MILE Working Group
Internet-Draft
Obsoletes: 5070, 6685 (if approved)
Intended status: Standards Track
Expires: April 8, 2017

R. Danyliw
CERT

October 5, 2016

The Incident Object Description Exchange Format v2
draft-ietf-mile-rfc5070-bis-26

Abstract

The Incident Object Description Exchange Format (IODEF) defines a data representation for security incident reports and indicators commonly exchanged by operational security teams for mitigation and watch and warning. This document describes an updated information model for the IODEF and provides an associated data model specified with XML Schema. This new information and data model obsoletes Request for Comment (RFC) 5070 and 6685.

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 8, 2017.

Copyright Notice

Table of Contents

1. Introduction

Organizations require help from other parties to mitigate malicious activity targeting their network and to gain insight into potential threats. This coordination might entail working with an ISP to filter attack traffic, contacting a remote site to take down a botnet, or sharing watch-lists of known malicious indicators in a consortium.

The Incident Object Description Exchange Format (IODEF) is a format for representing computer security information commonly exchanged between Computer Security Incident Response Teams (CSIRTs) or other operational security teams. It provides an XML representation for conveying:

- indicators to characterize a threat;
- security incident reports to document attacks against an organization;
- response activity taken or that could be taken in response to an incident; and
- meta-data so that these various classes of information can be exchanged among parties.

The purpose of the IODEF is to enhance the operational capabilities of CSIRTs. Adoption of the IODEF will improve the ability of a CSIRT to resolve security incidents; understand threats; and coordinate response activities and proactive mitigations by simplifying collaboration and data sharing with its partners. This structured format provided by the IODEF allows for:

- machine-to-machine exchange of incident and indicator data;
- automated processing of this data whereby allowing more rapid execution of appropriate courses of action; and
- the development of an ecosystem of interoperable tools enabling security operations.

Sharing and coordinating with other organizations is not strictly a technical problem. There are numerous procedural, cultural, legal and trust-related barriers to overcome. The IODEF does not attempt to address them directly. However, operational implementations of the IODEF will need to consider these challenges.

Section 1 provides the background for the IODEF. Sections 3 and 8 specify the IODEF information and data model respectively. The data types used in this document are described in Section 2. Processing considerations, extending the specification, internationalization and security issues are covered in Sections 4, 5, 6 and 9 respectively. Examples are listed in Section 7.

1.1. 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].

1.2. Notations

The IODEF is specified as an Extensible Markup Language (XML) [W3C.XML] Schema [W3C.SCHEMA]. The normative IODEF data model is found in the XML schema in Section 8. To aid in the understanding of the data elements, Section 3 also depicts the underlying information model using Unified Modeling Language (UML). This abstract presentation of the IODEF is not normative.

For clarity in this document, the term "XML document" will be used when referring generically to any instance of an XML document. The term "IODEF document" will be used to refer to an XML document conforming to the IODEF specification. The terms "schema" will be used to refer to Section 8 of this document. The terms "data model" and "schema" will be used interchangeably. The terms "class" and "element" will be used to reference either the corresponding data element in the UML-based information or XML Schema-based data models, respectively.

1.3. About the IODEF Data Model

A number of considerations were made in the design of the IODEF data model.

- The data model found in this document is an evolution of the one previously specified in [RFC5070]. New fields were added to represent additional information. [RFC5070] was developed primarily to represent incident reports. This document builds upon it by adding support for indicators and revising it to reflect the current challenges faced by CSIRTs. An attempt was made to preserve backward compatibility but this was not possible in all cases. See Section 4.4. This document obsoletes [RFC5070].

- The IODEF is a transport format. Therefore, the data model may not be the optimal archival or in-memory processing format.

- The IODEF is intended to be a framework to convey only commonly exchanged information. It ensures that there are mechanisms for extensibility to support organization-specific information and techniques to reference information kept outside of the data model.

- Not all commonly exchanged information has a well-defined format or taxonomy. The IODEF attempts to strike a balance between enforcing sufficient structure to allow automated processing and supporting free-form content that enables maximum flexibility.

- The IODEF fits into a broader ecosystem of standards and conventions. An attempt was made to harmonize the data model with this context.

1.4. Changelog

A detailed list of additions made to the [RFC5070] data model are enumerated in this section. See Section 4.4 for a list of incompatible changes.

- Updated the data types (Section 2) to improve internationalization, clarify ambiguity, and ensure consistency in extensions.

- Added the observable-id attribute (Section 3.3.2) and IndicatorData (Section 3.28) class (Section 3.28) to represent indicators.

- Added the private-enum-name and -id attributes to the IODEF-Document class (Section 3.1) to disambiguate private extensions.

- Updated the Incident class (Section 3.2) to represent additional timing and workflow information.

- Added the ThreatActor (Section 3.7) and Campaign (Section 3.8) classes to represent attack attribution information.

- Updated the Contact class (Section 3.9) and its children to improve internationalization and represent additional information about an entity.

- Updated the Method class (Section 3.11) to improve extensibility through externally referenced resources.

- Added the Discovery class (Section 3.10) to describe how an incident was discovered.
- Updated the Assessment class (Section 3.12) to enable more descriptive characterizations of the impact of an incident.
- Updated the HistoryItem (Section 3.13.1) and Expectation (Section 3.15) classes to support a reference to a course of action.
- Updated the EventData class (Section 3.14) with additional meta-data added to the Incident class.
- Updated the System (Section 3.17) class with additional meta-data.
- Updated the Counter class (Section 3.18.3) to support additional rate metrics.
- Added the DomainData (Section 3.19), EmailData (Section 3.21), WindowsRegistryKeysModified (Section 3.23), CertificateData (Section 3.24) and FileData (Section 3.25) to improve the description of an incident and support this data as indicators.
- Added the SignatureData (Section 3.27) and HashData classes (Section 3.26) to represent digital signatures and hashes.
- Added support for public enumerated attribute extensions using IANA registries (Section 5.1.2).
- Updated numerous enumerated attributes for completeness.

2. IODEF Data Types

The IODEF uses a number of simple and complex types. This section describes these data types.

2.1. Integers

An integer is represented in the information model by the INTEGER data type. Integer data MUST be encoded in Base 10.

The INTEGER data type is implemented in the data model as a "xs:integer" type per Section 3.3.13 of [W3C.SCHEMA.DTYPES].

2.2. Real Numbers

A real (floating-point) number is represented in the information model by the REAL data type. Real data MUST be encoded in Base 10.

The REAL data type is implemented in the data model as a "xs:float" type per Section 3.2.4 of [W3C.SCHEMA.DTYPES].

2.3. Characters and Strings

A single character is represented in the information model by the CHARACTER data type. A string is represented by the STRING data type. Special characters MUST be encoded using entity references. See Section 4.1.

The CHARACTER and STRING data types are implemented in the data model as a "xs:string" type per Section 3.2.1 of [W3C.SCHEMA.DTYPES].

2.4. Multilingual Strings

A string that needs to be represented in a human-readable language different than the default encoding of the document is represented in the information model by the ML_STRING data type.

The ML_STRING data type is implemented in the data model as the "iodef:MLStringType" type. This type extends the "xs:string" to include two attributes.

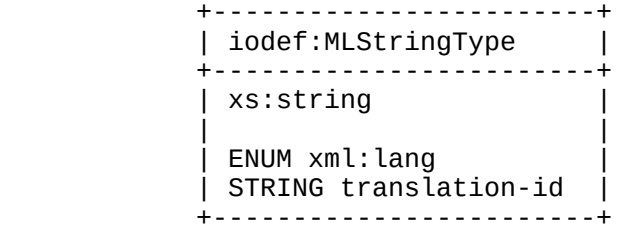

Figure 1: The iodef:MLStringType Type

The content of the class is a character string of type "xs:string" whose language MAY be specified by the xml:lang attribute.

The attributes of the iodef:MLStringType type are:

xml:lang
: Optional. ENUM. A language identifier per Section 2.12 of [W3C.XML] whose values and format are described in [RFC5646]. The interpretation of this code is described in Section 6.

translation-id
: Optional. STRING. An identifier to relate other instances of this class with the same parent as translations of this text. The scope of this identifier is limited to all of the direct, peer child classes of a given parent class.

Using this class enables representing translations of the same text in multiple languages. Each translation is a distinct instance of this class with a common parent. A group of classes each with a translated instance of text is related by setting a common identifier in the translation-id attribute. The language of a given class is set by the xml:lang attribute. See Section 6 for more details on representing translations of free-form text.

2.5. Binary Strings

Binary octets can be represented with two encodings.

2.5.1. Base64 Bytes

A binary octet encoded with Base64 is represented in the information model by the BYTE data type. A sequence of these octets is of the BYTE[] data type.

The BYTE and BYTE[] data types are implemented in the data model as a "xs:base64Binary" type per Section 3.2.16 of [W3C.SCHEMA.DTYPES].

2.5.2. Hexadecimal Bytes

A binary octet encoded as a character tuple consistent of two hexadecimal digits is represented in the information model by the HEXBIN data type. A sequence of these octets is of the HEXBIN[] data type.

The HEXBIN and HEXBIN[] data types are implemented in the data model as a "xs:hexBinary" type per Section 3.2.15 of [W3C.SCHEMA.DTYPES].

2.6. Enumerated Types

An enumerated type is represented in the information model by the ENUM data type. It is an ordered list of acceptable string values. Each value has a representative keyword. Within the data model, the enumerated type keywords are used as attribute values.

The ENUM data type is implemented in the data model as values of a "xs:NMTOKEN" type per Section 3.3.4 of [W3C.SCHEMA.DTYPES].

2.7. Date-Time String

A date-time strings that describes a particular instant in time is represented in the information model by the DATETIME data type. Ranges are not supported.

The DATETIME data type is implemented in the data model as a "xs:dateTime" type per Section 3.2.7 of [W3C.SCHEMA.DTYPES].

2.8. Timezone String

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

The TIMEZONE data type is implemented in the data model as an "iodef:TimezoneType" type.

2.9. Port Lists

A list of network ports is represented in the information model by the PORTLIST data type. A PORTLIST consists of a comma-separated list of numbers and ranges (N-M means ports N through M, inclusive). It is formatted according to the following regular expression: "\d+(\-\d+)?(,\d+(\-\d+)?)*". For example, "2,5-15,30,32,40-50,55-60".

