
  </xsd:complexType>
</xsd:element>

<xsd:element name="Vulnerability">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:choice>
        <xsd:element name="RawData" type="iodef-sci:XMLDATA"
          minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="iodef:Reference" minOccurs="0"
          maxOccurs="unbounded"/>
      </xsd:choice>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

```
        maxOccurs="unbounded"/>
    </xsd:choice>
    <xsd:element ref="iodef-sci:Platform" minOccurs="0"
        maxOccurs="unbounded"/>
    <xsd:element ref="iodef-sci:Scoring" minOccurs="0"
        maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:attribute name="SpecID" type="xsd:string" use="required"/>
<xsd:attribute name="ext-SpecID" type="xsd:string"
    use="optional"/>
<xsd:attribute name="ContentID" type="xsd:string"
    use="optional"/>
</xsd:complexType>
</xsd:element>

<xsd:element name="Weakness">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:choice>
                <xsd:element name="RawData" type="iodef-sci:XMLDATA"
                    minOccurs="0" maxOccurs="unbounded"/>
                <xsd:element ref="iodef:Reference" minOccurs="0"
                    maxOccurs="unbounded"/>
            </xsd:choice>
            <xsd:element ref="iodef-sci:Platform" minOccurs="0"
                maxOccurs="unbounded"/>
            <xsd:element ref="iodef-sci:Scoring" minOccurs="0"
                maxOccurs="unbounded"/>
        </xsd:sequence>
        <xsd:attribute name="SpecID" type="xsd:string" use="required"/>
        <xsd:attribute name="ext-SpecID" type="xsd:string"
            use="optional"/>
        <xsd:attribute name="ContentID" type="xsd:string"
            use="optional"/>
    </xsd:complexType>
</xsd:element>

<xsd:element name="Platform">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:choice>
                <xsd:element name="RawData" type="iodef-sci:XMLDATA"
                    minOccurs="0" maxOccurs="unbounded"/>
                <xsd:element ref="iodef:Reference" minOccurs="0"
                    maxOccurs="unbounded"/>
            </xsd:choice>
        </xsd:sequence>
        <xsd:attribute name="SpecID" type="xsd:string" use="required"/>
    </xsd:complexType>
</xsd:element>
```

```
<xsd:attribute name="ext-SpecID" type="xsd:string"
  use="optional"/>
<xsd:attribute name="ContentID" type="xsd:string"
  use="optional"/>
</xsd:complexType>
</xsd:element>

<xsd:element name="EventReport">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:choice>
        <xsd:element name="RawData" type="iodef-sci:XMLDATA"
          minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="iodef:Reference" minOccurs="0"
          maxOccurs="unbounded"/>
      </xsd:choice>
    </xsd:sequence>
    <xsd:attribute name="SpecID" type="xsd:string" use="required"/>
    <xsd:attribute name="ext-SpecID" type="xsd:string"
      use="optional"/>
    <xsd:attribute name="ContentID" type="xsd:string"
      use="optional"/>
  </xsd:complexType>
</xsd:element>

<xsd:element name="Verification">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:choice>
        <xsd:element name="RawData" type="iodef-sci:XMLDATA"
          minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="iodef:Reference" minOccurs="0"
          maxOccurs="unbounded"/>
      </xsd:choice>
    </xsd:sequence>
    <xsd:attribute name="SpecID" type="xsd:string" use="required"/>
    <xsd:attribute name="ext-SpecID" type="xsd:string"
      use="optional"/>
    <xsd:attribute name="ContentID" type="xsd:string"
      use="optional"/>
  </xsd:complexType>
</xsd:element>

<xsd:element name="Remediation">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:choice>
        <xsd:element name="RawData" type="iodef-sci:XMLDATA"
```

```
        minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="iodef:Reference" minOccurs="0"
          maxOccurs="unbounded"/>
      </xsd:choice>
    </xsd:sequence>
    <xsd:attribute name="SpecID" type="xsd:string" use="required"/>
    <xsd:attribute name="ext-SpecID" type="xsd:string"
      use="optional"/>
    <xsd:attribute name="ContentID" type="xsd:string"
      use="optional"/>
  </xsd:complexType>
</xsd:element>

</xsd:schema>
```

