

Individual Submission
Internet-Draft
Intended status: Standards Track
Expires: March 25, 2012

K. Goodier
L-3 Com
D. Rajnovic
Cisco
September 22, 2011

Guidelines for Extensions to IODEF for Managed Incident Lightweight
Exchange
draft-goodier-mile-data-markers-00.txt

Abstract

This document provides extensions to Managed Incident Lightweight Exchange (MILE). MILE describes a subset of Incident Object Description Exchange Format (IODEF) defined in RFC 5070. The Data Markers extension is aimed at exchanging data tags or markers that label categories of information that have significance in the exchange of incident information. These data marker extension is aimed at exchanging data tags or markers that label information exchanged during incident handling. Data markers include sensitivity and data handling requirements that can prevent possible criminal errors in mismarking data. Both network and information security incidents typically result in the loss of service, data, and resources both human and system. Existing extensions to the IODEF-Document Class for Reporting Phishing [RFC 5901] have already been introduced for network security incidents. Data markers introduce extensions for information security incidents so that network providers and Computer Security Incident Response Teams (CSIRT) are equipped and ready to assist in communicating and tracing security incidents with tools and procedures in place before the occurrence of an attack. Data Markers also support Real-time Inter-network Defense (RID) [RFC 6045] that outlines a proactive inter-network communication method to facilitate sharing incident handling data while integrating existing detection, tracing, source identification, and mitigation mechanisms for a complete incident handling solution. Combining these capabilities in a communication system provides a way to achieve higher security levels on networks. Policy guidelines for handling incidents are recommended and can be agreed upon by a consortium using the security recommendations and considerations.

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 March 25, 2012.

Copyright Notice

Copyright (c) 2011 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. Terminology	4
2. Applicability of Data Marker Extensions to IODEF	4
2.1. Applicability	4
2.2. Extension Definition	7
2.2.1. IODEF Data Types	7
2.2.2. Example Enumerated Type Extension Definition: E.164 Address	8
2.2.3. Example Element Definition: Test	9
2.3. Examples	10
2.4. Security Considerations	10
2.5. IANA Considerations	10
2.6. Appendix: XML Schema Definition for Extension	11
3. Security Considerations	11
4. IANA Considerations	11
5. References	11
5.1. Normative References	11
5.2. Informative References	12
Authors' Addresses	12

1. Introduction

Guidance has improved for the handling of privacy and other data markers to ensure the consistent application of security controls in profiled implementations. Organizations require help from data marking parties to digitally define the related context and semantics. Rapid incident detection and coordination requires automation. Increases in internet engineering outsourcing via cloud services requires tighter security and knowledge on how data is protected and incidents that could affect that data. It is critical to have automated means to both detect and mitigate/stop attack traffic while augmenting the governance of any internet-based enterprise. Enterprise data marking standards that secure cyberspace particularly within private and hybrid networks require governance methodologies that enable a workforce who has a responsibility to locate and retrieve data in support of Lines of Businesses (LoBs) and specific missions. Data markers provide additional semantic or metadata labeling of IODEF Documents (e.g., for handling or disposition instructions, or compliance with data protection and data retention regulations).

1.1. Terminology

Data marker attributes containing enumerated values within IODEF elements may be further extended. For a data marker attribute named "foo", this is achieved by giving the value of "foo" as "ext-marker-value", and adding a new attribute named "ext-marker-foo" containing the extended value. The attributes which can be extended in this way are defined in [RFC5070], and limited by these values.

2. Applicability of Data Marker 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 answers to this question for data markers:

1. Data markers are critical to the reporting or sharing of information about an incident.
2. Without a data makers extension, IODEF can not adequately represent information about an incident.

2.1. Applicability

The five standard use cases that apply to the data markers extension follow:

1. Use Case 1:Information Sharing
2. Use Case 2:Incident Query
3. Use Case 3:Investigation Request-Results Sent
4. Use Case 4:Investigation Request-Request Sent
5. Use Case 5:Trace Back Request

Use Case 1: Information Sharing: An incident type is identified and marked and CSIRTs would like to share that information with other CSIRTs. The incident information may be a list of IP addresses known to be malicious or a type of an attack described (for example) in MAEC and embedded in an IODEF document. In this use case, a central authority, US CERT, may have knowledge of several instances of an attack type for which the supported community should be notified to increase awareness and detection capabilities for the attack type or sources. Information Sharing Flow: US CERT generates an IODEF document, using the relevant SCAP and marked data information sources, and sends a RID Report message out to all Agency CSIRTs with one or more attack type descriptions or information about malicious entities. No response is required for this communication type.

Use Case 2: Incident Query: An incident query communication is used when one CSIRT would like to know if a type of attack has been detected by other CSIRTs. The information provided back can be limited to descriptions of the attack without providing source and destination information if that data is marked as controlled or classified. This use case is sending the request to US CERT because they may have a broad knowledge set of attack types within the government sector to share with Agency CSIRTs. Incident Query Flow: An Agency CSIRT sends an appropriately marked IncidentQuery to US CERT. US CERT responds with an appropriately marked Report message.

Use Case 3: Investigation Request-Results Sent: An incident is detected by a CSIRT and further investigation is required to identify and mitigate or stop the attack. In this use case instance, the Agency CSIRT will detect and data mark the incident. It could be identified by any CSIRT including US CERT or the Provider CSIRT in other use cases. Investigation Request Flow: An Agency CSIRT detects an incident. The source of the incident is identified using SCAP and event information and an IODEF document with data markers with data markers is generated. The IODEF document is sent to the Provider CSIRT in a RID Investigation message using the appropriate transport protocol and data markers. The Provider CSIRT decides to work on the incident investigation, then sends the properly data marked Result message when the investigation is complete. Note: The Result message

can contain the information deemed appropriate for sharing with the Agency CSIRT. Data markers for policy and privacy considerations relative to the incident are required. In this use case, the Provider CSIRT sends the full investigation Report including the source of the attack and the action taken to stop the attack, traffic from the source address was blocked.

Use Case 4: Investigation Request-Request Sent: An incident is detected by a CSIRT and further investigation is required to identify and mitigate or stop the attack. In this use case instance, the Agency CSIRT will detect the incident. It could be identified by any CSIRT including US CERT or the Provider CSIRT in other use cases. Investigation Request Flow: An Agency CSIRT detects an incident. The source of the incident is identified using SCAP and event information and an IODEF document with data markers is generated. The IODEF document is sent to the Provider CSIRT in a RID Investigation message using the appropriate transport protocol and data markers. The Provider CSIRT is unable to work on the Investigation request, a RequestAuthorization message is sent to the Agency CSIRT to notify them of the inability to respond at this time. The Agency CSIRT takes an action to block the source address from accessing the application that was targeted using the tools available to them from the Provider.

Use Case 5: Trace Back Request: In the case where the source of an incident is unknown (possibly spoofed), the ability to iteratively track an incident through providers or networks may be necessary. This communication flow is similar to the Investigation request, but could involve multiple CSIRTs until a source is found or a party does not have the resources to participate. The actions taken in this case may be close to the source of an attack or downstream Provider depending on who cooperates and marks data. This use case just describes one of the many possible flows that could occur in the trace back request. Trace Back Request Flow: An Agency CSIRT detects an incident using event information and the appropriate information for that event (application server is targeted in a DDoS attack). The Agency CSIRT generates an IODEF document and encapsulates it in a RID wrapper with data markers for a TraceRequest. The TraceRequest is sent to the upstream to their Provider's CSIRT. The Provider CSIRT confirms receipt with a RequestAuthorization message indicating that this can be looked at now by the Provider CSIRT. The investigation begins at the Provider CSIRT, and the next upstream provider has been found (where the traffic is originating), a TraceRequest message is sent to the next Provider CSIRT. The next Provider CSIRT sends a RequestAuthorization response to both the Agency CSIRT (originator of request) and the Provider CSIRT who sent the TraceRequest. The response provided in the AuthorizationRequest is yes and the incident will be investigated. The investigation has

completed and a Result message is sent to the Agency CSIRT. The information provided in the report must be marked according to the policy of the CSIRT that sends the report. In this use case, the information provided is limited to a description of the actions taken with appropriate data markers, the traffic has been rate limited with no information on the true source of the attack.

2.2. Extension Definition

This section defines the data markers 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 the new value. An example enumeration extension is shown in Section 2.2.2, 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 Section 2.2.3, 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 MILE extension document.

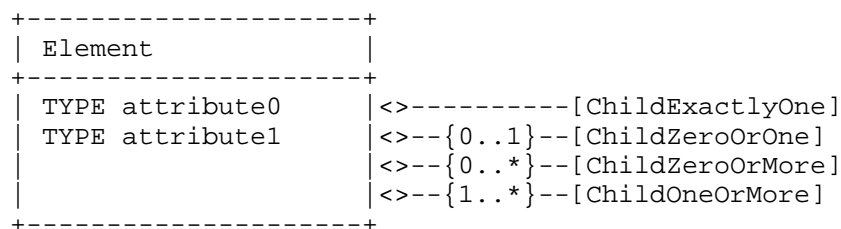


Figure 1: Example UML Element Diagram

Elements containing child elements should indicate the multiplicity of those child elements, as shown in the figure above. Allowable TYPEs are discussed in the following subsection.

2.2.1. IODEF Data Types

The allowable TYPEs for attributes within IODEF are enumerated in section 2 of [RFC5070], and consist of:

- o INTEGER

- o REAL
- o CHARACTER
- o STRING
- o ML_STRING (for strings in encodings other than that of the enclosing document)
- o BYTE for bytes or byte vectors in Base 64 encoding
- o HEXBIN for bytes in ascii-hexadecimal encoding
- o ENUM for enumerated types; allowable values of the enumeration must be defined in the attribute definition
- o DATETIME for ISO 8601:2000 [RFC3339] encoded timestamps
- o TIMEZONE for timezones as encoded in section 2.9 of [RFC5070].
- o PORTLIST for port lists as encoded in section 2.10 of [RFC5070].
- o POSTAL for postal addresses as defined in section 2.23 of [RFC4519].
- o NAME for names of natural or legal persons as defined in section 2.3 of [RFC4519].
- o PHONE for telephone numbers as defined in section 2.35 of [RFC4519].
- o EMAIL for email addresses as defined in section 3.4.1. of [RFC2822].
- o URL for URLs as in [RFC2396].

In addition to these simple data types, IODEF provides a compound data type for representing network address information. Addresses included within an extension element should be represented by containing an IODEF:Address element, which supports IPv4 and [RFC2373] IPv6 addresses, as well as MAC, ATM, and BGP autonomous system numbers. Application-layer addresses should be represented with the URL simple attribute type, instead.

2.2.2. Example Enumerated Type Extension Definition: E.164 Address

This example extends the IODEF Address element to support the encoding of ENUM-mapped telephone numbers [RFC6116].

Attribute: Address@category

Extended value(s): enum-e164

Content format: An E.164 telephone number encoded as a domain name in the e164.int space, e.g. "2.1.2.1.5.5.5.2.1.2.1.e164.int." for +1 212 555 1212, as per section 3.2 of [RFC6116].

Additional considerations: none.

2.2.3. 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 reference to a real incident.

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

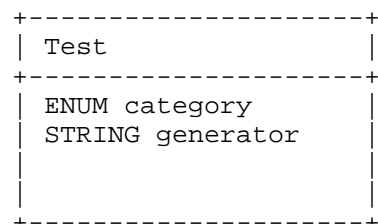


Figure 2: The Test class

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.

2.3. 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 should be tested to validate against the schema given in the appendix.

2.4. 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 3 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 or storage of the information represented by an extension.

2.5. 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 4 for IANA Considerations for this document.]

Any IANA considerations [RFC5226] for the document should be detailed in this section; if none, the section should exist and contain the text "this document has no actions for IANA".

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 the appendix of the document, or to a well-known external reference in the case of an extension with an externally-defined schema.

2.6. Appendix: 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 section Section 2.3 should be verified to validate against this schema by automated tools.

3. Security Considerations

This document defines a template for MILE extensions to the IODEF and RID documents; as such, it has no security considerations on its own.

4. IANA Considerations

This section will be updated.

5. References

5.1. Normative References

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

- [RFC6045] Moriarty, K., "Real-time Inter-network Defense (RID)", RFC 6045, November 2010.

5.2. Informative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2373] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 2373, July 1998.
- [RFC2396] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifiers (URI): Generic Syntax", RFC 2396, August 1998.
- [RFC2822] Resnick, P., "Internet Message Format", RFC 2822, April 2001.
- [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.
- [RFC4519] Sciberras, A., "Lightweight Directory Access Protocol (LDAP): Schema for User Applications", RFC 4519, June 2006.
- [RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 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.

Authors' Addresses

Dr. Katherine S. Goodier
L-3 Communications"
2720 Technology Drive
Annapolis Junction
USA

Phone: +01 301 547 7043
Email: katherine.goodier@l-3com.com

Damir Rajnovic
Cisco

Email: gaus@cisco.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 9, 2012

K. Moriarty
S. Tabet
EMC
October 7, 2011

GRC Report Exchange
draft-moriarty-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 RFC6045-bis, 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 April 9, 2012.