The PORTLIST data type is implemented in the data model as an "iodef:PortlistType" type.

2.10. Postal Address

A postal address is represented in the information model by the POSTAL data type. The format of the POSTAL data type is documented in Section 2.23 of [RFC4519] as a free-form multi-line string separated by the "$" character.

The POSTAL data type is implemented in the data model as an "iodef:MLStringType" type.

2.11. Telephone Number

A telephone number is represented in the information model by the PHONE data type. The format of the PHONE data type is documented in [E.164].

The PHONE data type is implemented in the data model as a "xs:string" type per Section 3.2.1 of [W3C.SCHEMA.DTYPES].

2.12. Email String

An email address is represented in the information model by the EMAIL data type. The format of the EMAIL data type is documented in Section 3.4.1 of [RFC5322] and Section 3.3 of [RFC6531].

The EMAIL data type is implemented in the data model as a "xs:string" type per Section 3.2.1 of [W3C.SCHEMA.DTYPES].

2.13. Uniform Resource Locator strings

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

The URL data type is implemented as a "xs:anyURI" type per Section 3.2.17 of [W3C.SCHEMA.DTYPES].

2.14. Identifiers and Identifier References

An identifier unique to the IODEF document is represented in the information model by the ID data type. A reference to this identifier is represented by the IDREF data type.

The ID and IDREF data types are implemented in the model as "xs:ID" and "xs:IDREF" types per Sections 3.3.8 and 3.3.9 of [W3C.SCHEMA.DTYPES].

2.15. Software

A particular version of software is represented in the information model by the SOFTWARE data type. This software can be described by using a reference, a URL or with free-form text.

The SOFTWARE data type is implemented in the data model as the "iodef:SoftwareType" type.

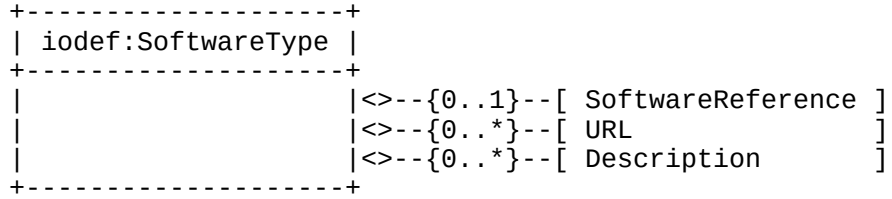

Figure 2: The SoftwareType Type

The aggregate classes of the SoftwareType type are:

SoftwareReference
Zero or one. Reference to a software application. See Section 2.15.1.

URL
Zero or more. URL. A URL to a resource describing the software.

Description
Zero or more. ML_STRING. A free-form text description of the software.

At least one of these classes MUST be present.

The iodef:SoftwareType type has no attributes.

2.15.1. SoftwareReference Class

The SoftwareReference class is a reference to a particular version of software.

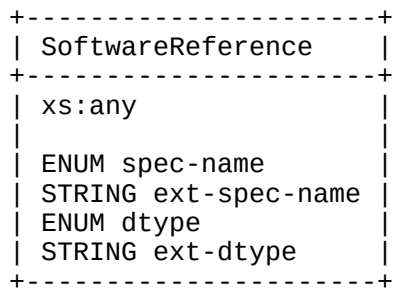

Figure 3: The SoftwareReference Class

The element content varies according to the value of the spec-name attribute. It is defined in the data model as "xs:any" per [W3C.SCHEMA].

The attributes of the SoftwareReference class are:

spec-name
Required. ENUM. Identifies the format and semantics of the element body of this class. Formal standards and specifications can be referenced as well as a free-form text description with a user-provided data type. These values are maintained in the "SoftwareReference-spec-id" IANA registry per Section 10.2

1. custom. The element content is free-form and of the data type specified by the dtype attribute. If this value is selected, then the dtype attribute MUST be set.

2. cpe. The element content describes a Common Platform Enumeration (CPE) entry per [NIST.CPE].

3. swid. The element content describes a software identification (SWID) tag per [ISO19770].

4. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

ext-spec-name
Optional. STRING. A means by which to extend the spec-name attribute. See Section 5.1.1.

dtype
Optional. ENUM. The data type of the element content. The permitted values for this attribute are shown below. The default value is "string". These values are maintained in the "SoftwareReference-dtype" IANA registry per Section 10.2.

1. bytes. The element content is of type HEXBIN.

2. integer. The element content is of type INTEGER.

3. real. The element content is of type REAL.

4. string. The element content is of type STRING.

5. xml. The element content is XML. See Section 5.2.

6. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

ext-dtype
Optional. STRING. A means by which to extend the dtype attribute. See Section 5.1.1.

2.16. Extension

Information not otherwise represented in the IODEF can be added using the EXTENSION data type. This data type is a generic extension mechanism.

The EXTENSION data type is implemented in the data model as the "iodef:ExtensionType" type.

The data type of an EXTENSION is described by the dtype attribute. For simple information, atomic data types (e.g., integers, strings) are supported. Their semantics are further described by the meaning and formatid attributes. Encapsulating XML documents conforming to another schema is also supported. A detailed discussion of extending the schema can be found in Section 5. Additional coordination may be required to ensure that a recipient of a document using this type can parse and process it.

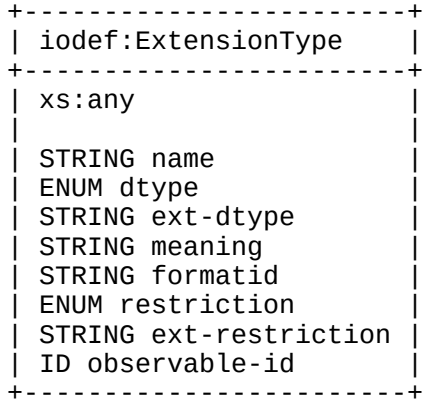

Figure 4: The iodef:ExtensionType Type

The element content of this type is the extension being added to the data model. This content is defined in the data model as "xs:any" per [W3C.SCHEMA].

The attributes of the iodef:ExtensionType type are:

name
: Optional. STRING. A free-form name of the field or data element.

dtype
: Required. ENUM. The data type of the element content. The default value is "string". These values are maintained in the "ExtensionType-dtype" IANA registry per Section 10.2.

1. boolean. The element content is of type BOOLEAN.

2. byte. The element content is of type BYTE.

3. bytes. The element content is of type HEXBIN.

4. character. The element content is of type CHARACTER.

5. date-time. The element content is of type DATETIME.

6. ntpstamp. Same as date-time.

7. integer. The element content is of type INTEGER.

8. portlist. The element content is of type PORTLIST.

9. real. The element content is of type REAL.

10. string. The element content is of type STRING.

11. file. The element content is a base64 encoded binary file encoded as a BYTE[] type.

12. path. The element content is a file-system path encoded as a STRING type.

13. frame. The element content is a layer-2 frame encoded as a HEXBIN type.

14. packet. The element content is a layer-3 packet encoded as a HEXBIN type.

15. ipv4-packet. The element content is an IPv4 packet encoded as a HEXBIN type.

16. ipv6-packet. The element content is an IPv6 packet encoded as a HEXBIN type.

17. url. The element content is of type URL.

18. csv. The element content is a common separated value (CSV) list per Section 2 of [RFC4180] encoded as a STRING type.

19. winreg. The element content is a Windows registry key encoded as a STRING type.

20. xml. The element content is XML. See Section 5.

21. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

ext-dtype

Optional. STRING. A means by which to extend the dtype attribute. See Section 5.1.1.

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

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

restriction
Optional. ENUM. See Section 3.3.1.

ext-restriction
Optional. STRING. A means by which to extend the restriction attribute. See Section 5.1.1.

observable-id
Optional. ID. See Section 3.3.2.

3. The IODEF Information Model

The specifics of the IODEF information model are discussed in this section. Each class and its relationships with the other classes is described. When necessary, clarifications are made about translating this information model to the schema in Section 8.

3.1. IODEF-Document Class

The IODEF-Document class is the top level class in the IODEF data model. All IODEF documents are an instance of this class.

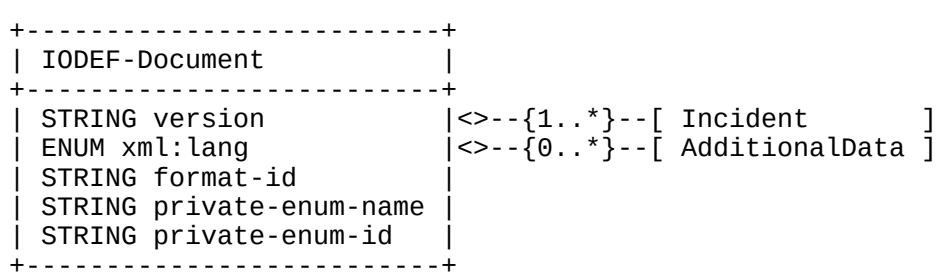

Figure 5: IODEF-Document Class

The aggregate classes of the IODEF-Document class are:

Incident

One or more. The information related to a single incident. See Section 3.2.

AdditionalData
Zero or more. EXTENSION. Mechanism by which to extend the data model.

The attributes of the IODEF-Document class are:

version
Required. STRING. The IODEF specification version number to which this IODEF document conforms. The value of this attribute MUST be "2.00"

xml:lang
Optional. ENUM. A language identifier per Section 2.12 of [W3C.XML] whose values and form are described in [RFC5646]. The interpretation of this code is described in Section 6.

format-id
Optional. STRING. A free-form string to convey processing instructions to the recipient of the document. Its semantics must be negotiated out-of-band.

private-enum-name
Optional. STRING. A globally unique identifier for the CSIRT generating the document to deconflict private extensions used in the document. The fully qualified domain name associated with the CSIRT MUST be used as the identifier. See Section 5.3.

private-enum-id
Optional. STRING. An organizationally unique identifier for an extension used in the document. If this attribute is set, the private-enum-name MUST also be set. See Section 5.3.

3.2. Incident Class

The Incident class describes commonly exchanged information when reporting or sharing derived analysis from security incidents.