## 6. Security Considerations

This document specifies a format for encoding a particular class of security incidents appropriate for exchange across organizations. As merely a data representation, it does not directly introduce security issues. However, it is guaranteed that parties exchanging instances of this specification will have certain concerns. For this reason, the underlying message format and transport protocol used MUST ensure the appropriate degree of confidentiality, integrity, and authenticity for the specific environment. Specific security considerations are detailed in the messaging and transport documents, where the exchange of formatted information is automated. See Real-time Inter-network Defense (RID) [RFC6545] Section 9 for a detailed overview of security requirements and considerations.

It is RECOMMENDED that organizations who exchange data using this document develop operating procedures that minimally consider the following areas of concern.

### 6.1. Transport-Specific Concerns

The underlying messaging format, IODEF, provides data markers to indicate the sensitivity level of specific classes within the structure as well as for the entire XML document. The "restriction" attribute accomplishes this with four attribute values in IODEF. These values are RECOMMENDED for use at the application level, prior to transport, to protect data as appropriate. A standard mechanism to apply XML encryption using these attribute values as triggers is defined in RID [RFC6545] Section 9.1. This mechanism may be used whether or not the RID and RID Transport binding [RFC6546] are used in the exchange to provide object level security on the data to prevent possible intermediary systems or middle-boxes from having

access to the data being exchanged. In areas where transmission security or secrecy is questionable, the application of a XML digital signature [xmldsig] and/or encryption on each report will counteract both of these concerns. The data markers are RECOMMENDED for use by applications for managing access controls, however access controls and management of those controls are out-of-scope for this document. Options such as the usage of a standard language (e.g. XACML [XACML]) for the expression of authorization policies can be used to enable source and destination systems to better coordinate and align their respective policy expressions.

Any transport protocol used to exchange instances of IODEF documents MUST provide appropriate guarantees of confidentiality, integrity, and authenticity. The use of a standardized security protocol is encouraged. The RID protocol [RFC6545] and its associated transport binding [RFC6546] provide such security with options for mutual authentication session encryption and include application levels concerns such as policy and work flow.

The critical security concerns are that these structured information may be falsified, accessed by unintended entities, or they may become corrupt during transit. We expect that each exchanging organization will determine the need, and mechanism, for transport protection.

## 6.2. Protection of Sensitive and Private Information

For a complete review of privacy considerations when transporting incident related information, please see RID [RFC6545] Section 9.5. Whether or not the RID protocol is used, the privacy considerations are important to consider as incident information is often sensitive and may contain privacy related information about individuals/organizations or endpoints involved. Often times, organizations will require legal review and formal policies to be established which outline specific details of what information can be exchanged with specific entities. Typically, identifying information is anonymized where possible and appropriate. In some cases, information brokers are used to further anonymize the source of exchanged information so that other entities are unaware of the origin of a detected threat, whether or not that threat was realized.

It is RECOMMENDED that policies and procedures for the exchange of cybersecurity information are established prior to participation in data exchanges. Policy and workflow procedures for the exchange of cybersecurity information often require executive level approvals and legal reviews to appropriately establish limits on what information can be exchanged with specific organizations. RID [RFC6545] Section 9.6 outlines options and considerations for application developers to consider for the policy and workflow design.



### 6.3. Application and Server Security

The Cybersecurity Information extension is merely a data format. Applications and transport protocols that store or exchange IODEF documents using information that can be represented through this extension will be a target for attacks. It is RECOMMENDED that systems and applications storing or exchanging this information are properly secured, have minimal services enabled, maintain access controls and monitoring procedures.

## 7. IANA Considerations

This document uses URNs to describe XML namespaces and XML schemata [XMLschemaPart1] [XMLschemaPart2] conforming to a registry mechanism described in [RFC3688].

Registration request for the IODEF structured cybersecurity information extension namespace:

URI: urn:ietf:params:xml:ns:iodef-sci-1.0

Registrant Contact: Refer here to the authors' addresses section of the document.

XML: None.