Copyright Notice

Copyright (c) 2011 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
3.5. GRC Report Exchange Message Types	8
4. GRC-Exchange Schema	8
4.1. GRCPolicy Class	11
4.2. RequestStatus	17
4.3. ReportSchema	19
4.4. Reference Class	20
4.5. GRC-Exchange Name Spaces	21
5. Extending the Enumerated Values of Attributes	21
6. GRC Report Exchange Messages	22
6.1. RequestAuthorization	22
6.2. Result	23
6.3. ReportRequest	24
6.4. Report	25
6.5. ReportQuery	26
7. GRC-Exchange Communication Flows	27
7.1. Report Communication Flow	27
7.1.1. GRC-Exchange Report Example	27
7.2. Report Request Communication Flow	28
7.2.1. ReportRequest Example	28
7.2.2. RequestAuthorization Message Example	28
7.3. ReportQuery Communication Flow	29
7.3.1. ReportQuery Example	29
7.3.2. RequestAuthorization Message Example	30
7.3.3. Result Message Example	30
8. GRC-Exchange Schema Definition	30

9. Requirements for GRC XML Schemas for GRC-Exchange	31
10. Security Requirements	32
10.1. XML Digital Signatures and Encryption	32
10.2. Message Transport	35
10.3. Message Delivery Protocol - Integrity and Authentication	35
10.4. Transport Communication	36
10.5. Authentication of The GRC Report Exchange Protocol	37
10.5.1. Multi-Hop Authentication	38
10.6. Consortiums and Public Key Infrastructures	38
10.7. Privacy Concerns and System Use Guidelines	39
11. Security Considerations	41
12. IANA Considerations	42
13. Summary	42
14. References	43
14.1. Normative References	43
14.2. Informative References	44
Authors' Addresses	45

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 incident information, enabling the exchange of GRC extensible markup language (XML) documents. GRC Report Exchange considers security, policy, and privacy issues related to the exchange of potentially sensitive information, enabling organizations accepting GRC report filings, service providers, or enterprises the options to make appropriate decisions according to their policies. GRC Report Exchange includes provisions for confidentiality, integrity, and authentication.

The data in GRC reports to be included in GRC Report Exchange are represented in an XML [XML1.0] document using the appropriate XML schema for the GRC report and the GRC Report Exchange schema. 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 [RFC6046] [RFC6046]. 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 [XMLschema] 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 XXXX.

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 ms likely scenario for the initial use cases of GRC report exchange.

3.1. Inter-network Provider GRC Messaging

GRC Report Exchange provides a standard protocol and format is required to ensure inter-operability between vendors for the exchange of GRC reports and GRC filing of reports. GRC Report Exchange provides the framework necessary for communication between entities filing or exchanging GRC reports. Several message types described in Section 5 are necessary to facilitate the exchange or filing of reports. The message types include the Report, ReportQuery, RequestAuthorization, Result, and the ReportRequest 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 ReportQuery message is used to request information on a particular report. The RequestAuthorization and Report messages are used to communicate the status and result of a ReportQuery or ReportRequest request.

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, 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 be noted in a confidence rating based on relationships and experience working with peers. The confidence ratings must adhere to specifications for selecting the percentage used to avoid abuse of the system.

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

3.4. GRC Report Exchange Data Types

GRC Report Exchange is derived from the RID and IODEF data models and inherits all of the data types defined in those models.

3.4.1. Boolean

A boolean value is represented by the BOOLEAN data type.

The BOOLEAN data type is implemented as "xs:boolean" [XMLschema] in the schema.

This Section will be expanded when the schema is further finalized

3.5. GRC Report Exchange Message Types

The five GRC Report Exchange message types are as follows:

1. RequestAuthorization. This message is sent to the requestor of a report (ReportRequest) or in response to a ReportQuery 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 (ReportRequest) or in response to ReportQuery. The Result may contain the full report requested or a section of the report as appropriate for the request in the ReportQuery.
3. ReportRequest. This message type is used to request a specific type of GRC report. The ReportRequest 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. ReportQuery. 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 ReportRequest or ReportQuery; the ReportSchema 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-Exchange is a separate schema that includes the appropriate GRC XML schema, the GRC-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-Exchange is maintained separate from the GRC XML schema used to send or request a report. This segregation enables flexibility for GRC-Exchange to be used with any GRC XML schema and removes the need for decrypting and parsing the entire GRC Report and GRC-Exchange document to determine how it should be handled at each entity communicating via GRC-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-Exchange messages.

The policy information and guidelines are discussed in Section 9.7. 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-Exchange schema is defined as follows:

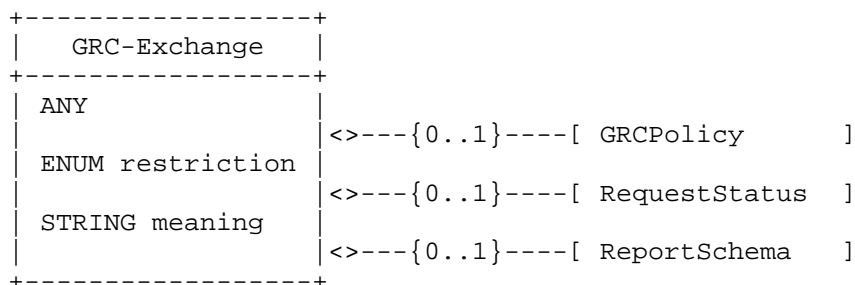


Figure 2: The GRC-Exchange Schema

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 RequestAuthorization messages to report back to the entity requesting a report or sending a ReportQuery if the request is denied or remains in a pending state.

ReportSchema

Zero or One. The ReportSchema 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 two attributes as follows:

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 Incident class. In other classes where this attribute is used, no default is specified.

This attribute is derived from IODEF [RFC5070] and 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.

meaning

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

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 [RFC6046], as that information should be as small as possible and may not be encrypted. The RequestStatus message 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 [RFC6046].

4.1. GRCPolicy Class

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

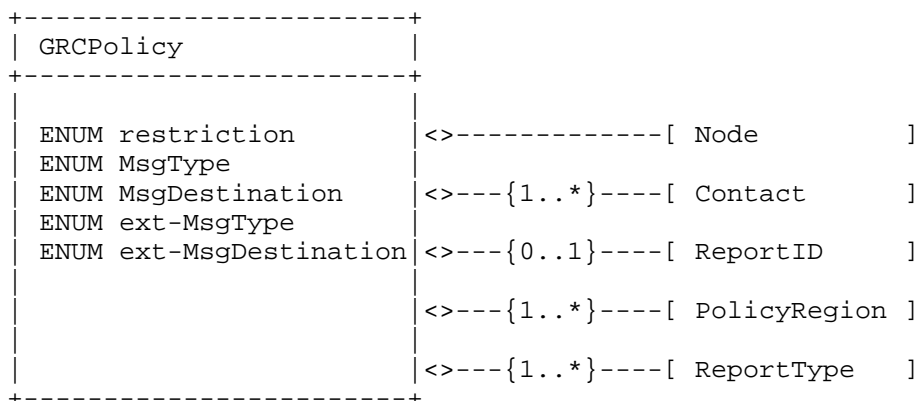


Figure 3: The GRCPolicy Class

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

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 to prevent the need to use the IODEF schema.

Contact

One or more. Contact information for the parties involved in the GRC Report Exchange. 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.

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.

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. AcrossNationalBoundaries. This selection must be set if the message type will cross national boundaries. There could be instances of reports or report requests where that may not be known in advance, but should be known at the instance where boundaries are crossed during the communication exchange. This must also be set if the security requirements of the request is based upon regulations of a specific nation that may not apply to all nations. The stricter requirements should be upheld. This is different from the "BetweenConsortiums" setting since it may be possible to have

multiple nations as members of the same consortium, and this option must be selected if the report information may have different restrictions in other nations.

7. LawEnforcement. This option is used when GRC information is exchanged with LawEnforcement, local government, or other authorities requesting or receiving a report. Examples may include eDiscovery or eWarrant use cases. If the law enforcement agency resides within a different nation than the sending entity, the "AcrossNationalBoundaries" enumeration MUST also be set.
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.

ReportType

One or many. REQUIRED. The values for the attribute "type" are meant to assist in determining if a report or report request is appropriate for the entity receiving the request or report. 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 provider (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 ReportSchema.
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 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 five 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.

MsgType

REQUIRED. ENUM. The type of GRC-Exchange message sent. The five types of messages are described in Section 5 and can be noted as one of the five selections below.

2. RequestAuthorization. This message is sent to the initiating GRC-Exchange entity if a ReportRequest or ReportQuery has been denied or is pending.
3. Result. This message provides the result of a ReportQuery.
4. ReportRequest. The purpose of the ReportRequest is to request a report from an entity.
5. Report. This message is used to provide a GRC XML report.
6. ReportQuery. 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.

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

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.

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 GRC-Exchange system 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.

MsgType-ext

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-ext

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.

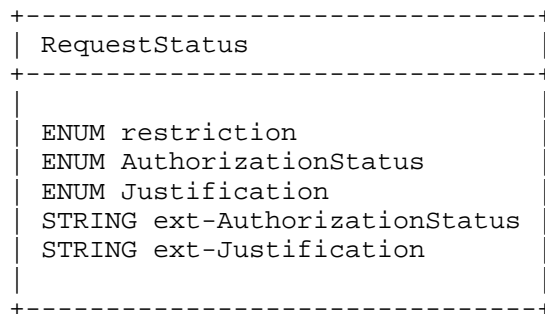


Figure 4: The RequestStatus Class

The RequestStatus class has five 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.

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 RequestAuthorization 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 RequestAuthorization 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.

Justification

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

2. SystemResource. A resource issue exists on the systems that would be involved in the request.
3. Authentication. The enveloped digital signature [RFC3275] failed to validate.
4. AuthenticationOrigin. The detached digital signature for the original requestor on the RecordItem entry failed to validate.
5. Encryption. Unable to decrypt the request.
6. Other. There were other reasons this request could not be processed.
7. 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.

AuthorizationStatus-ext

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-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. ReportSchema

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

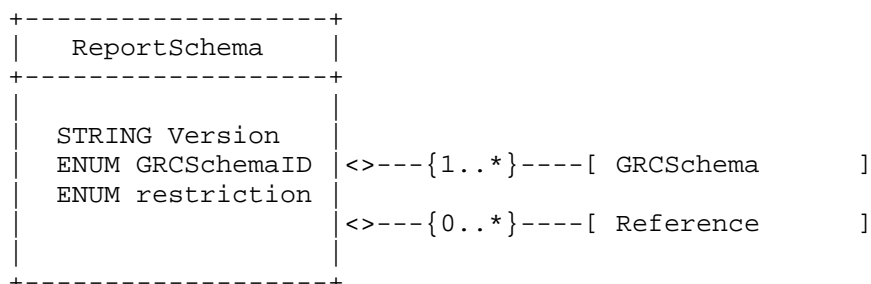


Figure 5: The ReportSchema Class

The elements that constitute the ReportShema class are as follows:

GRCSchema

One or more. EM_XML. The GRCSchema is a complete XML document formatted according to the specification identified in the attributes of this class.

Reference

OPTIONAL. One. A reference to the XML schema types included for the exchange. The Reference class is derived from IODEF [RFC5070] and fully included within this specification to avoid the need for including the IODEF schema.

The ReportSchema class has three attributes:

Version

REQUIRED. One. The Version attribute is the version number of the specified XMLSchema.

GRCSchemaID

REQUIRED. One. The GRCSchemaID attribute is the identifier (ID) of the XMLSchema used to exchange data. The GRCSchemaID and Version specify the format of the GRCSchema element.

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.

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.