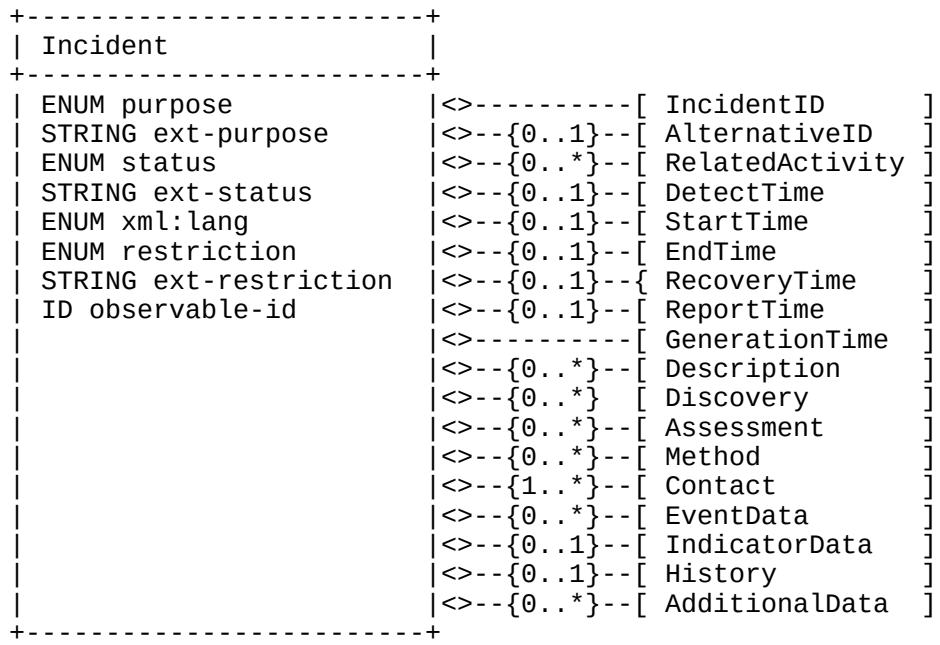

Figure 6: The Incident Class

The aggregate classes of the Incident class are:

IncidentID
: One. An incident tracking number assigned to this incident by the CSIRT that generated the IODEF document. See Section 3.4.

AlternativeID
: Zero or one. The incident tracking numbers used by other CSIRTs to refer to the incident described in the document. See Section 3.5.

RelatedActivity
: Zero or more. Related activity and attribution of this activity. See Section 3.6.

DetectTime
: Zero or one. DATETIME. The time the incident was first detected.

StartTime
: Zero or one. DATETIME. The time the incident started.

EndTime
: Zero or one. DATETIME. The time the incident ended.

RecoveryTime
Zero or one. DATETIME. The time the site recovered from the incident.

ReportTime
Zero or one. DATETIME. The time the incident was reported.

GenerationTime
One. DATETIME. The time the content in this Incident class was generated.

Description
Zero or more. ML_STRING. A free-form text description of the incident.

Discovery
Zero or more. The means by which this incident was detected. See Section 3.10.

Assessment
Zero or more. A characterization of the impact of the incident. See Section 3.12.

Method
Zero or more. The techniques used by the threat actor in the incident. See Section 3.11.

Contact
One or more. Contact information for the parties involved in the incident. See Section 3.9.

EventData
Zero or more. Description of the events comprising the incident. See Section 3.14.

IndicatorData
Zero or one. Indicators from the analysis of an incident. See Section 3.28.

History
Zero or one. A log of significant events or actions that occurred during the course of handling the incident. See Section 3.13.

AdditionalData
Zero or more. EXTENSION. Mechanism by which to extend the data model.

The attributes of the Incident class are:

purpose
Required. ENUM. The purpose attribute represents describes the rational for document the information in this class. It is closely related to the Expectation class (Section 3.15). These values are maintained in the "Incident-purpose" IANA registry per Section 10.2. This attribute is defined as an enumerated list:

1. traceback. The Incident was sent for trace-back purposes.

2. mitigation. The Incident was sent to request aid in mitigating the described activity.

3. reporting. The Incident was sent to comply with reporting requirements.

4. watch. The Incident was sent to convey indicators that should be monitored.

5. other. The Incident was sent for purposes specified in the Expectation class.

6. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

ext-purpose
Optional. STRING. A means by which to extend the purpose attribute. See Section 5.1.1.

status
Optional. ENUM. The status attribute conveys the state in a workflow where the incident is currently found. These values are maintained in the "Incident-status" IANA registry per Section 10.2. This attribute is defined as an enumerated list:

1. new. The Incident is newly reported and has not been actioned.

2. in-progress. The contents of this Incident are under investigation.

3. forwarded. The Incident has been forwarded to another party for handling.

4. resolved. The investigation into the activity in this Incident has concluded.

5. future. The described activity has not yet been detected.

6. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

ext-status
Optional. STRING. A means by which to extend the status attribute. See Section 5.1.1.

xml:lang
Optional. ENUM. A language identifier per Section 2.12 of [W3C.XML] whose values and form are described in [RFC5646]. The interpretation of this code is described in Section 6.

restriction
Optional. ENUM. See Section 3.3.1. The default value is "private".

ext-restriction
Optional. STRING. A means by which to extend the restriction attribute. See Section 5.1.1.

observable-id
Optional. ID. See Section 3.3.2.

3.3. Common Attributes

There are a number of recurring attributes used in the information model. They are documented in this section.

3.3.1. restriction Attribute

The restriction 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 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.

These values are maintained in the "Restriction" IANA registry per Section 10.2.

1. public. The information can be freely distributed without restriction.

2. partner. The information may be shared within a closed community of peers, partners, or affected parties, but cannot be openly published.

3. need-to-know. The information may be shared only within the organization with individuals that have a need to know.

4. private. The information may not be shared.

5. default. The information can be shared according to an information disclosure policy pre-arranged by the communicating parties.

6. white. Same as ’public’.

7. green. Same as ’partner’.

8. amber. Same as ’need-to-know’.

9. red. Same as ’private’.

10. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

3.3.2. observable-id Attribute

The observable-id attribute tags information in the document as an observable so that it can be referenced later in the description of an indicator. The value of this attribute is a unique identifier in the scope of the document. It is used by the ObservableReference class to enumerate observables when defining an indicator with the IndicatorData class.

3.4. IncidentID Class

The IncidentID class represents a tracking number that is unique in the context of the CSIRT. It serves as an identifier for an incident or a document identifier when sharing indicators. This identifier would serve as an index into a CSIRT's incident handling or knowledge management system.

The combination of the name attribute and the string in the element content MUST be a globally unique identifier describing the activity. Documents generated by a given CSIRT MUST NOT reuse the same value unless they are referencing the same incident.

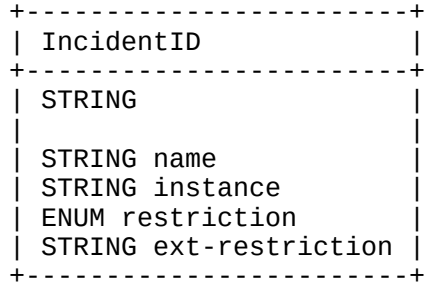

Figure 7: The IncidentID Class

The content of the class is an incident identifier of type STRING.

The attributes of the IncidentID class are:

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

instance
: Optional. STRING. An identifier referencing a subset of the named incident.

restriction
: Optional. ENUM. See Section 3.3.1.

ext-restriction
: Optional. STRING. A means by which to extend the restriction attribute. See Section 5.1.1.

3.5. AlternativeID Class

The AlternativeID class lists the tracking numbers used by CSIRTs, other than the one generating the document, to refer to the identical activity described in the IODEF document. A tracking number listed as an AlternativeID references the same incident detected by another CSIRT. The tracking numbers of the CSIRT that generated the IODEF document must never be considered an AlternativeID.

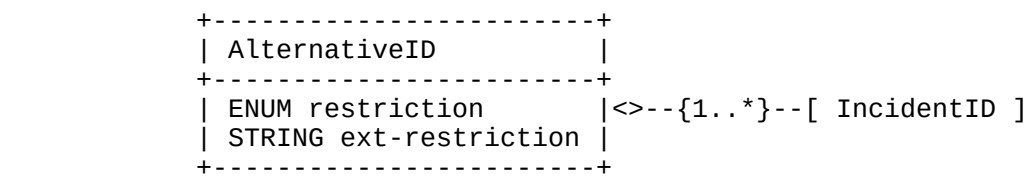

Figure 8: The AlternativeID Class

The aggregate class of the AlternativeID class is:

IncidentID
: One or more. The tracking number of another CSIRT. See Section 3.4.

The attributes of the AlternativeID class are:

restriction
: Optional. ENUM. See Section 3.3.1.

ext-restriction
: Optional. STRING. A means by which to extend the restriction attribute. See Section 5.1.1.

3.6. RelatedActivity Class

The RelatedActivity class relates the information described in the rest of the document to previously observed incidents or activity; and allows attribution to a specific actor or campaign.

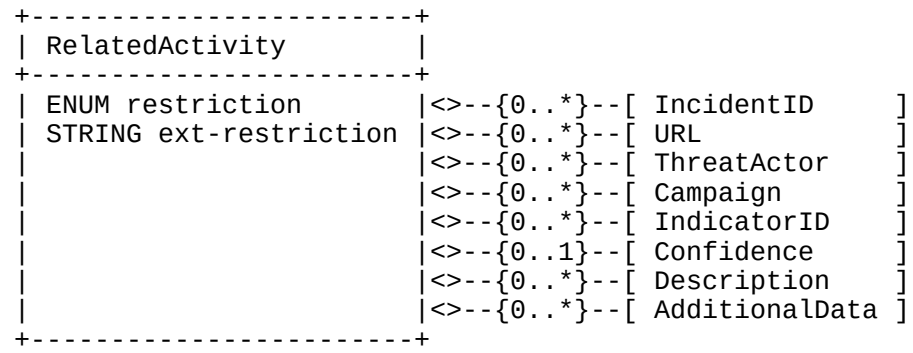

Figure 9: RelatedActivity Class

The aggregate classes of the RelatedActivity class are:

IncidentID
: Zero or more. The tracking number of a related incident. See Section 3.4.

URL
: Zero or more. URL. A URL to activity related to this incident.

ThreatActor
: Zero or more. The threat actor to whom the incident activity is attributed. See Section 3.7.