Registration request for the IODEF structured cybersecurity information extension XML schema:

URI: urn:ietf:params:xml:schema:iodef-sci-1.0

Registrant Contact: Refer here to the authors' addresses section of the document.

XML: Refer here to the XML Schema in Section 5.2.

This memo creates the following registry for IANA to manage:

Name of the registry: "Structured Cybersecurity Information (SCI) specifications"

Name of its parent registry: "Incident Object Description Exchange Format (IODEF)"

URL address of the registry: <http://www.iana.org/assignments/iodef>

Namespace details: A registry entry for a Structured Cybersecurity Information Specification (SCI specification) consists of:

Namespace: A URI [RFC3986] that identifies the XML namespace used by the registered SCI specification. In the case where the registrant does not request a particular URI, the IANA will assign it a Uniform Resource Name (URN) that follows RFC 3553 [RFC3553]

Specification Name: A string containing the spelled-out name of the SCI specification in human-readable form.

Reference URI: A list of one or more of the URIs [RFC3986] from which the registered specification can be obtained. The registered specification MUST be readily and publicly available from that URI.

Applicable Classes: A list of one or more of the extension classes specified in Section 4.5 of this document. The registered SCI specification MUST only be used with the extension classes in the registry entry.

Information that must be provided to assign a new value: The above list of information.

Fields to record in the registry: Namespace/Specification Name/Version/Reference URI/Applicable Classes. Note that it is not necessary to include defining reference for all assignments in this new registry.

Initial registry contents: only one entry with the following values.

Namespace: urn:ietf:params:xml:ns:mile:mmdef:1.0

Specification Name: Malware Metadata Exchange Format

Version: 1.2

Reference URI: <http://standards.ieee.org/develop/indconn/icsg/mmdef.html>, <http://grouper.ieee.org/groups/malware/malwg/Schemal.2/>

Applicable Classes: AttackPattern

Allocation Policy: Specification Required (which includes Expert Review) [RFC5226].

The Designated Expert is expected to consult with the mile (Managed Incident Lightweight Exchange) working group or its successor if any such WG exists (e.g., via email to the working group's mailing list). The Designated Expert is expected to retrieve the SCI specification from the provided URI in order to check the public availability of the specification and verify the correctness of the URI. An important responsibility of the Designated Expert is to ensure that the registered Applicable Classes are appropriate for the registered SCI specification.

## 8. Acknowledgment

We would like to acknowledge David Black from EMC, who kindly provided generous support, especially on the IANA registry issues. We also would like to thank Jon Baker from MITRE, Eric Burger from Georgetown University, Paul Cichonski from NIST, Panos Kampanakis from CISCO, Pearl Liang from IANA, Ivan Kirillov from MITRE, Robert Martin from MITRE, Alexey Melnikov from Isode, Kathleen Moriarty from EMC, Lagadec Philippe from NATO, Sean Turner from IECA Inc., Shuhei Yamaguchi from NICT, Anthony Rutkowski from Yaana Technology, Brian Trammell from ETH Zurich, David Waltermire from NIST, and James Wendorf from IEEE, for their sincere discussion and feedback on this document.

## 9. References

### 9.1. Normative References

- [MMDEF] IEEE ICSG Malware Metadata Exchange Format Working Group, "Malware Metadata Exchange Format".
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, January 2005.
- [RFC5070] Danyliw, R., Meijer, J., and Y. Demchenko, "The Incident Object Description Exchange Format", RFC 5070, December 2007.
- [RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.

- [RFC6545] Moriarty, K., "Real-time Inter-network Defense (RID)", RFC 6545, April 2012.
- [RFC6546] Trammell, B., "Transport of Real-time Inter-network Defense (RID) Messages over HTTP/TLS", RFC 6546, April 2012.
- [XML1.0] Bray, T., Maler, E., Paoli, J., Sperberg-McQueen, C., and F. Yergeau, "Extensible Markup Language (XML) 1.0 (Fifth Edition)", W3C Recommendation, November 2008.
- [XMLSchemaPart1] Thompson, H., Beech, D., Maloney, M., and N. Mendelsohn, "XML Schema Part 1: Structures Second Edition", W3C Recommendation, October 2004.
- [XMLSchemaPart2] Biron, P. and A. Malhotra, "XML Schema Part 2: Datatypes Second Edition", W3C Recommendation, October 2004.
- [XMLNames] Bray, T., Hollander, D., Layman, A., Tobin, R., and H. Thomson, "Namespaces in XML (Third Edition)", W3C Recommendation, December 2009.