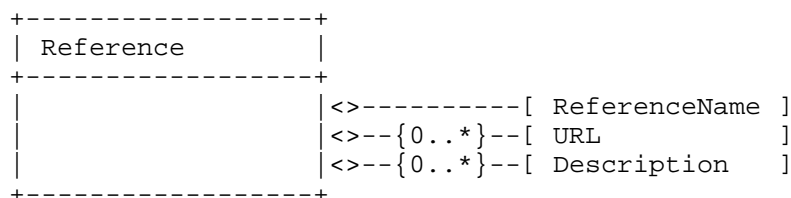


Figure 6: The Reference Class

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. GRC-Exchange Name Spaces

The GRC-Exchange schema declares a namespace of "grc-exchange-1.0" and registers it per [XMLnames]. Each XXXXX document MUST use the "grc-exchange-1.0" namespace in the top-level element GRC-Exchange-Document. It can be referenced as follows:

```
<GRC-Exchange-Document version="1.00" lang="en-US"
  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. RequestAuthorization

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

The following information is required for RequestAuthorization 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], [XMLsig]:

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 ReportQuery. The ReportQuery 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, ReportRequest MUST be used. The details of the ReportQuery 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

ReportSchema

The ReportSchema class specifies the specific GRC-Exchange XML schema that is required per the ReportQuery. 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 ReportQuery.

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 ReportQuery. 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. ReportRequest

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

The following information is required for ReportRequest 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 ReportRequest.

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 [XMLencrypt] 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 RequestAuthorization 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 ReportRequest 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 ReportRequest.

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 [XMLencrypt] 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. ReportQuery

Description: The ReportQuery 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 ReportQuery 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 ReportQuery.

Standards for encryption and digital signatures [RFC3275]:

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

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

The proper response to the ReportQuery message is a Result message. Security requirements include the ability to encrypt [XMLencrypt] the contents of the ReportRequest message using the public key of the destination entity communicating via GRC-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 RequestAuthorization 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.

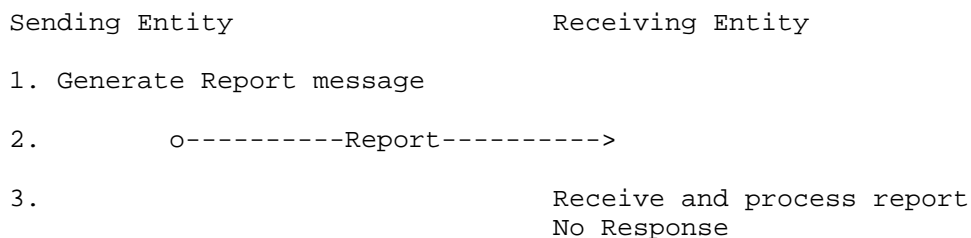


Figure 7: GRC-Exchange Report Communication Flow

The Report message MAY be encrypted [XMLencrypt] 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 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. .

7.1.1. GRC-Exchange Report Example

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

In the following example, use of [XMLsig] to generate digital signatures does not currently provide digest algorithm agility, as [XMLsig] only supports SHA-1. A future version of [XMLsig] may support additional digest algorithms to support digest algorithm agility.

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

7.2. Report Request Communication Flow

The diagram below outlines the GRC-Exchange report request communication flow between participating entities. The proper response to a ReportRequest 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 RequestAuthorization message is sent to the requestor. The RequestAuthorization message should provide the reason why the message could not be processed.

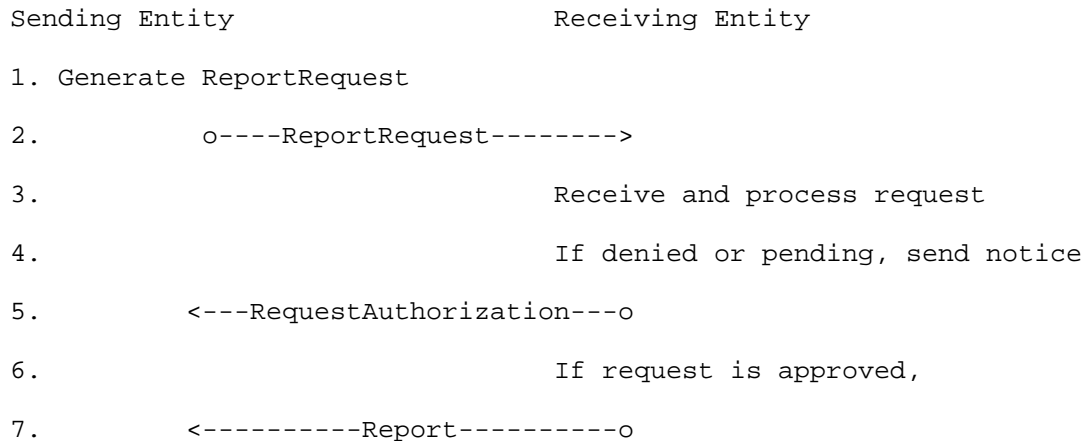


Figure 8: ReportRequest Communication Flow

7.2.1. ReportRequest Example

The following example of the ReportRequest is based on the ReportID time-based identifier tied to the specified GRC XML ReportSchema.

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

7.2.2. RequestAuthorization Message Example

The example RequestAuthorization message is in response to the ReportRequest 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.

An example Report has been provided in the previous section. No Report message would be provided in this example as a result of the denied request.

7.3. ReportQuery Communication Flow

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

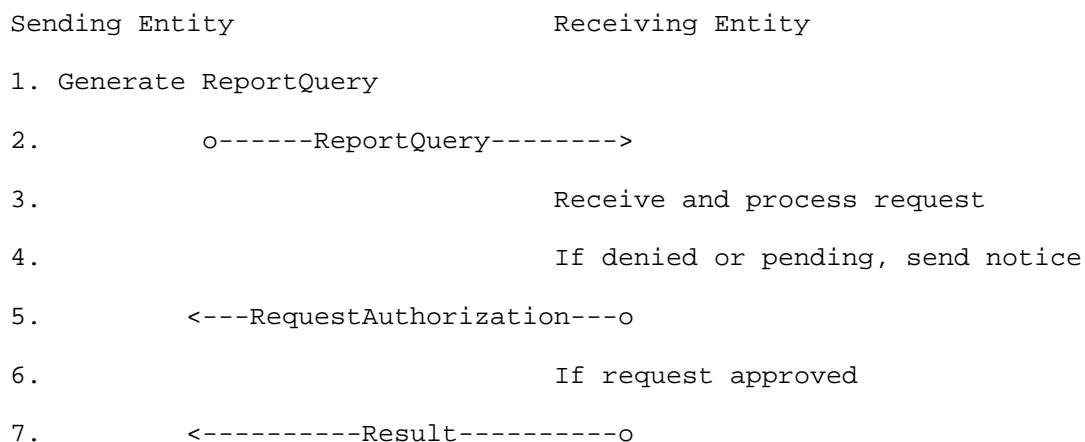


Figure 9: ReportQuery Communication Flow

The ReportQuery 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 ReportQuery message, such as a failure to validate the digital signature [RFC3275] or decrypt the request, a RequestAuthorization message is sent to the requestor. The RequestAuthorization message should provide the reason why the message could not be processed.

7.3.1. ReportQuery Example

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

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

7.3.2. RequestAuthorization Message Example

The example RequestAuthorization message is in response to the ReportQuery message listed above. The entity that received the request is responding with an answer to the ReportQuery. The Result in this instance will be delayed for more than the 5-minute default time period, hence a RequestAuthorization 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 ReportQuery request. This message type may be preceded by a RequestAuthorization within the ReportQuery flow of messages. It may be a direct response to an ReportQuery request if the request is approved prior to the timeout period. This message provides a response to the request in the ReportQuery.

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

8. 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 ***

```

9. Requirements for GRC XML Schemas for GRC-Exchange

GRC Report Exchange is a generalized version of the Real-time Inter-network Defense (RID) [RFC6045-bis] 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 enumerations to facilitate the automated security and confidentiality of 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 [XMLencrypt] 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.

10. Security Requirements

10.1. 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 [XMLencrypt] and digital signatures [RFC3275], [XMLsig]. 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 [XMLencrypt] are used in the following cases:

XML Digital Signature

- o The originator of the ReportRequest or ReportQuery MUST use an enveloped signature to sign the included GRC XML document. The enveloped digital signature provides authentication to all participants receiving the original Report or a forwarded Report. If the Report details could change, the XML digital signature specification [XMLsig] MUST be followed to specify what elements may be changed within the signed document to allow the signature to be validated. If the GRC XML document is changed within the allowed parameters, the entity including the changes MUST digitally sign [XMLsig] the updated document, while including the original detached signature. This signature MUST be passed to all recipients of the Report or Result. If this option is used, the implementation MUST follow the guidance specified in the XML Digital Signature [XMLsig] specification for two or more enveloped signatures.
- o For all message types, the full GRC XML and GRC-Exchange document MUST be signed using an enveloped signature by the sending peer to provide authentication and integrity to the receiving entity.

XML Encryption

- o The GRC XML and GRC-Exchange document may be encrypted to provide an extra layer of security between peers so that the message is not only encrypted for the transport, but also while stored. 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 [RFC6046-bis].
- o A Report, Result, or any other message type that may be relayed through multiple entities may be in part of whole encrypted for a set of intended recipients. This may be necessary if some parties receiving the XML document require full access based on need-to-know requirements and some only require access to a portion of the XML document. In cases such as this, the GRC-Exchange information is maintained in clear text while the appropriate portions of the XML document are encrypted.
- o The restriction attribute sets expectations for the privacy of an XML document and is defined in section 3.2 of RFC5070. 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 a GRC XML document, within specific classes of the GRC XML schema 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 GRC XML document or the complete GRC 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. The use of PKI between entities SHOULD adhere to any applicable certificate policy and practices agreements for the use of GRC-Exchange.
 - * 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 a report 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 GRC XML document.
 - * If restriction is set within individual classes of the GRC XML schema included, the restriction applies to the specific class of the GRC XML document and the children of that class.

The formation of policies is a very important aspect of using a messaging system 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 8.5 and 8.6 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 client information. GRCPolicy 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 communicate request and Report messages between peers, from a provider to client, or to file reports

to a centralized repository as required. GRC-Exchange provides the ability for participating entities to manage their own security controls, leveraging the information listed in GRCPolicy, mapped to their policies and agreements.

10.2. Message Transport

The transport specifications are fully defined in a separate document, leveraging the transport for RID in [RFC6046-bis]. The specified transport protocols MUST use encryption to provide an additional level of security and integrity, while supporting mutual authentication through bi-directional certificate usage. Any subsequent transport method defined should take advantage of existing standards for ease of implementation and integration of RID systems. Session encryption for the transport of GRC Report Exchange messages is enforced in the transport specification. The privacy and security considerations are addressed fully in GRC Report Exchange to protect sensitive portions of documents and provide a method to authenticate the messages. Therefore, GRC Report Exchange messages do not rely on the security provided by the transport layer alone. The encryption requirements and considerations for RID are discussed in Section 9.1 of this document.

XML security functions such as the digital signature [RFC3275] and encryption [XMLencrypt] provide a standards-based method to encrypt and digitally sign messages. GRC-Exchange messages specify system use and privacy guidelines through the GRCPolicy class. A public key infrastructure (PKI) provides the base for authentication and authorization, encryption, and digital signatures to establish trust relationships between members of a consortium or a peering consortium.

XML security functions such as the digital signature [RFC3275] and encryption [XMLencrypt] can be used within the contents of the message for privacy and security in cases for which certain elements must remain encrypted or signed as they are sent or forwarded to multiple recipients. For example, the digital signature on a Report can be used to verify the identity of originator of the Report.

10.3. Message Delivery Protocol - Integrity and Authentication

The GRC-Exchange protocol must be able to guarantee delivery and meet the necessary security requirements of a state-of-the-art protocol. In order to guarantee delivery, TCP should be considered as the underlying protocol within the current network standard practices.

Security requirements must include the integrity, authentication, privacy, and authorization of the messages sent between systems

communicating via GRC-Exchange. The communication between GRC-Exchange systems must be authenticated and encrypted to ensure the integrity of the messages and the systems involved in the communications. Another concern that needs to be addressed is authentication for a request that traverses multiple networks. In this scenario, systems in the path of the multi-hop ReportRequest need to authorize a request from not only the direct peer but also from the initiating entity as discussed in Section 9.5. Several methods can be used to ensure integrity and privacy of the communication.

The transport mechanism selected MUST follow the defined transport protocol [RFC6046] when using GRC-Exchange messaging to ensure consistency among the peers. Consortia may vary their selected transport mechanisms and thus must decide upon a mutual protocol to use for transport when communicating with peers in a neighboring consortium using GRC-Exchange. GRC-Exchange systems MUST implement and deploy HTTPS as defined in the transport document [RFC6046] and optionally support other protocols such as the Blocks Extensible Exchange Protocol (BEEP). GRC-Exchange, the XML security functions, and transport protocols must properly integrate with a public key infrastructure (PKI) managed by the consortium or one managed by a trusted entity. For the Internet, a few of examples of existing efforts that could be leveraged to provide the supporting PKI include the American Registry for Internet Numbers (ARIN) and 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 consortia are discussed in Sections 8.5 and 8.6.

10.4. Transport Communication

In order to address the integrity and authenticity of messages, transport encryption MUST be used to secure the traffic sent between entities exchanging GRC requests and reports. Systems with predefined relationships for GRC-Exchange include those who peer within a consortium with agreed-upon appropriate use regulations and for peering consortia. Trust relationships may also be defined through a bridged or hierarchical PKI in which both peers belong.

Systems used to send authenticated GRC-Exchange messages between networks MUST use a secured system and interface to connect to a border network's RID systems. Each connection to a GRC-Exchange system MUST meet the security requirements agreed upon through the consortium regulations, peering, or SLAs. The GRC-Exchange system MUST only listen for and send GRC-Exchange messages on the designated port, which also MUST be over an encrypted tunnel meeting the minimum requirement of algorithms and key lengths established by the

consortium, peering, or SLA. The selected cryptographic algorithms for symmetric encryption, digital signatures, and hash functions MUST meet minimum security levels of the times. The encryption strength MUST adhere to import and export regulations of the involved countries for data exchange.

10.5. Authentication of The GRC Report Exchange Protocol

In order to ensure the authenticity of the GRC-Exchange messages, a message authentication scheme is used to secure the protocol. 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 network providers. The PKI used for authentication also provides the necessary certificates needed for encryption used for the RID transport protocol [RFC6046].

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

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. 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 [XMLencrypt] or digital signature [XMLsig], [RFC3275] is used within a document. Transport specifications are detailed in a separate document [RFC6046].

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.

10.5.1. Multi-Hop Authentication

The use of multi-hop authentication for a Report message may be used when a Report is sent to multiple entities in an iterative manner. For practical reasons, the entities may want to prioritize report requests based upon the immediate peer making the request, the originator of a request (if different from the peer sending the request), and other information contained in the GRC XML schema.

A second measure MUST be taken to ensure the identity of the originator of the request or report. The originating entity MUST include an enveloped digital signature of the GRC XML document following the XML digital signature specifications from W3C [XMLsig]. The signature MUST be passed to all parties that receive the forwarded message, and each party MUST be able to perform full path validation on the digital signature [RFC5280]. Full path validation verifies the chaining relationship to a trusted root and also performs a certificate revocation check. In order to accommodate that requirement, the XML Digital Signature [XMLsig] specification must be followed for two or more enveloped signatures. If additions are made to a Report, the specification for two or more digital signatures [XMLsig] MUST be followed to authenticate the origin of the XML document components.