Campaign
: Zero or more. The campaign of a given threat actor to whom the described activity is attributed. See Section 3.8.

IndicatorID
: Zero or more. A reference to a related indicator. See Section 3.4.

Confidence
: Zero or one. An estimate of the confidence in attributing this RelatedActivity to the events described in the document. See Section 3.12.5.

Description
: Zero or more. ML_STRING. A description of how these relationships were derived.

AdditionalData
: Zero or more. EXTENSION. A mechanism by which to extend the data model.

The RelatedActivity class MUST have at least one instance of any of the following child classes: IncidentID, URL, ThreatActor, Campaign, Description or AdditionalData.

The attributes of the RelatedActivity class are:

restriction
: Optional. ENUM. See Section 3.3.1.

ext-restriction
: Optional. STRING. A means by which to extend the restriction attribute. See Section 5.1.1.

3.7. ThreatActor Class

The ThreatActor class describes a threat actor.

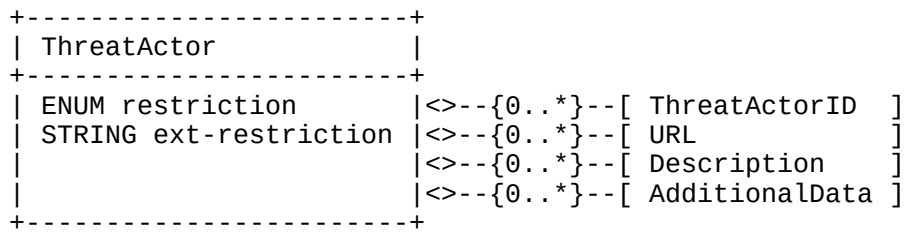

Figure 10: ThreatActor Class

The aggregate classes of the ThreatActor class are:

ThreatActorID
: Zero or more. STRING. An identifier for the threat actor.

URL
: Zero or more. URL. A URL to a reference describing the threat actor.

Description
: Zero or more. ML_STRING. A description of the threat actor.

AdditionalData
: Zero or more. EXTENSION. A mechanism by which to extend the data model.

The ThreatActor class MUST have at least one instance of a child class.

The attributes of the ThreatActor class are:

restriction
Optional. ENUM. See Section 3.3.1.

ext-restriction
Optional. STRING. A means by which to extend the restriction attribute. See Section 5.1.1.

3.8. Campaign Class

The Campaign class describes a campaign of attacks by a threat actor.

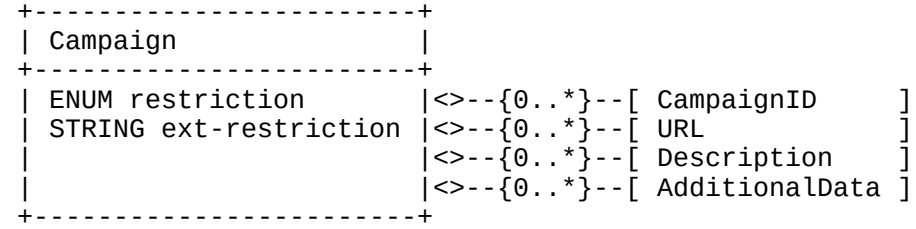

Figure 11: Campaign Class

The aggregate classes of the Campaign class are:

CampaignID
Zero or more. STRING. An identifier for the campaign.

URL
Zero or more. URL. A URL to a reference describing the campaign.

Description
Zero or more. ML_STRING. A description of the campaign.

AdditionalData
Zero or more. EXTENSION. A mechanism by which to extend the data model.

The Campaign class MUST have at least one instance of a child class.

The attributes of the Campaign class are:

restriction
Optional. ENUM. See Section 3.3.1.

ext-restriction
Optional. STRING. A means by which to extend the restriction attribute. See Section 5.1.1.

3.9. Contact Class

The Contact class describes contact information for organizations and personnel involved in the incident. This class allows for the naming of the involved party, specifying contact information for them, and identifying their role in the incident.

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

The recursive definition of Contact provides a way to relate information without requiring the explicit use of identifiers or duplication of data. A complete point of contact is derived by a particular traversal from the root Contact class to the leaf Contact class. Each child Contact class logically inherits contact information from its ancestors.

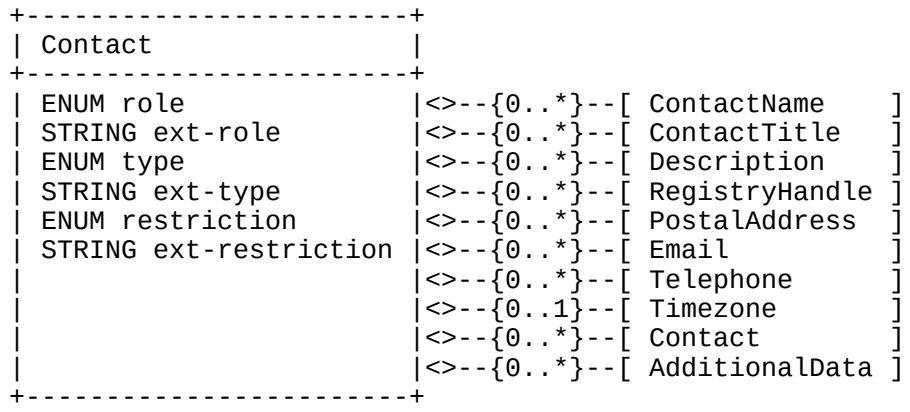

Figure 12: The Contact Class

The aggregate classes of the Contact class are:

ContactName
: Zero or more. ML_STRING. The name of the contact. The contact may either be an organization or a person. The type attribute disambiguates the semantics.

ContactTitle
: Zero or more. ML_STRING. The title for the individual named in the ContactName.

Description
Zero or more. ML_STRING. A free-form text description of the contact.

RegistryHandle
Zero or more. A handle name into the registry of the contact. See Section 3.9.1.

PostalAddress
Zero or more. The postal address of the contact. See Section 3.9.2.

Email
Zero or more. The email address of the contact. See Section 3.9.3.

Telephone
Zero or more. The telephone number of the contact. See Section 3.9.4.

Timezone
Zero or one. TIMEZONE. The timezone in which the contact resides.

Contact
Zero or more. A recursive definition of the Contact class. This definition can be used to group common data pertaining to multiple points of contact and is especially useful when listing multiple contacts at the same organization.

AdditionalData
Zero or more. EXTENSION. A mechanism by which to extend the data model.

At least one of the aggregate classes MUST be present in an instance of the Contact class.

The attributes of the Contact class are:

role
Required. ENUM. Indicates the role the contact fulfills. These values are maintained in the "Contact-role" IANA registry per Section 10.2.

1. creator. The entity that generate the document.

2. reporter. The entity that reported the information.

3. admin. An administrative contact or business owner for an asset or organization.

4. tech. An entity responsible for the day-to-day management of technical issues for an asset or organization.

5. provider. An external hosting provider for an asset.

6. user. An end-user of an asset or part of an organization.

7. billing. An entity responsible for billing issues for an asset or organization.

8. legal. An entity responsible for legal issue related to an asset or organization.

9. irt. An entity responsible for handling security issues for an asset or organization.

10. abuse. An entity responsible for handling abuse originating from an asset or organization.

11. cc. An entity that is to be kept informed about the events related to an asset or organization.

12. cc-irt. A CSIRT or information sharing organization coordinating activity related to an asset or organization.

13. leo. A law enforcement organization supporting the investigation of activity affecting an asset or organization.

14. vendor. The vendor that produces an asset.

15. vendor-support. A vendor that provides services.

16. victim. A victim in the incident.

17. victim-notified. A victim in the incident who has been notified.

18. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

ext-role
: Optional. STRING. A means by which to extend the role attribute. See Section 5.1.1.

type
Required. ENUM. Indicates the type of contact being described. This attribute is defined as an enumerated list. These values are maintained in the "Contact-type" IANA registry per Section 10.2.

1. person. The information for this contact references an individual.

2. organization. The information for this contact references an organization.

3. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

ext-type
Optional. STRING. A means by which to extend the type attribute. See Section 5.1.1.

restriction
Optional. ENUM. See Section 3.3.1.

ext-restriction
Optional. STRING. A means by which to extend the restriction attribute. See Section 5.1.1.

3.9.1. RegistryHandle Class

The RegistryHandle class represents a handle into an Internet registry or community-specific database.

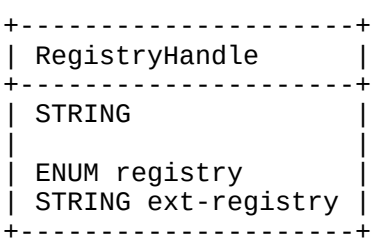

Figure 13: The RegistryHandle Class

The content of the class is a handle into a registry of type STRING.

The attributes of the RegistryHandle class are:

registry

Required. ENUM. The database to which the handle belongs. These values are maintained in the "RegistryHandle-registry" IANA registry per Section 10.2. The possible values are:

1. internic. Internet Network Information Center

2. apnic. Asia Pacific Network Information Center

3. arin. American Registry for Internet Numbers

4. lacnic. Latin-American and Caribbean IP Address Registry

5. ripe. Reseaux IP Europeens

6. afrinic. African Internet Numbers Registry

7. local. A database local to the CSIRT

8. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

ext-registry
Optional. STRING. A means by which to extend the registry attribute. See Section 5.1.1.

3.9.2. PostalAddress Class

The PostalAddress class specifies an postal address and associated annotation.

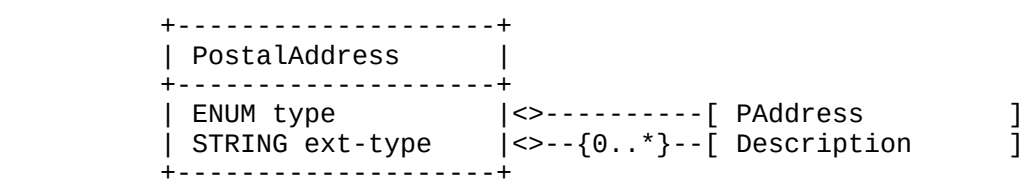

Figure 14: The PostalAddress Class

The aggregate classes of the PostalAddress class are:

PAddress
One. POSTAL. A postal address.