## 9.2. Informative References

- [RFC3339] Klyne, G., Ed. and C. Newman, "Date and Time on the Internet: Timestamps", RFC 3339, July 2002.
- [RFC3552] Rescorla, E. and B. Korver, "Guidelines for Writing RFC Text on Security Considerations", BCP 72, RFC 3552, July 2003.
- [RFC3553] Mealling, M., Masinter, L., Hardie, T., and G. Klyne, "An IETF URN Sub-namespace for Registered Protocol Parameters", BCP 73, RFC 3553, June 2003.
- [RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, January 2004.
- [RFC5322] Resnick, P., Ed., "Internet Message Format", RFC 5322, October 2008.
- [RFC6116] Bradner, S., Conroy, L., and K. Fujiwara, "The E.164 to Uniform Resource Identifiers (URI) Dynamic Delegation Discovery System (DDDS) Application (ENUM)", RFC 6116, March 2011.

- [CAPEC] The MITRE Corporation, "Common Attack Pattern Enumeration and Classification (CAPEC)".
- [CCE] The MITRE Corporation, "Common Configuration Enumeration (CCE)".
- [CCSS] Scarfone, K. and P. Mell, "The Common Configuration Scoring System (CCSS)", NIST Interagency Report 7502, December 2010.
- [CEE] The MITRE Corporation, "Common Event Expression (CEE)".
- [CPE] National Institute of Standards and Technology, "Common Platform Enumeration", June 2011.
- [CVE] The MITRE Corporation, "Common Vulnerability and Exposures (CVE)".
- [CVRF] ICASI, "Common Vulnerability Reporting Framework (CVRF)".
- [CVSS] Peter Mell, Karen Scarfone, and Sasha Romanosky, "The Common Vulnerability Scoring System (CVSS) and Its Applicability to Federal Agency Systems".
- [CWE] The MITRE Corporation, "Common Weakness Enumeration (CWE)".
- [CWSS] The MITRE Corporation, "Common Weakness Scoring System (CWSS)".
- [EICAR] European Expert Group for IT-Security, "Anti-Malware Testfile", 2003.
- [MAEC] The MITRE Corporation, "Malware Attribute Enumeration and Characterization".
- [OCIL] David Waltermire and Karen Scarfone and Maria Casipe, "The Open Checklist Interactive Language (OCIL) Version 2.0", April 2011.
- [OVAL] The MITRE Corporation, "Open Vulnerability and Assessment Language (OVAL)".
- [SCAP] Waltermire, D., Quinn, S., Scarfone, K., and A. Halbardier, "The Technical Specification for the Security Content Automation Protocol (SCAP): SCAP Version 1.2", NIST Special Publication 800-126 Revision 2, September 2011.

- [XACML] Rissanen, E., "eXtensible Access Control Markup Language (XACML) Version 3.0", January 2013, <<http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.pdf>>.
- [XCCDF] David Waltermire and Charles Schmidt and Karen Scarfone and Neal Ziring, "Specification for the Extensible Configuration Checklist Description Format (XCCDF) version 1.2 (DRAFT)", July 2011.
- [xmldsig] W3C Recommendation, "XML Signature Syntax and Processing (Second Edition)", June 2008.