In the case in which an enterprise network using GRC-Exchange sends a report or request to its service provider (SP), the signature from the enterprise MUST be included in the initial request. The SP may generate forward the request if appropriate. If the original request is sent, the originating SP, acting on behalf of the enterprise network, MUST also digitally sign, with an enveloped signature, the full GRC XML document to assure the authenticity of the request. An SP that offers GRC-Exchange to provide reports as a service may be using its own PKI to secure communications between the SP and its tenants or clients.

10.6. Consortiums and Public Key Infrastructures

Direct relationships may be ideal for most GRC-Exchange communications, such as those between a service provider and its tenants for which reports will be issued. Consortiums can be used to establish a communication web of trust for GRC-Exchange messaging where appropriate. The consortium could provide centralized resources, such as a PKI, and established guidelines for use of the GRC-Exchange protocol. The consortium may assist in establishing trust relationships between the participating entities 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 entities in a common area (Internet, region, government, education, private networks, etc.).

Using a PKI to distribute certificates used in GRC-Exchange communication provides an already established method to link trust relationships between entities or consortiums that peer with entities belonging to a separate consortium. In other words, consortiums could peer with other consortiums to enable communication of GRC-Exchange messages between the participating entities to extend trust relationships. 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.

Consortiums also need to establish guidelines for each participating entities to adhere. The RECOMMENDED guidelines include:

- o Physical and logical practices to protect of systems;
- o Network and application layer protection for systems and communications;
- o Proper use guidelines for GRC-Exchange messages and requests; and
- o A PKI and policy 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 SP. The PKI may be a subordinate CA or in the CA hierarchy from the SP or consortium to establish the trust relationships necessary as the request is made to other connected networks.

10.7. Privacy Concerns and System Use Guidelines

The GRCPolicy class is used to automate the enforcement of the privacy concerns listed within this document. The privacy and system use concerns that MUST be addressed in the GRC-Exchange communication system and other integrated components include the following:

Privacy Concerns:

- o Privacy of data collected for continuous monitoring for IT Operations, Security, Compliance, and Audit reports. This data may be of the entity being monitored or tenant data in a service provider model. Agreements MUST be established for the proper handling of this data to the data owners requirements.
- o Privacy of data monitored and stored on systems used in legal exchanges such as eDiscovery or eWarrant reports.

Information Sharing Considerations:

- o System use between peering consortiums MUST also adhere to any government communication regulations that apply between those two regions, such as encryption export and import restrictions. This may include consortiums that are categorized as "BetweenConsortiums" or "AcrossNationalBoundaries".
- o System use between consortiums MUST NOT request reports beyond the scope intended and permitted by law or established agreements.
- o GRC Report Exchange communications between entities classified as "AcrossNationalBoundaries" MUST respect national boundary issues and limit requests to appropriate agreements including those which involve peering consortia.

The security and privacy considerations listed above are for the consortiums, SPs, and enterprises to agree upon. The agreed-upon policies may be facilitated through use of the GRCPolicy class. Some privacy considerations are addressed through the GRC-Exchange guidelines for encryption and digital signatures as described in Section 9.1.

Privacy becomes an issue whenever sensitive data traverses a network. The specific concerns for each GRC XML schema used must be considered individually. The exchange of information in provided Reports, should be detailed in agreements and contracts prior to information sharing exchanges taking place. The agreements should consider the level of detail to be provided in a report, what information MUST not be communicated, and how information is protected in exchanges.

Intra-consortium GRC-Exchange communications raise additional issues, especially when the peering consortiums reside in different regions or nations. Data privacy regulations and other applicable regulations MUST be considered in information sharing or exchange policies.

The privacy concerns listed in this section address issues among the trusted parties involved in a GRC Report exchanges. Data used in 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. Each system communicating GRC-Exchange messages 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 the transport protocol [RFC6046]. 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 Report Exchange messages.

11. 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.

12. IANA Considerations

This document uses URNs to describe XML namespaces and XML schemas [XMLschema] 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 7, "GRC-Exchange Schema Definition", of this document.

13. Summary

Governance, Risk, and Compliance reports may contain some of the most sensitive information for a business. Reports may contain the 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.

14. References

14.1. 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.
- [RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, January 2004.
- [RFC4279] Eronen, P. and H. Tschofenig, "Pre-Shared Key Ciphersuites for Transport Layer Security (TLS)", RFC 4279, December 2005.
- [RFC5070] Danyliw, R., Meijer, J., and Y. Demchenko, "The Incident Object Description Exchange Format", RFC 5070, December 2007.
- [RFC5275] Turner, S., "CMS Symmetric Key Management and Distribution", RFC 5275, June 2008.
- [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.
- [RFC5755] Farrell, S., Housley, R., and S. Turner, "An Internet Attribute Certificate Profile for Authorization", RFC 5755, January 2010.
- [RFC6045] Moriarty, K., "Real-time Inter-network Defense (RID)", RFC 6045, November 2010.

- [RFC6046] Moriarty, K. and B. Trammell, "Transport of Real-time Inter-network Defense (RID) Messages", RFC 6046, November 2010.
- [XML1.0] Bray, T., Maler, E., Paoli, J., Sperberg-McQueen, C., and F. Yergeau, "Extensible Markup Language (XML) 1.0", W3C Recommendation XML 1.0, November 2008, <<http://www.w3.org/TR/xml/>>.
- [XMLNames] Bray, T., Hollander, D., Layman, A., Tobin, R., and H. Thomson, "Namespaces in XML 1.0 (Third Edition)", W3C Recommendation, December 2009, <<http://www.w3.org/TR/xml-names/>>.
- [XMLencrypt] Imaura, T., Dillaway, B., and E. Simon, "XML Encryption Syntax and Processing", W3C Recommendation, December 2002, <<http://www.w3.org/TR/xmlenc-core/>>.
- [XMLschema] Thompson, H., Beech, D., Maloney, M., and N. Mendelsohn, "XML Schema Part 1: Structures", W3C Recommendation Second Edition, October 2004, <<http://www.w3.org/TR/xmlschema-1/>>.
- [XMLsig] Bartel, M., Boyer, J., Fox, B., LaMaccia, B., and E. Simon, "XML-Signature Syntax and Processing", W3C Recommendation Second Edition, June 2008, <<http://www.w3.org/TR/xmldsig-core/>>.

14.2. Informative References

- [RFC1930] Hawkinson, J. and T. Bates, "Guidelines for creation, selection, and registration of an Autonomous System (AS)", BCP 6, RFC 1930, March 1996.
- [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.
- [RFC5735] Cotton, M. and L. Vegoda, "Special Use IPv4 Addresses", BCP 153, RFC 5735, January 2010.

Acknowledgements

Many thanks to colleagues and the Internet community for reviewing

and commenting on the document.

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

Individual Submission
Internet-Draft
Intended status: Standards Track
Expires: May 1, 2012

T. Takahashi
NICT
K. Landfield
McAfee
T. Millar
USCERT
Y. Kadobayashi
NICT
Oct 29, 2011

IODEF-extension to support structured cybersecurity information
draft-takahashi-mile-sci-02.txt

Abstract

This document extends the Incident Object Description Exchange Format (IODEF) defined in RFC 5070 [RFC5070] to facilitate enriched cybersecurity information exchange among cybersecurity entities by embedding structured information formatted by specifications, including CAPEC[TM] [CAPEC], CEE[TM] [CEE], CPE[TM] [CPE], CVE(R) [CVE], CVRF [CVRF], CVSS [CVSS], CWE[TM] [CWE], CWSS[TM] [CWSS], ISO/IEC 19770-2 [ISOIEC19770-2], OCIL [OCIL], OVAL(R) [OVAL], XCCDF [XCCDF], and XDAS [XDAS].

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 1, 2012.

Copyright Notice

Copyright (c) 2011 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	3
4. Extension Definition	4
4.1. Structured Cybersecurity Information Formats	4
4.2. Extended Data Types	5
4.2.1. EM_XML	5
4.3. Extended Classes	6
4.3.1. AttackPattern	7
4.3.2. PlatformID	8
4.3.3. Vulnerability	9
4.3.4. Scoring	11
4.3.5. Weakness	12
4.3.6. EventReport	13
4.3.7. Remediation	14
5. Examples	15
5.1. Reporting an attack	16
6. Security Considerations	18
6.1. Transport-Specific Concerns	18
6.2. Using the iodef:restriction Attribute	19
7. IANA Considerations	19
8. Acknowledgment	20
9. Appendix: XML Schema Definition for Extension	20
10. References	24
10.1. Normative References	24
10.2. Informative References	24
Authors' Addresses	25

1. Introduction

Cyber attacks are getting more sophisticated, and their numbers are increasing day by day. To cope with such situation, incident information needs to be reported, exchanged, and shared among organizations. IODEF is one of the tools enabling such exchange, and is already in use.

To efficiently run cybersecurity operations, these exchanged information needs to be machine-readable. IODEF provides a structured means to describe the information, but it needs to embed various non-structured such information in order to convey detailed information. Further structure within IODEF increases IODEF documents' machine-readability and thus facilitates streamlining cybersecurity operations.

On the other hand, there exist various other activities facilitating detailed and structured description of cybersecurity information, major of which includes CAPEC [CAPEC], CEE [CEE], CPE [CPE], CVE [CVE], CVRF [CVRF], CVSS [CVSS], CWE [CWE], CWSS [CWSS], ISO/IEC 19770-2 [ISOIEC19770-2], OCIL [OCIL], OVAL [OVAL], XCCDF [XCCDF], and XDAS [XDAS]. Since such structured description facilitates cybersecurity operations, it would be beneficial to embed and convey these information inside IODEF document.

To enable that, this document extends the IODEF to embed and convey various structured cybersecurity information, with which cybersecurity operations can be facilitated. 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 cybersecurity, organization needs to exchange

cybersecurity information, which includes the following information: attack pattern, platform information, vulnerability and weakness, countermeasure instruction, computer event log, and the severity.

IODEF provides a scheme to exchange such information among interested parties. However, the detailed common format to describe such information is not defined in the IODEF base document.

On the other hand, to describe those information and to facilitate exchange, a structured format for that is already available. Major of them are CAPEC, CEE, CPE, CVE, CVRF, CVSS, CWE, CWSS, OVAL, and XCCDF. By embedding them into the IODEF document, the document can convey more detailed contents to the receivers, and the document can be easily reused.

These structured cybersecurity information facilitates cybersecurity operation at the receiver side. Since the information is machine-readable, the data can be processed by computers. That expedites the automation of cybersecurity operations

For instance, an organization wishing to report a security incident wants to describe what vulnerability was exploited. Then the sender can simply use IODEF, where an CAPEC record is embedded instead of describing everything in free format text. Receiver can also identify the needed details of the attack pattern by looking up some of the xml tags defined by CAPEC. Receiver can accumulate the attack pattern information (CAPEC record) in its database and could distribute it to the interested parties if needed, without needing human interventions.

4. Extension Definition

This draft extends IODEF to embed structured cybersecurity information by introducing new classes, with which these information can be embedded inside IODEF document as element contents of AdditionalData and RecordItem classes.

4.1. Structured Cybersecurity Information Formats

This extension intends to embed various structured cybersecurity information. The below table describes the initial list of supported specifications and their IDs, versions, and namespaces; future assignments are to be made through Expert Review, as requested in Section 7.

ID	Specification Name	Version	Namespace
CAPEC_1.6	Common Attack Pattern Enumeration and Classification (CAPEC)	1.6	http://capec.mitre.org/observables
CEE_0.6	Common Event Expression (CEE)	0.6	http://cee.mitre.org
CPE_2.3	Common Platform Enumeration (CPE)	2.3	http://cpe.mitre.org/dictionary/2.0
CVE_1.0	Common Vulnerability and Exposures (CVE)	1.0	http://cve.mitre.org/cve/downloads/1.0
CVRF_1.0	Common Vulnerability Reporting Format (CVRF)	1.0	http://www.icasi.org/CVRF/schema/cvrf/1.0
CVSS_2.0	Common Vulnerability Scoring System (CVSS)	2	http://scap.nist.gov/schema/cvss-v2/1.0
CWE_5.0	Common Weakness Enumeration (CWE)	5.1	N/A
CWSS_0.8	Common Weakness Scoring System (CWSS)	0.8	N/A
OCIL_2.0	Open Checklist Interactive Language (OCIL)	2.0	http://scap.nist.gov/schema/ocil/2.0
OVAL_5.10	Open Vulnerability and Assessment Language (OVAL)	5.10	http://oval.mitre.org/XMLSchema/oval-definitions-5
XCCDF_1.2	Extensible Configuration Checklist Description Format (XCCDF)	1.2	http://checklists.nist.gov/xccdf/1.2
XDAS_1998	Distributed Audit Service (XDAS)	1998	N/A
19770-2	ISO/IEC 19770	Part 2	N/A

Figure 1: List of specifications

4.2. Extended Data Types

This extension inherits all of the data types defined in the IODEF model. One data type is added: EM_XML.

4.2.1. EM_XML

An embedded complete XML document is represented by the EM_XML data type. The elements of the document must match its root namespace element.