Description
Zero or more. ML_STRING. A free-form text description of the address.

The attributes of the PostalAddress class are:

type
: Optional. ENUM. Categorizes the type of address described in the PAddress class. These values are maintained in the "PostalAddress-type" IANA registry per Section 10.2.

  1. street. An address describing a physical location.

  2. mailing. An address to which correspondence should be sent.

  3. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

ext-type
: Optional. STRING. A means by which to extend the type attribute. See Section 5.1.1.

3.9.3. Email Class

The Email class specifies an email address and associated annotation.

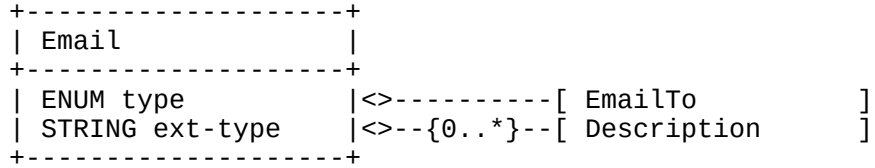

Figure 15: The Email Class

The aggregate classes of the Email class are:

EmailTo
: One. EMAIL. An email address.

Description
: Zero or more. ML_STRING. A free-form text description of the email address.

The attributes of the Email class are:

type
: Optional. ENUM. Categorizes the type of email address described in the EmailTo class. These values are maintained in the "Email-type" IANA registry per Section 10.2.

1. direct. A email address of an individual.

2. hotline. A email address regularly monitored for operational purposes.

3. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

ext-type
Optional. STRING. A means by which to extend the type attribute. See Section 5.1.1.

3.9.4. Telephone Class

The Telephone class describes a telephone number and associated annotation.

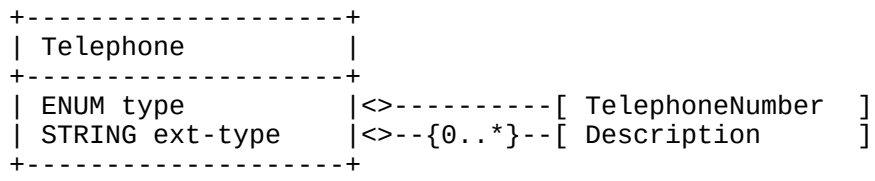

Figure 16: The Telephone Class

The aggregate classes of the Telephone class are:

TelephoneNumber
One. PHONE. A telephone number.

Description
Zero or more. ML_STRING. A free-form text description of the phone number.

The attributes of the Telephone class are:

type
Optional. ENUM. Categorizes the type of telephone number described in the TelephoneNumber class. These values are maintained in the "Telephone-type" IANA registry per Section 10.2.

1. wired. A number of a wire-line (land-line) phone.

2. mobile. A number of a mobile phone.

3. fax. A number to a fax machine.

4. hotline. A number to a regularly monitored operational hotline.

5. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

ext-type
Optional. STRING. A means by which to extend the type attribute. See Section 5.1.1.

3.10. Discovery Class

The Discovery class describes how an incident was detected.

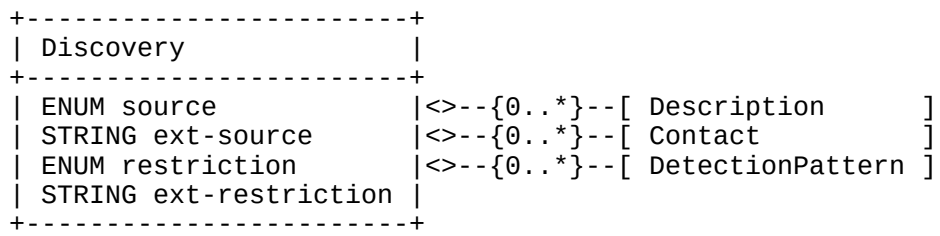

Figure 17: The Discovery Class

The aggregate classes of the Discovery class are:

Description
Zero or more. ML_STRING. A free-form text description of how this incident was detected.

Contact
Zero or more. Contact information for the party that discovered the incident. See Section 3.9.

DetectionPattern
Zero or more. Describes an application-specific configuration that detected the incident. See Section 3.10.1.

The attributes of the Discovery class are:

source
Optional. ENUM. Categorizes the techniques used to discover the incident. These values are partially derived from Table 3-1 of [NIST800.61rev2]. These values are maintained in the "Discovery-source" IANA registry per Section 10.2.

1. nidps. Network Intrusion Detection or Prevention system.

2. hips. Host-based Intrusion Prevention system.

3. siem. Security Information and Event Management System.

4. av. Antivirus or and antispam software.

5. third-party-monitoring. Contracted third-party monitoring service.

6. incident. The activity was discovered while investigating an unrelated incident.

7. os-log. Operating system logs.

8. application-log. Application logs.

9. device-log. Network device logs.

10. network-flow. Network flow analysis.

11. passive-dns. Passive DNS analysis.

12. investigation. Manual investigation initiated based on notification of a new vulnerability or exploit.

13. audit. Security audit.

14. internal-notification. A party within the organization reported the activity

15. external-notification. A party outside of the organization reported the activity.

16. leo. A law enforcement organization notified the victim organization.

17. partner. A customer or business partner reported the activity to the victim organization.

18. actor. The threat actor directly or indirectly reported this activity to the victim organization.

19. unknown. Unknown detection approach.

20. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

ext-source
Optional. STRING. A means by which to extend the source attribute. See Section 5.1.1.

restriction
Optional. ENUM. See Section 3.3.1.

ext-restriction
Optional. STRING. A means by which to extend the restriction attribute. See Section 5.1.1.

3.10.1. DetectionPattern Class

The DetectionPattern class describes a configuration or signature that can be used by an IDS/IPS, SIEM, anti-virus, end-point protection, network analysis, malware analysis, or host forensics tool to identify a particular phenomenon. This class requires the identification of the target application and allows the configuration to be described in either free-form or machine readable form.

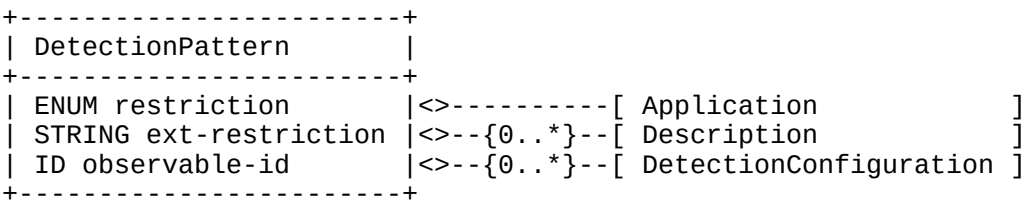

Figure 18: The DetectionPattern Class

The aggregate classes of the DetectionPattern class are:

Application
One. SOFTWARE. The application for which the DetectionConfiguration or Description is being provided.

Description
Zero or more. ML_STRING. A free-form text description of how to use the Application or provided DetectionConfiguration.

DetectionConfiguration
Zero or more. STRING. A machine consumable configuration to find a pattern of activity.

Either an instance of the Description or DetectionConfiguration class MUST be present.

The attributes of the DetectionPattern class are:

restriction
Optional. ENUM. See Section 3.3.1.

ext-restriction
Optional. STRING. A means by which to extend the restriction attribute. See Section 5.1.1.

observable-id
Optional. ID. See Section 3.3.2.

3.11. Method Class

The Method class describes the tactics, techniques, procedures or weakness used by the threat actor in an incident. This class consists of both a list of references describing the attack methods and weaknesses and a free-form text description.

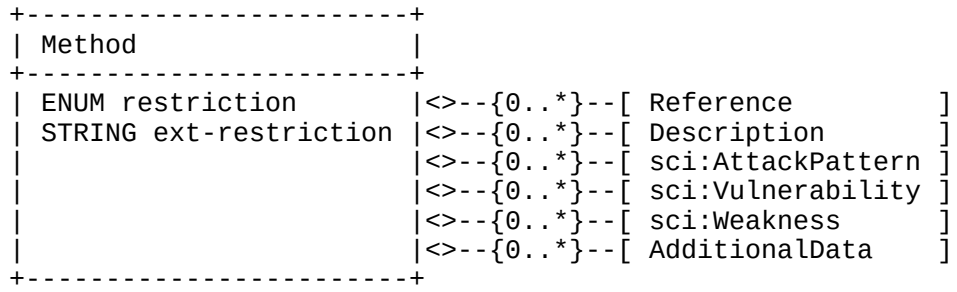

Figure 19: The Method Class

The aggregate classes of the Method class are:

Reference
Zero or more. A reference to a vulnerability, malware sample, advisory, or analysis of an attack technique. See Section 3.11.1.

Description
Zero or more. ML_STRING. A free-form text description of techniques, tactics, or procedures used by the threat actor.

sci:AttackPattern
Zero or more. A reference to an pattern of attack or exploitation per [RFC7203]

sci:Vulnerability
Zero or more. A reference to a vulnerability per [RFC7203]

sci:Weakness
Zero or more. A reference to the exploited weakness per [RFC7203]

AdditionalData
Zero or more. EXTENSION. A mechanism by which to extend the data model.

An instance of one of these child MUST be present.

The attributes of the Method class are:

restriction
Optional. ENUM. See Section 3.3.1.

ext-restriction
Optional. STRING. A means by which to extend the restriction attribute. See Section 5.1.1.

3.11.1. Reference Class

The Reference class is an external reference to relevant information such a vulnerability, IDS alert, malware sample, advisory, or attack technique.

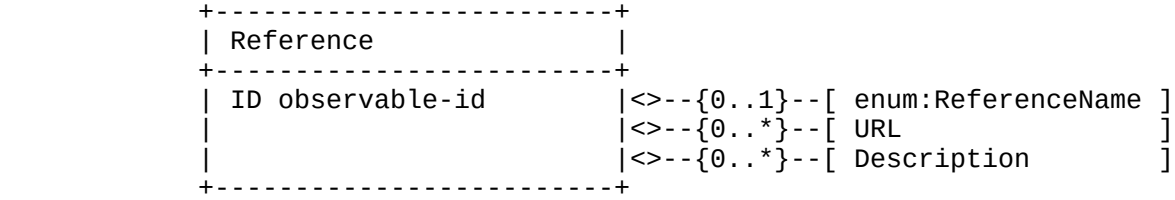

Figure 20: The Reference Class

The aggregate classes of the Reference class are:

enum:ReferenceName
Zero or one. Reference identifier per [RFC7495].