#### Authors' Addresses

Takeshi Takahashi  
National Institute of Information and Communications Technology  
4-2-1 Nukui-Kitamachi Koganei  
184-8795 Tokyo  
Japan

Phone: +80 423 27 5862  
Email: [takeshi\\_takahashi@nict.go.jp](mailto:takeshi_takahashi@nict.go.jp)

Kent Landfield  
McAfee, Inc  
5000 Headquarters Drive  
Plano, TX 75024  
USA

Email: [Kent\\_Landfield@McAfee.com](mailto:Kent_Landfield@McAfee.com)

Thomas Millar  
US Department of Homeland Security, NPPD/CS&C/NCSD/US-CERT  
245 Murray Lane SW, Building 410, MS #732  
Washington, DC 20598  
USA

Phone: +1 888 282 0870  
Email: [thomas.millar@us-cert.gov](mailto:thomas.millar@us-cert.gov)

Youki Kadobayashi  
Nara Institute of Science and Technology  
8916-5 Takayama, Ikoma  
630-0192 Nara  
Japan

Email: youki-k@is.aist-nara.ac.jp





mile Working Group  
Internet-Draft  
Intended status: Informational  
Expires: December 10, 2012

B. Trammell  
ETH Zurich  
June 8, 2012

Guidelines and Template for Defining Extensions to IODEF  
draft-ietf-mile-template-05.txt

Abstract

This document provides guidelines for extensions to the Incident Object Description Exchange Format (IODEF) [RFC5070] for exchange of incident management data, and contains a template for Internet-Drafts describing those extensions, in order to ease the work and improve the quality of extension descriptions.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on December 10, 2012.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	3
2. Applicability of Extensions to IODEF . . . . .	3
3. Selecting a Mechanism for IODEF Extension . . . . .	4
4. Security Considerations . . . . .	6
5. IANA Considerations . . . . .	6
6. Acknowledgments . . . . .	6
7. References . . . . .	6
7.1. Normative References . . . . .	6
7.2. Informative References . . . . .	6
Appendix A. Document Template . . . . .	7
A.1. Introduction . . . . .	7
A.2. Terminology . . . . .	7
A.3. Applicability . . . . .	7
A.4. Extension Definition . . . . .	8
A.5. Security Considerations . . . . .	8
A.6. IANA Considerations . . . . .	9
A.7. Manageability Considerations . . . . .	10
A.8. Appendix A: XML Schema Definition for Extension . . . . .	10
A.9. Appendix B: Examples . . . . .	10
Appendix B. Example Enumerated Type Extension Definition: Presentation Action . . . . .	10
Appendix C. Example Element Definition: Test . . . . .	11
Author's Address . . . . .	12

## 1. Introduction

In the five years since the specification of IODEF [RFC5070], the threat environment has evolved, as has the practice of cooperative network defense. These trends, along with experience gained through implementation and deployment, have indicated the need to extend IODEF. This document provides guidelines for defining these extensions. It starts by describing the applicability of IODEF extensions, and the IODEF extension mechanisms, before providing a section (Appendix A) that is itself designed to be copied out and filled in as the starting point for an Internet-Draft about an IODEF extension.

This document is designed to give guidance on the extension of IODEF, especially for those extension authors who may be new to the IETF process. Nothing in this document should be construed as defining policies for the definition of these extensions.

At publication time, the Managed Incident Lightweight Exchange (MILE) working group of the IETF provides a home for work on IODEF extensions that do not otherwise have a natural home. IODEF extensions that require the expertise of other IETF working groups or other standards development organizations may be done within those groups with consultation of IODEF experts, such as those appointed for review as in [I-D.ietf-mile-iodef-xmlreg].

## 2. Applicability of Extensions to IODEF

Before deciding to extend IODEF, the first step is to determine whether an IODEF extension is a good fit for a given problem. There are two sides to this question:

1. Does the problem involve the reporting or sharing of observations, indications, or other information about an incident, whether in progress or completed, hypothetical or real? "Incident" is defined in the terminology for the original IODEF requirements [RFC3067]: an event that involves a security violation, whether a single attack of a group thereof. If the answer to this question is unequivocally "No", then IODEF is probably not a good choice as a base technology for the application area.
2. Can IODEF adequately represent information about the incident without extension? IODEF has a rich set of incident-relevant classes. If, after detailed examination of the problem area and the IODEF specification, and consultation with IODEF experts, the answer to this question is "Yes", then extension is not

necessary.