4.3. Extended Classes

The IODEF Incident element [RFC5070] is summarized below. It is expressed in Unified Modeling Language (UML) syntax as used in the IODEF specification. The UML representation is for illustrative purposes only; elements are specified in XML as defined in Appendix A.

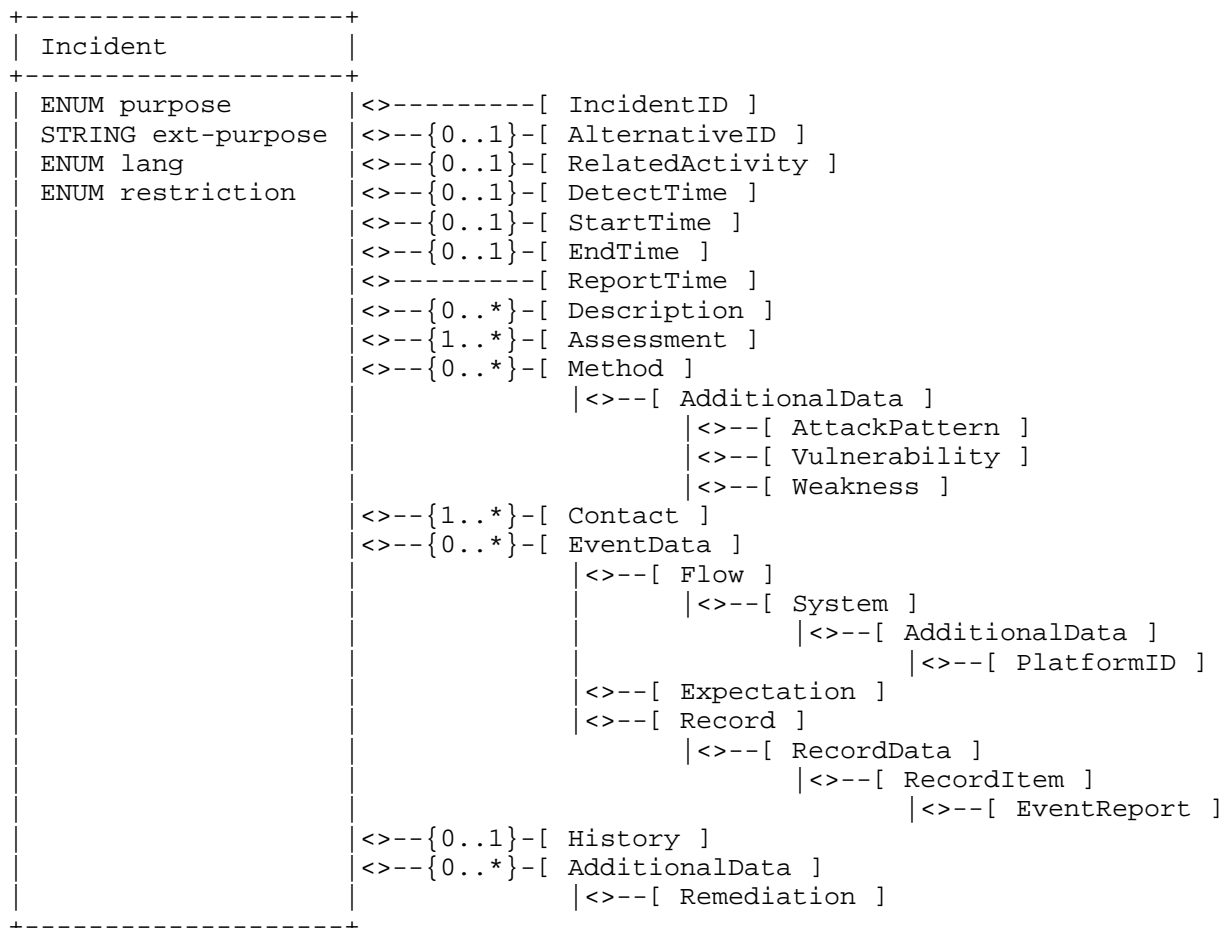


Figure 2: Incident class

This extension defines the following seven elements.

4.3.1. AttackPattern

An AttackPattern consists of an extension to the Incident.Method.AdditionalData element with a dtype of "xml". The extension describes attack patterns of incidents or events.

It is recommended that Method class SHOULD contain one or more of the extension elements whenever available.

An AttackPattern class is structured as follows.

```
+-----+
| AttackPattern          |
+-----+
| STRING Version         | <>--(0..*)-[ Record ]
| ENUM SpecificationID   | <>--(0..*)-[ Reference ]
| STRING AttackPatternID | <>--(0..*)-[ PlatformID ]
+-----+
```

Figure 3: AttackPattern class

This class has the following attributes.

Version: OPTIONAL. STRING. The version number of the extension specification to which this class conforms. This value should be 1.00, to be compliant with this document. Its default value is 1.00.

SpecificationID: REQUIRED. ENUM. The ID of the specification and its version specifying the format of the Record element. The value should be chosen from the IDs listed in Figure 1, such as CAPEC_1.6. Note that the lists in Figure 1 will be developed further by IANA.

AttackPatternID: OPTIONAL. STRING. An ID of an attack pattern to be reported. This attribute SHOULD be used whenever such ID is available. In case a Record or Reference element is provided along with this attribute, writers/senders MUST ensure that this ID is consistent with the one provided by the element; if a reader/receiver detects an inconsistency, it SHOULD prefer the value of this attribute, and SHOULD log the inconsistency so a human can correct the problem. Note that this attribute could be omitted if no such ID is available. In this case, either Record or Reference elements, or both of them, MUST be provided.

The AttackPattern class is composed of the following aggregate classes.

Record: Zero or more. EM_XML. A complete document that is formatted according to the specification and its version identified by the value of the SpecificationID with the Figure 1.

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

PlatformID: Zero or more. An identifier of software platform involved in the specific attack pattern, which is elaborated in Section 4.3.2. Some of the structured information embedded in the Record element may include the identifier within it. In this case, this PlatformID element SHOULD NOT be used. If a reader/receiver detects the identifiers in both Record and PlatformID elements and their inconsistency, it SHOULD prefer the identifiers derived from the PlatformID element, and SHOULD log the inconsistency so a human can correct the problem.

Writers/senders MUST ensure the specification name and version identified by the SpecificationID are consistent with the contents of the Record; if a reader/receiver detects an inconsistency, it SHOULD prefer the specification name and version derived from the content, and SHOULD log the inconsistency so a human can correct the problem.

4.3.2. PlatformID

A PlatformID identifies a software platform. It is recommended that AttackPattern, Vulnerability, Weakness, and System classes contain this elements whenever available.

A PlatformID element is structured as follows.

```
+-----+
| PlatformID |
+-----+
| STRING Version | <!--(1..*)-[ ID ]
| ENUM SpecificationID |
+-----+
```

Figure 4: PlatformID class

This class has the following attributes.

Version: OPTIONAL. STRING. The version number of the extension specification to which this class conforms. This value should be 1.00, to be compliant with this document. Its default value is 1.00.

SpecificationID: REQUIRED. ENUM. The ID of the specification and its version specifying the format of the ID element. The value should be chosen from the IDs listed in Figure 1, such as CPE_2.3 and ISO/IEC 19770-2. Note that the lists in Figure 1 will be developed further by IANA.

This class is composed of the following aggregate classes.

ID: One or more. ML_STRING. An ID that is formatted according to the rule defined by the specification and its version identified by the value of the SpecificationID with the Figure 1.

Writers/senders MUST ensure the specification name and version identified by the SpecificationID are consistent with the contents of the ID; if a reader/receiver detects an inconsistency, it SHOULD prefer the specification name and version derived from the content, and SHOULD log the inconsistency so a human can correct the problem.

4.3.3. Vulnerability

A Vulnerability consists of an extension to the Incident.Method.AdditionalData element with a dtype of "xml". The extension describes the (candidate) vulnerabilities of incidents or events.

It is recommended that Method class SHOULD contain one or more of the extension elements whenever available.

A Vulnerability element is structured as follows.

```
+-----+
| Vulnerability |
+-----+
| STRING Version | <!--(0..*)-[ Record ]
| ENUM SpecificationID | <!--(0..*)-[ Reference ]
| STRING VulnerabilityID | <!--(0..*)-[ PlatformID ]
| | <!--(0..*)-[ Scoring ]
+-----+
```

Figure 5: Vulnerability class

This class has the following attributes.

Version: OPTIONAL. STRING. The version number of the extension specification to which this class conforms. This value should be 1.00, to be compliant with this document. Its default value is 1.00.

SpecificationID: REQUIRED. ENUM. The ID of the specification and its version specifying the format of the Record element. The value should be chosen from the IDs listed in Figure 1, such as CVE_1.0 and CVRF_1.0. Note that the lists in Figure 1 will be developed further by IANA.

VulnerabilityID: OPTIONAL. STRING. An ID of a vulnerability to be reported. This attribute SHOULD be used whenever such ID is available. In case a Record or Reference element is provided along with this attribute, writers/senders MUST ensure that this ID is consistent with the one provided by the element; if a reader/receiver detects an inconsistency, it SHOULD prefer the value of this attribute, and SHOULD log the inconsistency so a human can correct the problem. Note that this attribute could be omitted if no such ID is available. In this case, either Record or Reference elements, or both of them, MUST be provided.

This class is composed of the following aggregate classes.

Record: Zero or one. EM_XML. A complete document that is formatted according to the specification and its version identified by the value of the SpecificationID with the Figure 1.

Reference: Zero or one of iodef:Reference [RFC5070]. This element allows an IODEF document to include a link to a structured information instead of directly embedding it into a Record element.

PlatformID: Zero or more. An identifier of software platform affected by the vulnerability, which is elaborated in Section 4.3.2. Some of the structured information embedded in the Record element may include the identifier within it. In this case, this PlatformID element SHOULD NOT be used. If a reader/receiver detects the identifiers in both Record and PlatformID elements and their inconsistency, it SHOULD prefer the identifiers derived from the PlatformID element, and SHOULD log the inconsistency so a human can correct the problem.

Scoring: Zero or more. An indicator of the severity of the vulnerability, such as CVSS score, which is elaborated in Section 4.3.4. Some of the structured information may include scores within it. In this case, the Scoring element SHOULD NOT be used since the Record element contains the scores. If a reader/

receiver detects scores in both Record and Scoring elements and their inconsistency, it SHOULD prefer the scores derived from the Record element, and SHOULD log the inconsistency so a human can correct the problem.

4.3.4. Scoring

A Scoring class describes the scores of the severity in terms of security. It is recommended that Vulnerability and Weakness classes contain the elements whenever available.

A Scoring class is structured as follows.

```
+-----+
| Scoring |
+-----+
| STRING Version | <>-----[ Score ]
| ENUM SpecificationID |
+-----+
```

Figure 6: Scoring class

This class has two attributes.

Version: OPTIONAL. STRING. The version number of the extension specification to which this class conforms. This value should be 1.00, to be compliant with this document. Its default value is 1.00.

SpecificationID: REQUIRED. STRING. The ID of the specification and its version specifying the format of the Score element. The value should be chosen from the IDs listed in Figure 1, such as CVSS_2.0 and CWSS_0.8. Note that the lists in Figure 1 will be developed further by IANA.

This class is composed of an aggregate class.

Score: One. EM_XML. Arbitrary information structured by the specification identified by the specification and its version identified by the value of the SpecificationID with the Figure 1.

Writers/senders MUST ensure the specification name and version identified by the SpecificationID are consistent with the contents of the Score; if a reader/receiver detects an inconsistency, it SHOULD prefer the specification name and version derived from the content, and SHOULD log the inconsistency so a human can correct the problem.

4.3.5. Weakness

A Weakness consists of an extension to the Incident.Method.AdditionalData element with a dtype of "xml". The extension describes the weakness types of incidents or events.

It is recommended that Method class SHOULD contain one or more of the extension elements whenever available.

A Weakness element is structured as follows.

```
+-----+
| Weakness |
+-----+
| STRING Version | <>--(0..*)-[ Record ]
| ENUM SpecificationID | <>--(0..*)-[ Reference ]
| STRING WeaknessID | <>--(0..*)-[ PlatformID ]
|               | <>--(0..*)-[ Scoring ]
+-----+
```

Figure 7: Weakness class

This class has the following attributes.

Version: OPTIONAL. STRING. The version number of the extension specification to which this class conforms. This value should be 1.00, to be compliant with this document. Its default value is 1.00.

SpecificationID: REQUIRED. ENUM. The ID of the specification and its version specifying the format of the Record element. The value should be chosen from the IDs listed in Figure 1, such as CWE_5.0. Note that the lists in Figure 1 will be developed further by IANA.

WeaknessID: OPTIONAL. STRING. An ID of a weakness to be reported. This attribute SHOULD be used whenever such ID is available. In case a Record or Reference elements is provided along with this attribute, writers/senders MUST ensure that this ID is consistent with the one provided by the element; if a reader/receiver detects an inconsistency, it SHOULD prefer the value of this attribute, and SHOULD log the inconsistency so a human can correct the problem. Note that this attribute could be omitted if no such ID is available. In this case, either Record or Reference elements, or both of them, MUST be provided.

This class is composed of the following aggregate classes.

Record: Zero or more. EM_XML. A complete document that is formatted according to the specification and its version identified by the value of the SpecificationID with the Figure 1.