URL
Zero or more. URL. A URL to a reference.

Description
Zero or more. ML_STRING. A free-form text description of this reference.

At least one of these classes MUST be present.

The attribute of the Reference class is:

observable-id
Optional. ID. See Section 3.3.2.

3.12. Assessment Class

The Assessment class describes the repercussions of the incident to the victim.

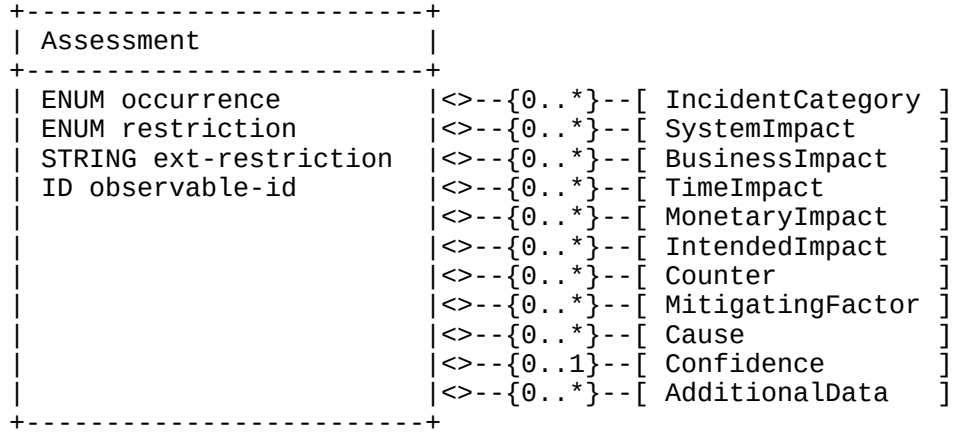

Figure 21: Assessment Class

The aggregate classes of the Assessment class are:

IncidentCategory
Zero or more. ML_STRING. A free-form text description categorizing the type of Incident.

SystemImpact
Zero or more. A technical characterization of the impact of the incident activity on the victim's enterprise. See Section 3.12.1.

BusinessImpact
Zero or more. Impact of the incident activity on the business functions of the victim organization. See Section 3.12.2.

TimeImpact
Zero or more. A characterization of the victim organization due to the incident activity as a function of time. See Section 3.12.3.

MonetaryImpact
Zero or more. The financial loss due to the incident activity. See Section 3.12.4.

IntendedImpact
Zero or more. The intended outcome to the victim sought by the threat actor. Defined identically to the BusinessImpact defined in Section 3.12.2, but describes intent rather than the realized impact.

Counter
Zero or more. A counter with which to summarize the magnitude of the activity. See Section 3.18.3.

MitigatingFactor
Zero or more. ML_STRING. A description of a mitigating factor relative to the impact on the victim organization.

Cause
Zero or more. ML_STRING. A description of an underlying cause of the impact.

Confidence
Zero or one. An estimate of confidence in the impact assessment. See Section 3.12.5.

AdditionalData
Zero or more. EXTENSION. A mechanism by which to extend the data model.

A least one instance of the possible five impact classes (i.e., SystemImpact, BusinessImpact, TimeImpact, MonetaryImpact or IntendedImpact) MUST be present.

The attributes of the Assessment class are:

occurrence
Optional. ENUM. Specifies whether the assessment is describing actual or potential outcomes.

1. actual. This assessment describes activity that has occurred.

2. potential. This assessment describes potential activity that might occur.

restriction
Optional. ENUM. See Section 3.3.1.

ext-restriction
Optional. STRING. A means by which to extend the restriction attribute. See Section 5.1.1.

observable-id
Optional. ID. See Section 3.3.2.

3.12.1. SystemImpact Class

The SystemImpact class describes the technical impact of the incident to the systems on the network.

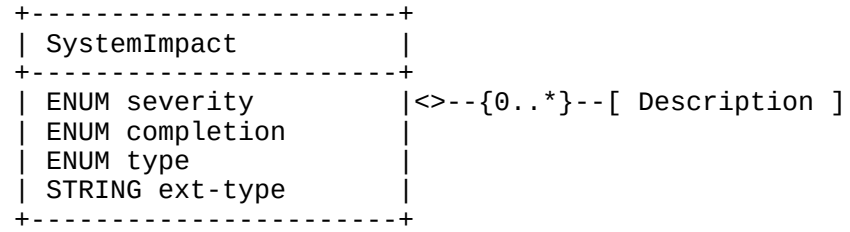

Figure 22: SystemImpact Class

The aggregate class of the SystemImpact class is:

Description
Zero or more. ML_STRING. A free-form text description of the impact to the system.

The attributes of the SystemImpact class are:

severity
Optional. ENUM. An estimate of the relative severity of the activity. The permitted values are shown below. There is no default value.

1. low. Low severity

2. medium. Medium severity

3. high. High severity

completion
Optional. ENUM. An indication whether the described activity was successful. The permitted values are shown below. There is no default value.

1. failed. The attempted activity was not successful.

2. succeeded. The attempted activity succeeded.

type
Required. ENUM. Classifies the impact. The permitted values are shown below. The default value is "unknown". These values are maintained in the "SystemImpact-type" IANA registry per Section 10.2.

1. takeover-account. Control was taken of a given account.

2. takeover-service. Control was taken of a given service.

3. takeover-system. Control was taken of a given system.

4. cps-manipulation. A cyber-physical system was manipulated.

5. cps-damage. A cyber-physical system was damaged.

6. availability-data. Access to particular data was degraded or denied.

7. availability-account. Access to an account was degraded or denied.

8. availability-service. Access to a service was degraded or denied.

9. availability-system. Access to a system was degraded or denied.

10. damaged-system. Hardware on a system was irreparably damaged.

11. damaged-data. Data on a system was deleted.

12. breach-proprietary. Sensitive or proprietary information was accessed or exfiltrated.

13. breach-privacy. Personally identifiable information was accessed or exfiltrated.

14. breach-credential. Credential information was accessed or exfiltrated.

15. breach-configuration. System configuration or data inventory was access or exfiltrated.

16. integrity-data. Data on the system was modified.

17. integrity-configuration. Application or system configuration was modified.

18. integrity-hardware. Firmware of a hardware component was modified.

19. traffic-redirection. Network traffic on the system was redirected

20. monitoring-traffic. Network traffic emerging from a host or enclave was monitored.

21. monitoring-host. System activity (e.g., running processes, keystrokes) were monitored.

22. policy. Activity violated the system owner's acceptable use policy.

23. unknown. The impact is unknown.

24. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

ext-type
Optional. STRING. A means by which to extend the type attribute. See Section 5.1.1.

3.12.2. BusinessImpact Class

The BusinessImpact class describes and characterizes the degree to which the function of the organization was impacted by the Incident.

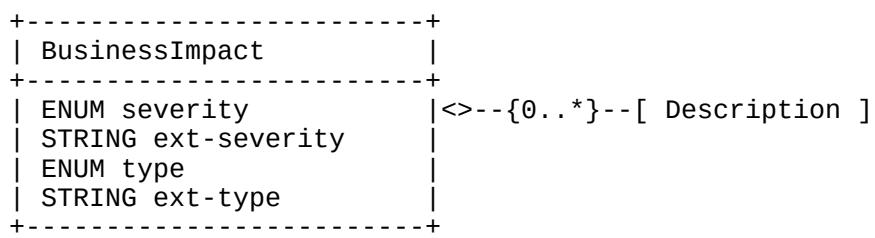

Figure 23: BusinessImpact Class

The aggregate class of the BusinessImpact class is:

Description
Zero or more. ML_STRING. A free-form text description of the impact to the organization.

The attributes of the BusinessImpact class are:

severity
Optional. ENUM. Characterizes the severity of the incident on business functions. The permitted values are shown below. They were derived from Table 3-2 of [NIST800.61rev2]. The default value is "unknown". These values are maintained in the "BusinessImpact-severity" IANA registry per Section 10.2.

1. none. No effect to the organization’s ability to provide all services to all users.

2. low. Minimal effect as the organization can still provide all critical services to all users but has lost efficiency.

3. medium. The organization has lost the ability to provide a critical service to a subset of system users.

4. high. The organization is no longer able to provide some critical services to any users.

5. unknown. The impact is not known.

6. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

ext-severity
Optional. STRING. A means by which to extend the severity attribute. See Section 5.1.1.

type
Required. ENUM. Characterizes the effect this incident had on the business. The permitted values are shown below. The default value is "unknown". These values are maintained in the "BusinessImpact-type" IANA registry per Section 10.2.

1. breach-proprietary. Sensitive or proprietary information was accessed or exfiltrated.

2. breach-privacy. Personally identifiable information was accessed or exfiltrated.

3. breach-credential. Credential information was accessed or exfiltrated.

4. loss-of-integrity. Sensitive or proprietary information was changed or deleted.

5. loss-of-service. Service delivery was disrupted.

6. theft-financial. Money was stolen.

7. theft-service. Services were misappropriated.

8. degraded-reputation. The reputation of the organization's brand was diminished.

9. asset-damage. A cyber-physical system was damaged.

10. asset-manipulation. A cyber-physical system was manipulated.

11. legal. The incident resulted in legal or regulatory action.

12. extortion. The incident resulted in actors extorting the victim organization.

13. unknown. The impact is unknown.

14. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

ext-type
Optional. STRING. A means by which to extend the type attribute. See Section 5.1.1.

3.12.3. TimeImpact Class

The TimeImpact class describes the impact of the incident on an organization as a function of time. It provides a way to convey down time and recovery time.

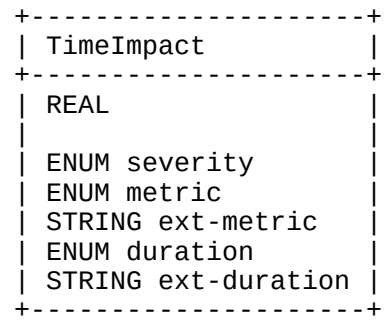

Figure 24: TimeImpact Class

The content of the class is of type REAL and specifies an amount of time. The duration attribute provides units for this content; and the metric attribute explains what this content is measuring.