Examples of such extensions to IODEF might include:

- o Leveraging existing work in describing aspects of incidents to make IODEF more expressive, by standardized reference to external information bases about incidents and incident-related information
- o Allowing the description of new types of entities (e.g., related actors) or new types of characteristics of entities (e.g., information related to financial services) involved in an IODEF incident report
- o Allowing the representation of new types of indicators, observables, or incidents in an IODEF incident report
- o Allowing additional semantic or metadata labeling of IODEF Documents (e.g., for handling or disposition instructions, or compliance with data protection and data retention regulations)

### 3. Selecting a Mechanism for IODEF Extension

IODEF was designed to be extended through any combination of:

1. extending the enumerated values of Attributes, as per section 5.1 of [RFC5070];
2. class extension through AdditionalData or RecordItem elements, as per section 5.2 of [RFC5070]; and/or
3. containment of the IODEF-Document element within an external XML Document, itself containing extension data, as done by RID [RFC6545].

Note that in this final case, the extension will not be directly interoperable with IODEF implementations, and must "unwrap" the IODEF document from its container; nevertheless, this may be appropriate for certain use cases involving integration of IODEF within external schemas. Extensions using containment of an IODEF-Document are not further treated in this document, though the document template in Appendix A may be of some use in defining them.

Certain attributes containing enumerated values within certain IODEF elements may be extended. For an attribute named "foo", this is achieved by giving the value of "foo" as "ext-value", and adding a new attribute named "ext-foo" containing the extended value. The attributes which can be extended this way are limited to the

following, denoted in 'Element@attribute' format, referencing the section in which they are defined in [RFC5070]:

- Incident@purpose, section 3.2
- AdditionalData@dtype, section 3.6
- Contact@role, section 3.7
- Contact@type, section 3.7
- RegistryHandle@registry, section 3.7.1
- Impact@type, section 3.10.1
- TimeImpact@metric, section 3.10.2
- TimeImpact@duration, section 3.10.2
- HistoryItem@action, section 3.11.1
- Expectation@action, section 3.13
- System@category, section 3.15
- Counter@type, section 3.16.1
- Counter@duration, section 3.16.1
- Address@category, section 3.16.2
- NodeRole@category, section 3.16.3
- RecordPattern@type, section 3.19.2
- RecordPattern@offsetunit, section 3.19.2
- RecordItem@dtype, section 3.19.3

Note that this list is current as of publication time; the set of IODEF Data Types may be extended by future specifications which update [RFC5070].

An example definition of an attribute extension is given in Appendix B.

IODEF documents can contain extended scalar or XML data using an AdditionalData element or a RecordItem element. Scalar data extensions must set the "dtype" attribute of the containing element to the data type to reference one of the IODEF data types as enumerated in section 2 of [RFC5070], and should use the "meaning" and "formatid" attributes to explain the content of the element.

XML extensions within an AdditionalData or RecordItem element use a dtype of "xml", and should define a schema for the topmost containing element within the AdditionalData or RecordItem element. An example definition of an element definition is given in Appendix C.

When adding elements to the AdditionalData section of an IODEF document, an extension's namespace and schema should be registered with IANA; see Appendix A.6 for details.

#### 4. Security Considerations

This document raises no security issues itself. Extensions defined using the template in Appendix A need to provide an analysis of security issues they may raise. See Appendix A.5 for details.

#### 5. IANA Considerations

This document contains no considerations for IANA.

#### 6. Acknowledgments

Thanks to David Black, Benoit Claise, Martin Duerst, Eran Hammer, Tom Millar, Kathleen Moriarty, Peter Saint-Andre, Robert Sparks, Takeshi Takahashi, Sean Turner, Samuel Weiler, and Peter Yee, for their reviews and comments. This work is materially supported by the European Union Seventh Framework Program under grant agreement 257315 (DEMONS).

#### 7. References

##### 7.1. Normative References

- [RFC5070] Danyliw, R., Meijer, J., and Y. Demchenko, "The Incident Object Description Exchange Format", RFC 5070, December 2007.

##### 7.2. Informative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC3067] Arvidsson, J., Cormack, A., Demchenko, Y., and J. Meijer, "TERENA'S Incident Object Description and Exchange Format Requirements", RFC 3067, February 2001.
- [RFC3552] Rescorla, E. and B. Korver, "Guidelines for Writing RFC Text on Security Considerations", BCP 72, RFC 3552, July 2003.
- [RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.
- [RFC5706] Harrington, D., "Guidelines for Considering Operations and

Management of New Protocols and Protocol Extensions",  
RFC 5706, November 2009.