Reference: Zero or one of iodef:Reference [RFC5070]. This element allows an IODEF document to include a link to a structured information instead of directly embedding it into a Record element.

PlatformID: Zero or more. An identifier of software platform affected by the weakness, which is elaborated in Section 4.3.2. Some of the structured information embedded in the Record element may include the identifier within it. In this case, this PlatformID element SHOULD NOT be used. If a reader/receiver detects the identifiers in both Record and PlatformID elements and their inconsistency, it SHOULD prefer the identifiers derived from the PlatformID element, and SHOULD log the inconsistency so a human can correct the problem.

Scoring: Zero or more. An indicator of the severity of the weakness, such as CWSS score, which is elaborated in Section 4.3.4. Some of the structured information may include scores within it. In this case, the Scoring element SHOULD NOT be used since the Record element contains the scores. If a reader/receiver detects scores in both Record and Scoring elements and their inconsistency, it SHOULD prefer the scores derived from the Record element, and SHOULD log the inconsistency so a human can correct the problem.

4.3.6. EventReport

An EventReport consists of an extension 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 SHOULD contain one or more of the extension elements whenever available.

An EventReport element is structured as follows.

```
+-----+
| EventReport |
+-----+
| STRING Version | <--(0..*)-[ Record ]
| ENUM SpecificationID | <--(0..*)-[ Reference ]
+-----+
```

Figure 8: EventReport class

This class has the following attributes.

Version: OPTIONAL. STRING. The version number of the extension specification to which this class conforms. This value should be 1.00, to be compliant with this document. Its default value is 1.00.

SpecificationID: REQUIRED. ENUM. The ID of the specification and its version specifying the format of the Record element. The value should be chosen from the IDs listed in Figure 1, such as CEE_0.6 and XDAS_1998. Note that the lists in Figure 1 will be developed further by IANA.

This class is composed of three aggregate classes.

Record: Zero or one. EM_XML. A complete document that is formatted according to the specification and its version identified by the value of the SpecificationID with the Figure 1.

Reference: Zero or one of iodef:Reference [RFC5070]. This element allows an IODEF document to include a link to a structured information instead of directly embedding it into a Record element.

This class MUST contain at least one of Record or Reference elements. Writers/senders MUST ensure the specification name and version identified by the SpecificationID are consistent with the contents of the Record; if a reader/receiver detects an inconsistency, it SHOULD prefer the specification name and version derived from the content, and SHOULD log the inconsistency so a human can correct the problem.

4.3.7. Remediation

A Remediation consists of an extension to the Incident.AdditionalData element with a dtype of "xml". The extension elements describes incident remediation information including instructions. Note that the term remediation includes a range of concepts, e.g., valudation.

It is recommended that Incident class SHOULD contain one or more of this extension elements whenever available.

A Remediation class is structured as follows.

```

+-----+
| Remediation |
+-----+
| STRING Version | <--(0..*)-[ Record ]
| ENUM SpecificationID | <--(0..*)-[ Reference ]
+-----+

```

Figure 9: Remediation class

This class has an attribute.

Version: OPTIONAL. STRING. The version number of the extension specification to which this class conforms. This value should be 1.00, to be compliant with this document. Its default value is 1.00.

SpecificationID: REQUIRED. ENUM. The ID of the specification and its version specifying the format of the Record element. The value should be chosen from the IDs listed in Figure 1, such as OVAL_5.10, OCIL_2.0, and XCCDF_1.2. Note that the lists in Figure 1 will be developed further by IANA.

This class is composed of three aggregate classes.

Record: Zero or one. EM_XML. A complete document that is formatted according to the specification and its version identified by the value of the SpecificationID with the Figure 1.

Reference: Zero or one of iodef:Reference [RFC5070]. This element allows an IODEF document to include a link to a structured information instead of directly embedding it into a Record element.

This class MUST contain at least either of Record and Reference elements. Writers/senders MUST ensure the specification name and version identified by the SpecificationID are consistent with the contents of the Record; if a reader/receiver detects an inconsistency, it SHOULD prefer the specification name and version derived from the content, and SHOULD log the inconsistency so a human can correct the problem.

5. Examples

This section provides examples of an incident encoded in the IODEF. These examples do not necessarily represent the only way to encode a particular incident.

5.1. Reporting an attack

An example of a CSIRT reporting an attack.

```
<?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="csirt.example.com">189493</IncidentID>
    <ReportTime>2001-09-13T23:19:24+00:00</ReportTime>
    <Description>Incident report in company xx</Description>
    <!-- An administrative privilege was attempted, but failed -->
    <Assessment>
      <Impact completion="failed" type="admin"/>
    </Assessment>
    <Method>
      <Description>Structured information on attack pattern, exploited
        vulnerability, and weakness</Description>
      <AdditionalData dtype="xml">
        <iodef-sci:AttackPattern SpecificationID="CAPEC_1.6"
          AttackPatternID="CAPEC-14">
          <iodef-sci:Record>[CAPEC-formatted data]</iodef-sci:Record>
          <Reference>
            <ReferenceName>Link to Capec-14</ReferenceName>
            <URL>http://capec.mitre.org/data/definitions/14.html</URL>
          </Reference>
        </iodef-sci:AttackPattern>
        <iodef-sci:Vulnerability SpecificationID="CVE_1.0"
          VulnerabilityID="CVE-2010-3654">
          <iodef-sci:Record>[CVE-formatted data]</iodef-sci:Record>
          <iodef-sci:PlatformID SpecificationID="CPE_2.3">
            <iodef-sci:ID>[CPE ID]</iodef-sci:ID>
          </iodef-sci:PlatformID>
          <iodef-sci:Scoring SpecificationID="CVSS_2.0">
            <iodef-sci:Score>[CVSS scores]</iodef-sci:Score>
          </iodef-sci:Scoring>
        </iodef-sci:Vulnerability>
        <iodef-sci:Weakness SpecificationID="CWE_5.0"
          WeaknessID="CWE-119">
          <iodef-sci:Record>[CWE-formatted data]</iodef-sci:Record>
          <iodef-sci:Scoring SpecificationID="CWSS_0.8">
            <iodef-sci:Score>[CWSS scores]</iodef-sci:Score>
          </iodef-sci:Scoring>
        </iodef-sci:Weakness>
      </AdditionalData>
```

```
</Method>
<Contact role="creator" type="organization">
  <ContactName>Example.com CSIRT</ContactName>
  <RegistryHandle registry="arin">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="6">
        <Port>80</Port>
      </Service>
      <AdditionalData dtype="xml">
        <iodef-sci:PlatformID SpecificationID="CPE_2.3">
          <iodef-sci:ID>[CPE ID]</iodef-sci:ID>
        </iodef-sci:PlatformID>
      </AdditionalData>
    </System>
  </Flow>
  <Expectation action="block-host" />
  <Expectation action="other"/>
  <!-- <RecordItem> has an excerpt from a log -->
  <Record>
    <RecordData>
      <DateTime>2001-09-13T18:11:21+02:00</DateTime>
      <Description>a Web-server event record</Description>
      <RecordItem dtype="xml">
        <iodef-sci:EventReport SpecificationID="CEE_0.6">
          <iodef-sci:Record>[CEE-formatted data]</iodef-sci:Record>
        </iodef-sci:EventReport>
      </RecordItem>
    </RecordData>
  </Record>
</EventData>
<History>
  <!-- Contact was previously made with the source network owner -->
  <HistoryItem action="contact-source-site">
    <DateTime>2001-09-14T08:19:01+00:00</DateTime>
    <Description>Notification sent to
```

```
        constituency-contact@192.0.2.200</Description>
    </HistoryItem>
</History>
<AdditionalData dtype="xml">
    <iodef-sci:Remediation SpecificationID="OVAL_5.10">
        <iodef-sci:Record>[OVAL-formatted data]</iodef-sci:Record>
    </iodef-sci:Remediation>
    <iodef-sci:Remediation SpecificationID="XCCDF_1.2">
        <iodef-sci:Record>[XCCDF-formatted data]</iodef-sci:Record>
    </iodef-sci:Remediation>
</AdditionalData>
</Incident>
</IODEF-Document>
```

Figure 10: Example UML Element Diagram

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.

Organizations that exchange data using this document are URGED to develop operating procedures that document the following areas of concern.

6.1. Transport-Specific Concerns

The underlying messaging format and protocol used to exchange instances of the IODEF MUST provide appropriate guarantees of confidentiality, integrity, and authenticity. The use of a standardized security protocol is encouraged. The Real-time Inter-network Defense (RID) protocol [RFC6045] and its associated transport binding [RFC6046] provide such security.

The critical security concerns are that these structured information may be falsified or they may become corrupt during transit. In areas where transmission security or secrecy is questionable, the application of a digital signature and/or message encryption on each report will counteract both of these concerns. We expect that each exchanging organization will determine the need, and mechanism, for transport protection.

6.2. Using the iodef:restriction Attribute

In some instances, data values in particular elements may contain data deemed sensitive by the reporter. Although there are no general-purpose rules on when to mark certain values as "private" or "need-to-know" via the iodef:restriction attribute, the reporter is cautioned not to apply element-level sensitivity markings unless they believe the receiving party (i.e., the party they are exchanging the event report data with) has a mechanism to adequately safeguard and process the data as marked.

7. IANA Considerations

This document uses URNs to describe XML namespaces and XML schemata 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 the appendix of the document.

Request for managing a namespace list:

the schemata of the embedded structured information are maintained outside of the IETF currently, but the list of the embedded specifications' namespaces need to be registered to IANA repository. The initial list of the namespaces are shown in Figure 1.

8. Acknowledgment

The following groups and individuals, listed alphabetically, contributed substantially to this document and should be recognized for their efforts.

Paul Cichonski, NIST

Black David, EMC

Robert Martin, MITRE

Kathleen Moriarty, EMC

Lagadec Philippe, NATO

Shuhei Yamaguchi, NICT

Anthony Rutkowski, Yaana Technology

Brian Trammell, CERT/NetSA

9. Appendix: 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 Section 5 should be verified to validate against this schema by automated tools. [Note: this section will be thoroughly checked later.]

```
<?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"/>

  <!--=====
  == Scoring Class ==
  =====>

  <xsd:element name="Scoring">
```

```

    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="Score" type="xsd:anyType" />
      </xsd:sequence>
      <xsd:attribute name="Version" type="xsd:string" use="optional"
        default="1.00" />
      <xsd:attribute name="SpecificationID" type="xsd:string"
        use="required" />
    </xsd:complexType>
  </xsd:element>

<!--=====
== AttackPattern Class                                ==
=====-->

  <xsd:element name="AttackPattern">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="Record" type="xsd:anyType" minOccurs="0"
          maxOccurs="unbounded" />
        <xsd:element name="Reference" ref="iodef:Reference"
          minOccurs="0" maxOccurs="unbounded" />
        <xsd:element name="PlatformID" ref="iodef-sci:PlatformID"
          minOccurs="0" maxOccurs="unbounded" />
      </xsd:sequence>
      <xsd:attribute name="Version" type="xsd:string" use="optional"
        default="1.00" />
      <xsd:attribute name="SpecificationID" type="xsd:string" use="required" />
      <xsd:attribute name="AttackPatternID" type="xsd:string" use="optional" />
    </xsd:complexType>
  </xsd:element>

<!--=====
== Vulnerability Class                                ==
=====-->

  <xsd:element name="Vulnerability">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="Record" type="xsd:anyType" minOccurs="0"
          maxOccurs="unbounded" />
        <xsd:element name="Reference" ref="iodef:Reference"
          minOccurs="0" maxOccurs="unbounded" />
        <xsd:element name="PlatformID" ref="iodef-sci:PlatformID"
          minOccurs="0" maxOccurs="unbounded" />
        <xsd:element name="Scoring" ref="iodef-sci:Scoring"
          minOccurs="0" maxOccurs="unbounded" />
      </xsd:sequence>

```

```

    <xsd:attribute name="Version" type="xsd:string"
      use="optional" default="1.00"/>
    <xsd:attribute name="SpecificationID" type="xsd:string"
      use="required"/>
    <xsd:attribute name="VulnerabilityID" type="xsd:string"
      use="optional"/>
  </xsd:complexType>
</xsd:element>

<!--=====
== Weakness Class ==
=====-->

<xsd:element name="Weakness">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="Record" type="xsd:anyType" minOccurs="0"
        maxOccurs="unbounded"/>
      <xsd:element name="Reference" ref="iodef:Reference"
        minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element name="PlatformID" ref="iodef-sci:PlatformID"
        minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element name="Scoring" ref="iodef-sci:Scoring"
        minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
    <xsd:attribute name="Version" type="xsd:string" use="optional"
      default="1.00"/>
    <xsd:attribute name="SpecificationID" type="xsd:string"
      use="required"/>
    <xsd:attribute name="WeaknessID" type="xsd:string"
      use="optional"/>
  </xsd:complexType>
</xsd:element>

<!--=====
== PlatformID Class ==
=====-->

<xsd:element name="PlatformID">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="ID" type="xsd:string" minOccurs="1"
        maxOccurs="unbounded"/>
    </xsd:sequence>
    <xsd:attribute name="Version" type="xsd:string" use="optional"
      default="1.00"/>
    <xsd:attribute name="SpecificationID" type="xsd:string"
      use="required"/>

```

```

    </xsd:complexType>
  </xsd:element>

<!--=====
== EventReport Class                                ==
=====-->

<xsd:element name="EventReport">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:choice>
        <xsd:element name="Record" type="xsd:anyType"/>
        <xsd:element name="Reference" ref="iodef:Reference"/>
      </xsd:choice>
    </xsd:sequence>
    <xsd:attribute name="Version" type="xsd:string"
      use="optional" default="1.00"/>
    <xsd:attribute name="SpecificationID" type="xsd:string"
      use="required"/>
  </xsd:complexType>
</xsd:element>

<!--=====
== Remediation Class                                ==
=====-->

<xsd:element name="Remediation">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:choice>
        <xsd:element name="Record" type="xsd:anyType"/>
        <xsd:element name="Reference" ref="iodef:Reference"/>
      </xsd:choice>
    </xsd:sequence>
    <xsd:attribute name="Version" type="xsd:string"
      use="optional" default="1.00"/>
    <xsd:attribute name="SpecificationID" type="xsd:string"
      use="required"/>
  </xsd:complexType>
</xsd:element>

</xsd:schema>

```

Example Schema Diagram

10. References

10.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC5070] Danyliw, R., Meijer, J., and Y. Demchenko, "The Incident Object Description Exchange Format", RFC 5070, December 2007.
- [RFC6045] Moriarty, K., "Real-time Inter-network Defense (RID)", RFC 6045, November 2010.
- [RFC6046] Moriarty, K. and B. Trammell, "Transport of Real-time Inter-network Defense (RID) Messages", RFC 6046, November 2010.