The attributes of the TimeImpact class are:

severity
: Optional. ENUM. An estimate of the relative severity of the activity. The permitted values are shown below. There is no default value.

  1. low. Low severity

  2. medium. Medium severity

  3. high. High severity

metric
: Required. ENUM. Defines the meaning of the value in the element content. These values are maintained in the "TimeImpact-metric" IANA registry per Section 10.2.

  1. labor. Total staff-time to recovery from the activity (e.g., 2 employees working 4 hours each would be 8 hours).

  2. elapsed. Elapsed time from the beginning of the recovery to its completion (i.e., wall-clock time).

  3. downtime. Duration of time for which some provided service(s) was not available.

  4. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

ext-metric
Optional. STRING. A means by which to extend the metric attribute. See Section 5.1.1.

duration
Optional. ENUM. Defines the unit of time for the value in the element content. The default value is "hour". These values are maintained in the "TimeImpact-duration" IANA registry per Section 10.2.

1. second. The unit of the element content is seconds.

2. minute. The unit of the element content is minutes.

3. hour. The unit of the element content is hours.

4. day. The unit of the element content is days.

5. month. The unit of the element content is months.

6. quarter. The unit of the element content is quarters.

7. year. The unit of the element content is years.

8. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

ext-duration
Optional. STRING. A means by which to extend the duration attribute. See Section 5.1.1.

3.12.4. MonetaryImpact Class

The MonetaryImpact class describes the financial impact of the activity on an organization. For example, this impact may consider losses due to the cost of the investigation or recovery, diminished productivity of the staff, or a tarnished reputation that will affect future opportunities.

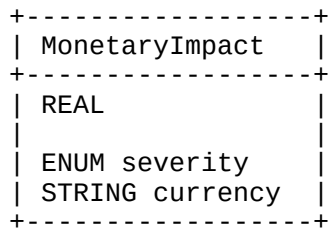

Figure 25: MonetaryImpact Class

The content of the class is of type REAL and specifies a quantity of money. The currency attribute defines the currently of this value.

The attributes of the MonetaryImpact class are:

severity
: Optional. ENUM. An estimate of the relative severity of the activity. The permitted values are shown below. There is no default value.

  1. low. Low severity

  2. medium. Medium severity

  3. high. High severity

currency
: Optional. STRING. Defines the currency in which the value in the element content is expressed. The permitted values are defined in "Codes for the representation of currencies and funds" of [ISO4217]. There is no default value.

3.12.5. Confidence Class

The Confidence class represents an estimate of the validity and accuracy of data expressed in the document. This estimate can be expressed as a category or a numeric calculation.

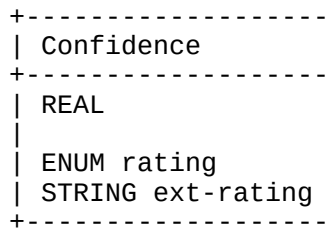

Figure 26: Confidence Class

The content of the class is of type REAL and specifies a numerical assessment in the confidence of the data when the value of the rating attribute is "numeric". Otherwise, this element MUST be empty.

The attributes of the Confidence class are:

rating
: Required. ENUM. A qualitative assessment of confidence. These values are maintained in the "Confidence-rating" IANA registry per Section 10.2

1. low. Low confidence.

2. medium. Medium confidence.

3. high. High confidence.

4. numeric. The element content contains a number that conveys the confidence of the data. The semantics of this number outside the scope of this specification.

5. unknown. The confidence rating value is not known.

6. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

ext-rating
: Optional. STRING. A means by which to extend the rating attribute. See Section 5.1.1.

3.13. History Class

The History class is a log of the significant events or actions performed by the involved parties during the course of handling the incident.

The level of detail maintained in this log is left up to the discretion of those handling the incident.

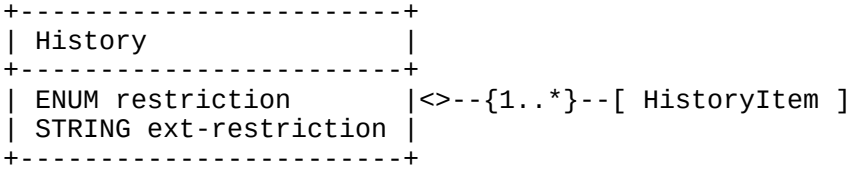

Figure 27: The History Class

The aggregate classes of the History class are:

HistoryItem
: One or more. An entry in the history log of significant events or actions performed by the involved parties. See Section 3.13.1.

The attributes of the History class are:

restriction
: Optional. ENUM. See Section 3.3.1.

ext-restriction
: Optional. STRING. A means by which to extend the restriction attribute. See Section 5.1.1.

3.13.1. HistoryItem Class

The HistoryItem class is an entry in the History (Section 3.13) log that documents a particular action or event that occurred in the course of handling the incident. The details of the entry are a free-form text description, but each can be categorized with the type attribute.

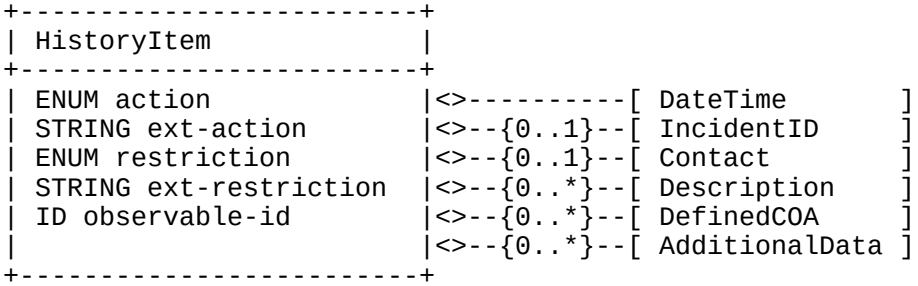

Figure 28: HistoryItem Class

The aggregate classes of the HistoryItem class are:

DateTime
One. DATETIME. A timestamp of this entry in the history log.

IncidentID
Zero or One. In a history log created by multiple parties, the IncidentID provides a mechanism to specify which CSIRT created a particular entry and references this organization's tracking number. When a single organization is maintaining the log, this class can be ignored. See Section 3.4.

Contact
Zero or One. Provides contact information for the entity that performed the action documented in this class. See Section 3.9.

Description
Zero or more. ML_STRING. A free-form text description of the action or event.

DefinedCOA
Zero or more. STRING. An identifier meaningful to the sender and recipient of this document that references a course of action (COA). This class MUST be present if the action attribute is set to "defined-coa".

AdditionalData
Zero or more. EXTENSION. A mechanism by which to extend the data model.

The attributes of the HistoryItem class are:

action
Required. ENUM. Classifies a performed action or occurrence documented in this history log entry. As activity will likely have been instigated either through a previously conveyed expectation or internal investigation. This attribute is identical to the action attribute of the Expectation class. The difference is only one of tense. When an action is in this class, it has been completed. See Section 3.15.

ext-action
Optional. STRING. A means by which to extend the action attribute. See Section 5.1.1.

restriction
Optional. ENUM. See Section 3.3.1.

ext-restriction
Optional. STRING. A means by which to extend the restriction attribute. See Section 5.1.1.

observable-id
Optional. ID. See Section 3.3.2.

3.14. EventData Class

The EventData class is a container class to organize data about events that occurred during an incident.

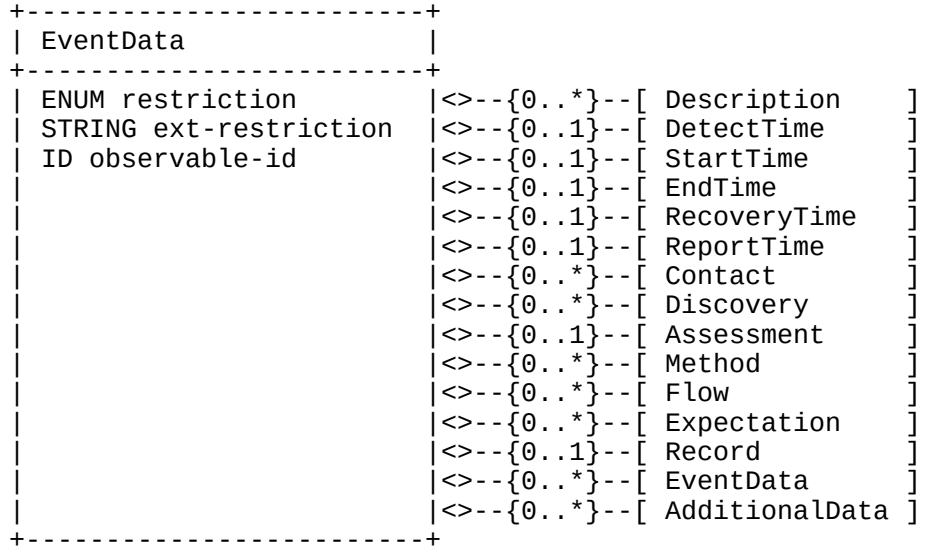

Figure 29: The EventData Class

The aggregate classes of the EventData class are:

Description
Zero or more. ML_STRING. A free-form text description of the event.

DetectTime
Zero or one. DATETIME. The time the event was detected.

StartTime
Zero or one. DATETIME. The time the event started.

EndTime
Zero or one. DATETIME. The time the event ended.

RecoveryTime
Zero or one. DATETIME. The time the site recovered from the event.

ReportTime
Zero or one. DATETIME. The time the event was reported.

Contact
Zero or more. Contact information for the parties involved in the event. See Section 3.9.

Discovery
Zero or more. The means by which the event was detected. See Section 3.10.

Assessment
Zero or one. The impact of the event on the victim and the actions taken. See Section 3.12.

Method
Zero or more. The technique used by the threat actor in the event. See Section 3.11.

Flow
Zero or more. A description of the systems or networks involved. See Section 3.16.

Expectation
Zero or more. The expected action to be performed by the recipient for the described event. See Section 3.15.

Record
Zero or one. Supportive data (e.g., log files) that provides additional information about the event. See Section 3.22.

EventData
Zero or more. A recursive definition of the EventData class. See Section 3.14.2 for an explanation on using this class.