[RFC6545] Moriarty, K., "Real-time Inter-network Defense (RID)",  
RFC 6545, April 2012.

[I-D.ietf-mile-iodef-xmlreg]  
Trammell, B., "Expert Review for IODEF Extensions in IANA  
XML Registry", draft-ietf-mile-iodef-xmlreg-01 (work in  
progress), May 2012.

## Appendix A. Document Template

The document template given in this section is provided as a starting point for writing an Internet-Draft describing an IODEF extension. RFCs are subject to additional formatting requirements and must contain additional sections not described in this template; consult the RFC Editor style guide (<http://www.rfc-editor.org/styleguide.html>) for more information.

This template is informational in nature; in case of any future conflict with RFC Editor requirements for Internet-Drafts, those requirements take precedence.

### A.1. Introduction

The introduction section introduces the problem being solved by the extension, and motivates the development and deployment of the extension.

### A.2. Terminology

The terminology section introduces and defines terms specific to the document. Terminology from [RFC5070] or [RFC6545] should be referenced in this section, but not redefined or copied. If [RFC2119] terms are used in the document, this should be noted in the terminology section.

### A.3. Applicability

The applicability section defines the use cases to which the extension is applicable, and details any requirements analysis done during the development of the extension. The primary goal of this section is to allow readers to see if an extension is indeed intended to solve a given problem. This section should also define and restrict the scope of the extension, as appropriate, by pointing out any non-obvious situations to which it is not intended to apply.

In addition to defining the applicability, this section may also present example situations, which should then be detailed in the examples section, below.

#### A.4. Extension Definition

This section defines the extension.

Extensions to enumerated types are defined in one subsection for each attribute to be extended, enumerating the new values with an explanation of the meaning of each new value. An example enumeration extension is shown in Appendix B, below.

Element extensions are defined in one subsection for each element, in top-down order, from the element contained within AdditionalData or RecordItem; an example element extension is shown in Appendix C, below. Each element should be described by a UML diagram as in Figure 1, followed by a description of each of the attributes, and a short description of each of the child elements. Child elements should then be defined in a subsequent subsection, if not already defined in the IODEF document itself, or in another referenced IODEF extension document.

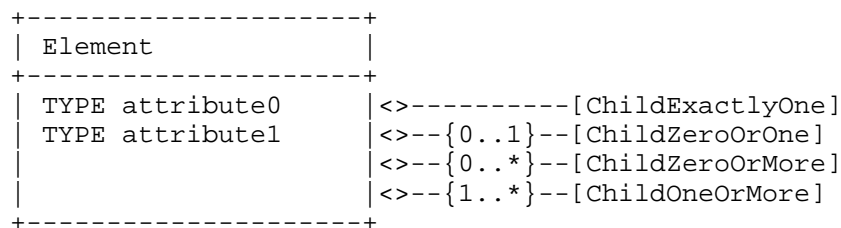


Figure 1: Example UML Element Diagram

Elements containing child elements should indicate the multiplicity of those child elements, as shown in the figure above. Allowable TYPES are enumerated in section 2 of [RFC5070].

#### A.5. Security Considerations

[SECDIR and RFC-EDITOR NOTE: Despite the title, this section is NOT a Security Considerations section, rather a template Security Considerations section for future extension documents to be built from this template. See Section 4 for Security Considerations for this document.]

Any security considerations [RFC3552] raised by this extension or its deployment should be detailed in this section. Guidance should focus



on ensuring the users of this extension do so in a secure fashion, with special attention to non-obvious implications of the transmission of the information represented by this extension. [RFC3552] may be a useful reference in determining what to cover in this section. This section is required by the RFC Editor.

It should also be noted in this section that the security considerations for IODEF [RFC5070] apply to the extension as well.