10.2. 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.
- [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.
- [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.
- [CVSS] Peter Mell, Karen Scarfone, and Sasha Romanosky, "The Common Vulnerability Scoring System (CVSS) and Its Applicability to Federal Agency Systems".
- [CAPEC] The MITRE Corporation, "Common Attack Pattern Enumeration and Classification (CAPEC)".
- [CEE] The MITRE Corporation, "Common Event Expression (CEE)".

- [CPE] Brant A. Cheikes and David Waltermire and Karen Scarfone, "Common Platform Enumeration: Naming Specificatino Version 2.3", Aug 2011.
- [CVE] The MITRE Corporation, "Common Vulnerability and Exposures (CVE)".
- [CVRF] ICASI, "<http://www.icasl.org/cvrf>".
- [CWE] The MITRE Corporation, "Common Weakness Enumeration (CWE)".
- [CWSS] The MITRE Corporation, "Common Weakness Scoring System (CWSS)".
- [ISO/IEC19770-2] ISO/IEC, "Information technology -- Software asset management -- Part 2: Software identification tag", 2009.
- [OCIL] David Waltermire and Karen Scarfone and Maria Casipe, "The Open Checklist Interactive Language (OCIL) Version 2.0", Apr 2011.
- [OVAL] The MITRE Corporation, "Open Vulnerability and Assessment Language (OVAL)".
- [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)", Jul 2011.
- [XDAS] The Open Group, "Distributed Audit Service (XDAS), Preliminary Specification", Jan 1998.

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

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

Email: 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

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

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

Individual Submission
Internet-Draft
Intended status: Informational
Expires: January 27, 2012

B. Trammell
ETH Zurich
July 26, 2011

Expert Review for IODEF Extensions in IANA XML Registry
draft-trammell-mile-iodef-xmlreg-00.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 January 27, 2012.

Copyright Notice

Copyright (c) 2011 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].

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 include a note at the top of <http://www.iana.org/assignments/xml-registry/schema.html>, stating "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]," and naming the designated expert.]

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

Individual Submission
Internet-Draft
Intended status: Informational
Expires: January 27, 2012

B. Trammell
ETH Zurich
July 26, 2011

Guidelines for Extensions to IODEF for Managed Incident Lightweight
Exchange
draft-trammell-mile-template-01.txt

Abstract

This document provides guidelines for extensions to IODEF [RFC5070] for lightweight 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 January 27, 2012.

Copyright Notice

Copyright (c) 2011 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. Document Template	5
4.1. Introduction	5
4.2. Terminology	6
4.3. Applicability	6
4.4. Extension Definition	6
4.4.1. IODEF Data Types	7
4.4.2. Example Enumerated Type Extension Definition: E.164 Address	8
4.4.3. Example Element Definition: Test	8
4.5. Examples	9
4.6. Security Considerations	9
4.7. IANA Considerations	10
4.8. Appendix: XML Schema Definition for Extension	11
5. Security Considerations	11
6. IANA Considerations	11
7. References	11
7.1. Normative References	11
7.2. Informative References	11
Author's Address	12

1. Introduction

Since the specification of IODEF [RFC5070], the evolution of the threat environment and the practice of cooperative network defense, along with continued implementation and deployment, has indicated the need for guidelines for the implementation and extension of IODEF. This document provides these guidelines. It starts by describing the applicability of IODEF extensions, and the IODEF extension mechanisms, before providing a section Section 4 that is itself designed to be copied out and filled in as the starting point of an Internet-Draft about an IODEF extension.

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 information about an incident? "Incident" is not defined in the terminology for IODEF, but for purposes of IODEF can be loosely described as "something that happened that has some impact on the information security situation of an entity", with quite a bit of leeway for interpretation. 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 reasonably rich set of incident-relevant classes. If, after examination of the problem area and the IODEF specification, the answer to this question is "Yes", then extension is not necessary.

A non-exhaustive list of good candidate extensions to IODEF includes:

1. Allowing standardized reference to external information bases about incidents and incident-relevant information: leveraging existing work in describing aspects of incidents to make IODEF more expressive
2. Allowing the description of new types of entities (e.g., related actors) or new types of characteristics of entities (e.g., financial services-related information) involved in an IODEF incident report
3. 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 and 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.

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 with IODEF within external schemas. Extensions using containment of an IODEF-Document are not further treated in this document, though the document template in Section 4 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 in this way are defined in [RFC5070], and limited to only these:

- o Incident@purpose
- o Contact@role
- o Contact@type
- o RegistryHandle@registry
- o Impact@type
- o TimeImpact@metric
- o TimeImpact@duration
- o HistoryItem@action

- o Expectation@action
- o System@category
- o Counter@type
- o Counter@duration
- o Address@category
- o NodeRole@category
- o RecordPattern@type
- o RecordPattern@offsetunit
- o AdditionalData@dtype
- o RecordItem@dtype

An example definition of an attribute extension is given in Section 4.4.2.

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 4.4.1, and SHOULD define the 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 root element within the AdditionalData or RecordItem attribute. An example definition of an element definition is given in Section 4.4.3.

4. Document Template

Documents describing an IODEF extension should follow the document template given in this section.

4.1. Introduction

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

4.2. Terminology

The terminology section introduces and defines terms specific to the document. Terminology from [RFC5070] or [RFC6045] 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.

4.3. 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 particular problem. This should also 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.

4.4. 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 the new value. An example enumeration extension is shown in Section 4.4.2, 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 Section 4.4.3, 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 MILE extension document.

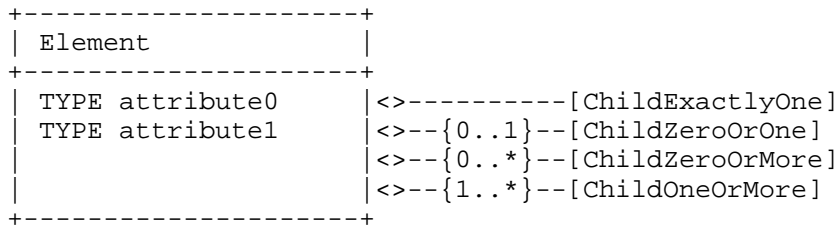


Figure 1: Example UML Element Diagram

Elements containing child elements should indicate the multiplicity of those child elements, as shown in the figure above. Allowable TYPES are discussed in the following subsection.

4.4.1. IODEF Data Types

The allowable TYPES for attributes within IODEF are enumerated in section 2 of [RFC5070], and consist of:

- o INTEGER
- o REAL
- o CHARACTER
- o STRING
- o ML_STRING (for strings in encodings other than that of the enclosing document)
- o BYTE for bytes or byte vectors in Base 64 encoding
- o HEXBIN for bytes in ascii-hexadecimal encoding
- o ENUM for enumerated types; allowable values of the enumeration must be defined in the attribute definition
- o DATETIME for ISO 8601:2000 [RFC3339] encoded timestamps
- o TIMEZONE for timezones as encoded in section 2.9 of [RFC5070].
- o PORTLIST for port lists as encoded in section 2.10 of [RFC5070].
- o POSTAL for postal addresses as defined in section 2.23 of [RFC4519].

- o NAME for names of natural or legal persons as defined in section 2.3 of [RFC4519].
- o PHONE for telephone numbers as defined in section 2.35 of [RFC4519].
- o EMAIL for email addresses as defined in section 3.4.1. of [RFC2822].
- o URL for URLs as in [RFC2396].

In addition to these simple data types, IODEF provides a compound data type for representing network address information. Addresses included within an extension element should be represented by containing an IODEF:Address element, which supports IPv4 and [RFC2373] IPv6 addresses, as well as MAC, ATM, and BGP autonomous system numbers. Application-layer addresses should be represented with the URL simple attribute type, instead.

4.4.2. Example Enumerated Type Extension Definition: E.164 Address

This example extends the IODEF Address element to support the encoding of ENUM-mapped telephone numbers [RFC6116].

Attribute: Address@category

Extended value(s): enum-e164

Content format: An E.164 telephone number encoded as a domain name in the e164.int space, e.g. "2.1.2.1.5.5.5.2.1.2.1.e164.int." for +1 212 555 1212, as per section 3.2 of [RFC6116].

Additional considerations: none.

4.4.3. 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 reference to a real incident.

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

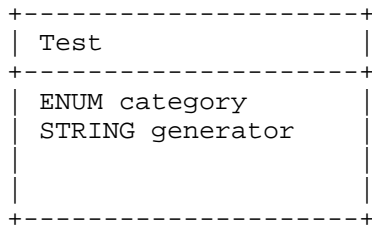


Figure 2: The Test class

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.

4.5. 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 should be tested to validate against the schema given in the appendix.

4.6. 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 5 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 or storage of the information represented by an extension.

4.7. 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 6 for IANA Considerations for this document.]

Any IANA considerations [RFC5226] for the document should be detailed in this section; if none, the section should exist and contain the text "this document has no actions for IANA".

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 the appendix of the document, or to a well-known external reference in the case of an extension with an externally-defined schema.

4.8. Appendix: 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 section Section 4.5 should be verified to validate against this schema by automated tools.

5. Security Considerations

This document defines a template for MILE extensions to the IODEF and RID documents; as such, it has no security considerations on its own.

6. IANA Considerations

This document has no actions for IANA.

7. References

7.1. Normative References

- [RFC5070] Danyliw, R., Meijer, J., and Y. Demchenko, "The Incident Object Description Exchange Format", RFC 5070, December 2007.
- [RFC6045] Moriarty, K., "Real-time Inter-network Defense (RID)", RFC 6045, November 2010.

7.2. Informative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2373] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 2373, July 1998.
- [RFC2396] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifiers (URI): Generic Syntax", RFC 2396, August 1998.
- [RFC2822] Resnick, P., "Internet Message Format", RFC 2822, April 2001.
- [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.
- [RFC4519] Sciberras, A., "Lightweight Directory Access Protocol (LDAP): Schema for User Applications", RFC 4519, June 2006.
- [RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 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.

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

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 6, 2012

A. Vesely
October 4, 2011

IODEF Extension to Support Mail Abuse Reporting
draft-vesely-mile-mail-abuse-00

Abstract

This document extends the Incident Object Description Exchange Format (IODEF) to allow mail-abuse reports to be shared as Incidents in the IODEF framework.

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 April 6, 2012.

Copyright Notice

Copyright (c) 2011 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
3.1. Conversion to IODEF	4
3.2. Conversion from IODEF	4
4. Extension Definition	4
4.1. AbuseReport	5
4.1.1. Text	5
4.1.2. ArfHeader	6
4.1.3. EmailMessage	7
4.2. Note on Newline Representation	7
5. Example	7
6. IANA Considerations	9
7. Security Considerations	9
8. References	9
8.1. Normative References	9
8.2. Informative References	9
Appendix A. ARF Extension Schema	10
Author's Address	12

1. Introduction

Spam victims may tag some received messages as abusive. The Abuse Reporting Format [ARF] was developed for reporting those messages to where they can be taken care of. For example, [ARF] is used by Mailbox Providers to forward abuse reports to senders who had established a Feedback Loop [FBL].

Mail-abuse can be reported to a Network Provider related to the IP address that relayed an abusive email message, whether using ARF or simply attaching the abusive message to a complaint, depending on the tools at hand. Both the "abuse@domain" role address specified by [MAILBOX-NAMES] and the abuse-mailbox contacts as found in whois databases are possible targets of unsolicited abuse reports. In addition, [REPORTING-DISCOVERY] defines ways to discover or publish contacts for FBL subscriptions.

When received by a party held responsible for the abusive message, an abuse report may highlight the need for corrective actions that can be carried out more conveniently if the report is converted in the Incident Object Description Exchange Format [IODEF]. For example, the original message may turn out to be a phishing attempt and needs to be further converted to the format defined by [FRAUD-EXT], it may call for trace-back, mitigation, or further reporting, or it may reveal that the emitting machine is infected.

This [IODEF] extension defines a representation of an abuse-report, whereby a IODEF Incident may contain one or more complaints, either ARF or plain text with a message attachment. It is beyond the scope of this memo to outline possible use cases, albeit they are mentioned above as examples.

2. Terminology

The terminology used in this document follows the one defined in [IODEF] and [TEMPLATE].

Network provider is defined in [RID].

Mailbox Provider is defined in [FBL]. It includes email facilities in corporations, universities, and similar organizations.

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

3.1. Conversion to IODEF

Network providers may want to convert abuse reports received over [SMTP] into EventData instances to deploy IODEF infrastructures. Conversion notes are given along with the definition (Section 4).

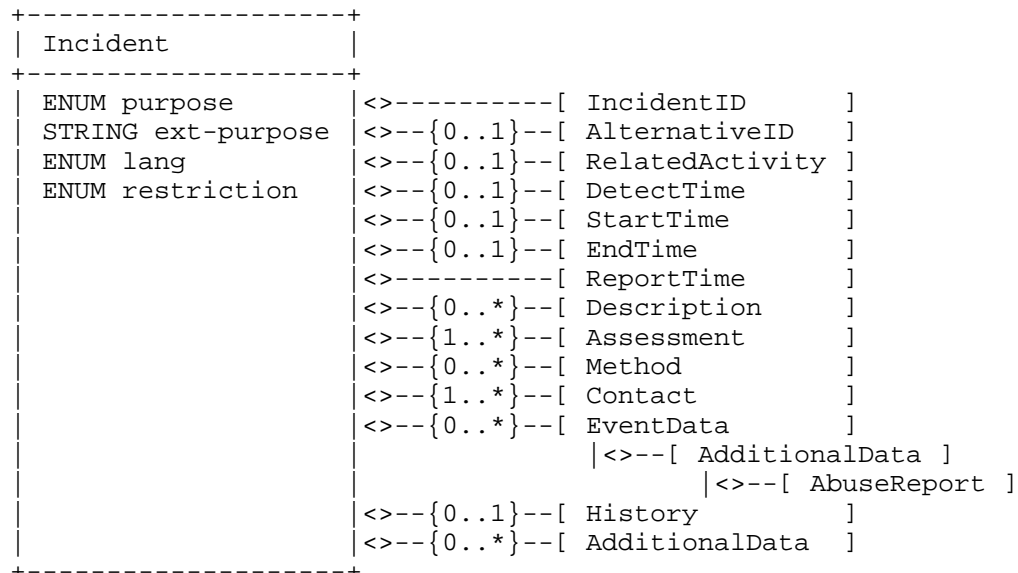
3.2. Conversion from IODEF

An Incident representing an abuse report may need to be converted back to an email message for transmitting it via SMTP if, for example, the target party does not support IODEF.

[ARF] is designed to be readable without specific tools, and machine-readable as well. It is advisable to use ARF, so as to allow automated processing. However, plain text with attachment(s) has to be used when it is necessary to convey information which cannot be formatted in ARF, as its human-readable part is likely to be discarded by automated processes.

4. Extension Definition

This extension defines a simple structure to represent an abuse report. The following diagram is reproduced from [IODEF] and modified to show where the AbuseReport class lies.



The AbuseReport position within the Incident Class

An incident normally refers to the abusive message, not to the act of reporting it. However, the EventData instance that hosts an AbuseReport SHOULD contain information about the report sender. If the report was sent via [SMTP], its header may contain authentication information such as [DKIM] or [SPF], possibly validated on reception and reported in an Authentication-Result [A-R] header field. The [EMAIL] header of the report will not be part of the AbuseReport. Therefore it is RECOMMENDED that any assessment on authenticity and trust be conveyed using general [IODEF] classes. By contrast, note that the full header of the abusive message being reported MUST be present, as abuse reports don't make sense without it.

4.1. AbuseReport

The AbuseReport structure closely resembles either [ARF] or a plain text complaint with an abusive message attached.

```
+-----+
| AbuseReport |
+-----+
|               | <>--{0..1}--[ Text           ]
|               | <>--{0..1}--[ ArfHeader      ]
|               | <>-----[ EmailMessage  ]
+-----+
```

AbuseReport structure

At least one of Text or ArfHeader SHOULD be present. The meaning of each element is as follows:

Text: Zero or one. ML_STRING. The human-readable part of an [ARF] report or the textual part of a manually submitted report.

ArfHeader: Zero or one. The machine-readable part of an [ARF] report.

EmailMessage: One. ML_STRING. The abusive message reported.

4.1.1. Text

Zero or one. ML_STRING.

Meaningful values of header fields MAY be retained from the report's header. In particular, the following fields are often considered meaningful: From, Subject, Date, To, CC, Reply-To. However, if the Subject merely repeats that of the reported message, and the Date as

well as the relevant contacts are being conveyed in other parts of the EventData, it is not necessary to repeat those values here.

The human-readable part of machine generated [ARF] reports usually consists of boilerplate informing what is an ARF message. Such kind of text MAY be omitted, and if no header field is retained, the Text element MAY be omitted altogether.

4.1.2. ArfHeader

The presence of this element indicates that this Incident represents an [ARF] report, as opposed to a manually written complaint.

```
+-----+
| ArfHeader |
+-----+
|           |<--{0..*}--[ Field ]
+-----+
```

ArfHeader structure

An ArfHeader consists of a sequence of fields.

Field: Zero or more. A field is a name/value pair.

[ARF] defines required and optional fields, appearing once or multiple times. It is an extensible specification, and there are two IANA registries tracking report types and allowed field names respectively. The order of the fields is unimportant.

```
+-----+
| Field |
+-----+
| STRING Name |
+-----+
```

ArfHeader Field

A Field is a name/value pair. The name is represented as an attribute, while the value is its content. [ARF] header fields are formatted like [EMAIL] header fields; that is, a name followed by a colon (":") and a value.

name: One. Restricted STRING. Allowable names are restricted by [EMAIL] to printable US-ASCII characters not including the colon. We further restrict names to lowercase, since they are meant to be case insensitive while the values of XML attributes are not.

element content: STRING. Leading and trailing whitespace MAY be omitted. Any internal newlines MAY be retained. If an internal newline is retained, some, but not all, of the leading whitespace in the following line MAY be omitted. See also Section 4.2.

4.1.3. EmailMessage

An EmailMessage contains one ML_STRING. This is the [EMAIL] message content; that is, the header possibly followed by the body. The header MUST be present, and the body SHOULD be present.

4.2. Note on Newline Representation

[EMAIL] states that messages consists of multiple lines, and that the CRLF two-byte sequence is the line termination when messages are transmitted over [SMTP]. The header is a sequence of header fields, where each line starting with a non-whitespace character is the beginning of a field. The line length is limited to 78 characters, excluding the CRLF, hence the limit on valid field names. Field values may span multiple lines, provided that each successive line starts with whitespace. An empty line delimits the end of the header.

[XML] end-of-line handling provides for CRLF to be normalized as LF. This poses no problems, as many mail agents handle local storage in the same way. Conversion to/from IODEF has to adapt to local conventions for receiving/sending email messages.

5. Example

The "simple report" given in Appendix B.1 of RFC 5965 can be converted as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<IODEF-Document lang="en-US"
  xmlns:arf="urn:ietf:params:xml:ns:iodef-arf-1.0"
  xmlns="urn:ietf:params:xml:ns:iodef-1.0"
  xmlns:iodef="urn:ietf:params:xml:ns:iodef-1.0">
  <Incident purpose="reporting">
    <IncidentID name="example.net">FBL20050308-3</IncidentID>
    <ReportTime>2005-03-08T17:40:36-04:00</ReportTime>
    <Assessment>
      <Impact type="policy" lang="en"/>
    </Assessment>
    <Contact role="creator" type="organization">
      <ContactName>example.net</ContactName>
      <Email>abuse@example.net</Email>
```

```
</Contact>
<EventData>
  <DetectTime>2005-03-08T17:40:36-04:00</DetectTime>
  <Contact role="irt" type="organization">
    <ContactName>example.com</ContactName>
    <Description>Feedback Generator</Description>
    <Email>abusedesk@example.com</Email>
  </Contact>
  <Flow>
    <System>
      <Node>
        <NodeName>fbl-out.example.com</NodeName>
        <Address category="ipv4-addr">192.0.2.129</Address>
      </Node>
    </System>
  </Flow>
  <AdditionalData dtype="xml">
    <arf:AbuseReport>
      <arf:ArfHeader>
        <arf:Field name="feedback-type">abuse</arf:Field>
        <arf:Field name="user-agent">SomeGenerator/1.0</arf:Field>
        <arf:Field name="version">1</arf:Field>
      </arf:ArfHeader>
      <arf:EmailMessage>
Received: from mailserver.example.net
(mailserver.example.net [192.0.2.1])
by example.com with ESMTP id M63d4137594e46;
Thu, 08 Mar 2005 14:00:00 -0400
From: &lt;somespammer@example.net&gt;
To: &lt;Undisclosed Recipients&gt;
Subject: Earn money
MIME-Version: 1.0
Content-type: text/plain
Message-ID: 8787KJKJ3K4J3K4J3K4J3.mail@example.net
Date: Thu, 02 Sep 2004 12:31:03 -0500

Spam Spam Spam
Spam Spam Spam
Spam Spam Spam
Spam Spam Spam
      </arf:EmailMessage>
    </arf:AbuseReport>
  </AdditionalData>
</EventData>
</Incident>
</IODEF-Document>
```

ARF Converted to Incident

6. IANA Considerations

TBD.

7. Security Considerations

TBD.

8. References

8.1. Normative References

- [ARF] Shafranovich, Y., Levine, J., and M. Kucherawy, "An Extensible Format for Email Feedback Reports", RFC 5965, August 2010.
- [IODEF] Danyliw, R., Meijer, J., and Y. Demchenko, "The Incident Object Description Exchange Format", RFC 5070, December 2007.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [XML] Yergeau, F., Bray, T., Sperberg-McQueen, C., Maler, E., 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>>.

8.2. Informative References

- [A-R] Kucherawy, M., "Message Header Field for Indicating Message Authentication Status", RFC 5451, April 2009.
- [DKIM] Allman, E., Callas, J., Delany, M., Libbey, M., Fenton, J., and M. Thomas, "DomainKeys Identified Mail (DKIM) Signatures", RFC 4871, May 2007.
- [EMAIL] Resnick, P., Ed., "Internet Message Format", RFC 5322, October 2008.
- [FBL] Falk, J., "Creation and Use of Email Feedback Reports: An Applicability Statement for the Abuse Reporting Format (ARF)", draft-ietf-marf-as-00 (work in progress), September 2011.

[FRAUD-EXT]

Cain, P. and D. Jevans, "Extensions to the IODEF-Documents Class for Reporting Phishing", RFC 5901, July 2010.

[MAILBOX-NAMES]

Crocker, D., "MAILBOX NAMES FOR COMMON SERVICES, ROLES AND FUNCTIONS", RFC 2142, May 1997.

[REPORTING-DISCOVERY]

Falk, J., "A DNS TXT Record for Advertising and Discovering Willingness to Provide or Receive ARF Reports", draft-ietf-marf-reporting-discovery-00 (work in progress), January 2011.

[RID]

Moriarty, K., "Real-time Inter-network Defense (RID)", RFC 6045, November 2010.

[SMTP]

Klensin, J., "Simple Mail Transfer Protocol", RFC 5321, October 2008.

[SPF]

Wong, M. and W. Schlitt, "Sender Policy Framework (SPF) for Authorizing Use of Domains in E-Mail, Version 1", RFC 4408, April 2006.

[TEMPLATE]

Trammell, B., "Guidelines for Extensions to IODEF for Managed Incident Lightweight Exchange", draft-trammell-mile-template-01 (work in progress), July 2011.

Appendix A. ARF Extension Schema

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

```
<!--
```

```
=====
=== Top-Level Class: AbuseReport ===
=====
```

It is incorporated within an

IODEF.Incident.EventData.AdditionalData element.

-->

```
<xs:element name="AbuseReport">
  <xs:complexType>
    <xs:sequence>

      <!-- the readable text part (boilerplate may be omitted) -->
      <xs:element minOccurs="0" name="Text"
        type="iodef:MLStringType"/>

      <!-- an ArfHeader is present iff the report was in ARF format -->
      <xs:element minOccurs="0" name="ArfHeader">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="unbounded" name="Field">
              <xs:complexType>
                <xs:simpleContent>
                  <xs:extension base="xs:string">
                    <!-- the name attribute indicates the field-name,
                        as defined by RFC 5322 (and RFC5335bis) i.e.
                        printable US-ASCII characters not including ":",
                        we additionally forbid uppercase letters -->
                    <xs:attribute name="name" use="required">
                      <xs:simpleType>
                        <xs:restriction base="xs:string">
                          <xs:pattern value="([&#33;-&#126;-[:A-Z]]{1,77}"/>
                        </xs:restriction>
                      </xs:simpleType>
                    </xs:attribute>
                  </xs:extension>
                </xs:simpleContent>
              </xs:complexType>
            </xs:element>
          </xs:sequence>
        </xs:complexType>
      </xs:element>

      <!-- the original message (header + body) -->
      <xs:element name="EmailMessage"
        type="iodef:MLStringType"/>

    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>
```

Author's Address

Alessandro Vesely
v. L. Anelli 13
Milano, MI 20122
IT

Email: vesely@tana.it