#### A.6. IANA Considerations

[IANA and RFC-EDITOR NOTE: Despite the title, this section is NOT an IANA Considerations section, rather a template IANA Considerations section for future extension documents to be built from this template. See Section 5 for IANA Considerations for this document.]

Any IANA considerations [RFC5226] for the document should be detailed in this section. Note that IODEF extension documents will generally register new namespaces and schemas. In addition, this section is required by the RFC Editor, so if there are no IANA considerations, the section should exist and contain the text "this document has no actions for IANA".

IODEF Extensions which represent an enumeration should reference an existing IANA registry or subregistry for the values of that enumeration. If no such registry exists, this section should define a new registry to hold the enumeration's values, and define the policies by which additions may be made to the registry.

IODEF Extensions adding elements to the AdditionalData section of an IODEF document should register their own namespaces and schemas for extensions with IANA; therefore, this section should contain at least a registration request for the namespace and the schema, as follows, modified as appropriate for the extension:

Registration request for the IODEF My-Extension namespace:

URI: urn:ietf:params:xml:ns:iodef-myextension-1.0

Registrant Contact: Refer here to the authors' addresses section of the document, or to an organizational contact in the case of an extension supported by an external organization.

XML: None

Registration request for the IODEF My-Extension XML schema:

URI: urn:ietf:params:xml:schema:iodef-myextension-1.0

Registrant Contact: Refer here to the authors' addresses section of the document, or to an organizational contact in the case of an extension supported by an external organization.

XML: Refer here to the XML Schema in Appendix A of the document, or to a well-known external reference in the case of an extension with an externally-defined schema.

#### A.7. Manageability Considerations

If any of the operational and/or management considerations listed in Appendix A of [RFC5706] apply to the extension, address them in this section. If no such considerations apply, this section can be omitted.

#### A.8. Appendix A: XML Schema Definition for Extension

The XML Schema describing the elements defined in the Extension Definition section is given here. Each of the examples in Appendix A.9 will be verified to validate against this schema by automated tools.

#### A.9. Appendix B: Examples

This section contains example IODEF-Documents illustrating the extension. If example situations are outlined in the applicability section, documents for those examples should be provided in the same order as in the applicability section. Example documents will be tested to validate against the schema given in the appendix.

#### Appendix B. Example Enumerated Type Extension Definition: Presentation Action

This example extends the IODEF Expectation element to represent the expectation that a slide deck be derived from the IODEF Incident, and that a presentation be given by the recipient's organization thereon.

Attribute: Expectation@action

Extended value(s): give-a-presentation

Value meaning: generate a slide deck from the provided incident information and give a presentation thereon.

Additional considerations: the format of the slide deck is left to the recipient to determine in accordance with its established practices for the presentation of incident reports.

## Appendix C. Example Element Definition: Test

This example defines the Test class for labeling IODEF test data.

The Test class is intended to be included within an AdditionalData element in an IODEF Document. If a Test element is present, it indicates that an IODEF Document contains test data, not a information about a real incident.

The Test class contains information about how the test data was generated.

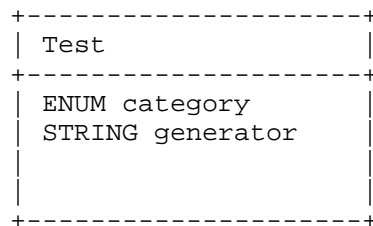


Figure 2: The Test class

The Test class has two attributes:

category: Required. ENUM. The type of test data. The permitted values for this attribute are shown below. The default value is "unspecified".

1. unspecified. The document contains test data, but no further information is available.
2. internal. The test data is intended for the internal use of an implementor, and should not be distributed or used outside the context in which it was generated.
3. unit. The test data is intended for unit testing of an implementation, and may be included with the implementation to support this as part of the build and deployment process.
4. interoperability. The test data is intended for interoperability testing of an implementation, and may be freely shared to support this purpose.

generator: Optional. STRING. A free-form string identifying the person, entity, or program which generated the test data.

Author's Address

Brian Trammell  
Swiss Federal Institute of Technology Zurich  
Gloriastrasse 35  
8092 Zurich  
Switzerland

Phone: +41 44 632 70 13  
Email: trammell@tik.ee.ethz.ch