AdditionalData
Zero or more. EXTENSION. An extension mechanism for data not explicitly represented in the data model.

At least one of the aggregate classes MUST be present in an instance of the EventData class.

The attributes of the EventData class are:

restriction
Optional. ENUM. See Section 3.3.1. The default value is "default".

ext-restriction
Optional. STRING. A means by which to extend the restriction attribute. See Section 5.1.1.

observable-id
Optional. ID. See Section 3.3.2.

3.14.1. Relating the Incident and EventData Classes

There is substantial overlap in the child classes aggregated in the Incident and EventData classes. Nevertheless, the semantics of these classes are quite different. The Incident class provides summary information about the entire incident, while the EventData class provides information about the individual events comprising the incident. In the common case, the EventData class will provide more specific information for the general description provided in the Incident class. However, in the case where the summarized information in the Incident class conflicts the detailed information in an EventData class the more specific EventData class MUST supersede the more generic information provided in Incident class.

3.14.2. Recursive Definition of EventData

The EventData class is container for the properties of an event in an incident. These properties include: the hosts involved, impact of the incident activity on the hosts, forensic logs, etc. The recursive definition of EventData allows for the grouping of related information with common properties. This approach eliminates the need for explicit identifiers to relate information or duplicate it. Instead, the relative depth (nesting) of a class is used to group (relate) information.

For example, consider a case where two hosts experience different impacts during an incident. However, these two hosts have common contact information. A depiction of how this situation would be represented can be found in Figure 30. EventData (2) and (3) group each of the two hosts with their unique impact. EventData (1) describes the common Contact class these two hosts share.

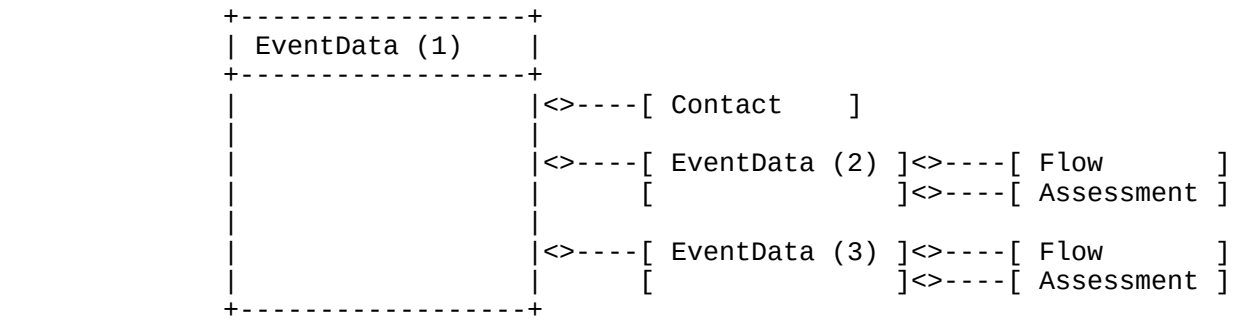

Figure 30: Recursion in the EventData Class

3.15. Expectation Class

The Expectation class conveys to the recipient of the IODEF document the actions the sender is requesting.

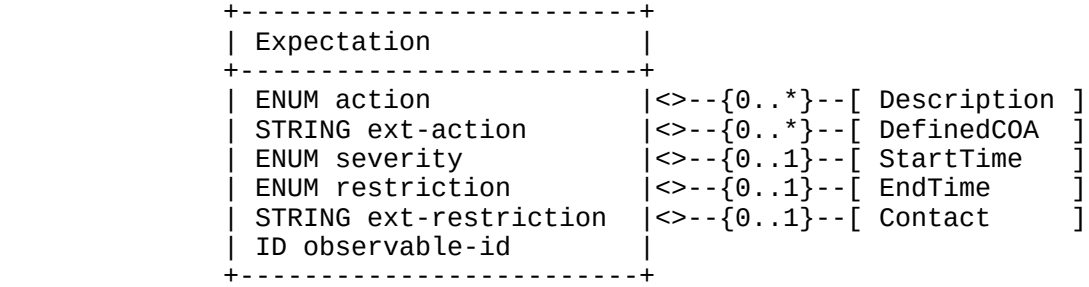

Figure 31: The Expectation Class

The aggregate classes of the Expectation class are:

Description
: Zero or more. ML_STRING. A free-form text description of the desired action(s).

DefinedCOA
: Zero or more. STRING. A unique identifier meaningful to the sender and recipient of this document that references a course of action. This class MUST be present if the action attribute is set to "defined-coa".

StartTime
: Zero or one. DATETIME. The time at which the sender would like the action performed. A timestamp that is earlier than the ReportTime specified in the Incident class denotes that the sender

would like the action performed as soon as possible. The absence of this element indicates no expectations of when the recipient would like the action performed.

EndTime
Zero or one. DATETIME. The time by which the sender expects the recipient to complete the action. If the recipient cannot complete the action before EndTime, the recipient MUST NOT carry out the action. Because of transit delays and clock drift the sender MUST be prepared for the recipient to have carried out the action, even if it completes past EndTime.

Contact
Zero or one. The entity expected to perform the action. See Section 3.9.

The attributes of the Expectation class are:

action
Optional. ENUM. Classifies the type of action requested. The default value of "other". These values are maintained in the "Expectation-action" IANA registry per Section 10.2.

1. nothing. No action is requested. Do nothing with the information.

2. contact-source-site. Contact the site(s) identified as the source of the activity.

3. contact-target-site. Contact the site(s) identified as the target of the activity.

4. contact-sender. Contact the originator of the document.

5. investigate. Investigate the systems(s) listed in the event.

6. block-host. Block traffic from the machine(s) listed as sources the event.

7. block-network. Block traffic from the network(s) lists as sources in the event.

8. block-port. Block the port listed as sources in the event.

9. rate-limit-host. Rate-limit the traffic from the machine(s) listed as sources in the event.

10. rate-limit-network. Rate-limit the traffic from the network(s) lists as sources in the event.

11. rate-limit-port. Rate-limit the port(s) listed as sources in the event.

12. redirect-traffic. Redirect traffic from the intended recipient for further analysis.

13. honeypot. Redirect traffic from systems listed in the event to a honeypot for further analysis.

14. upgrade-software. Upgrade or patch the software or firmware on an asset listed in the event.

15. rebuild-asset. Reinstall the operating system or applications on an asset listed in the event.

16. harden-asset. Change the configuration an asset listed in the event to reduce the attack surface.

17. remediate-other. Remediate the activity in a way other than by rate limiting or blocking.

18. status-triage. Confirm receipt and begin triaging the incident.

19. status-new-info. Notify the sender when new information is received for this incident.

20. watch-and-report. Watch for the described activity or indicators; and notify the sender when seen.

21. training. Train user to identify or mitigate the described threat.

22. defined-coa. Perform a predefined course of action (COA). The COA is named in the DefinedCOA class.

23. other. Perform a custom action described in the Description class.

24. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

ext-action

Optional. STRING. A means by which to extend the action attribute. See Section 5.1.1.

severity
Optional. ENUM. Indicates the desired priority of the action. This attribute is an enumerated list with no default value, and the semantics of these relative measures are context dependent.

1. low. Low priority

2. medium. Medium priority

3. high. High priority

restriction
Optional. ENUM. See Section 3.3.1. The default value is "default".

ext-restriction
Optional. STRING. A means by which to extend the restriction attribute. See Section 5.1.1.

observable-id
Optional. ID. See Section 3.3.2.

3.16. Flow Class

The Flow class describes the systems and networks involved in the incident; and the relationships between them.

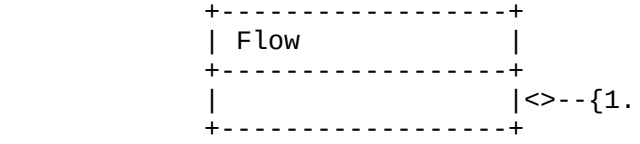

Figure 32: The Flow Class

The aggregate class of the Flow class is:

System
One or More. A host or network involved in an event. See Section 3.17.

The Flow class has no attributes.

3.17. System Class

The System class describes a system or network involved in an event.

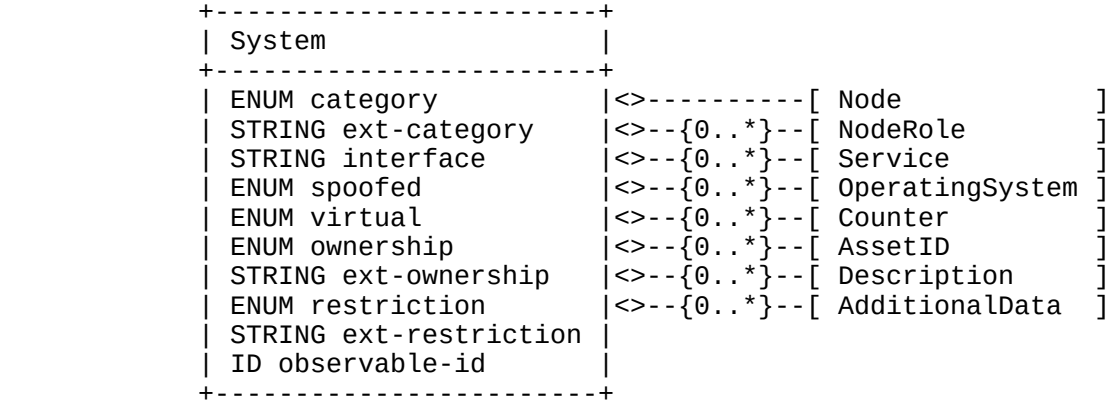

Figure 33: The System Class

The aggregate classes of the System class are:

Node
: One. A host or network involved in the incident. See Section 3.18.

NodeRole
: Zero or more. The intended purpose of the system. See Section 3.18.2.

Service
: Zero or more. A network service running on the system. See Section 3.20.

OperatingSystem
: Zero or more. SOFTWARE. The operating system running on the system.

Counter
: Zero or more. A counter with which to summarize properties of this host or network. See Section 3.18.3.

AssetID
: Zero or more. STRING. An asset identifier for the System.

Description

Zero or more. ML_STRING. A free-form text description of the System.

AdditionalData
Zero or more. EXTENSION. A mechanism by which to extend the data model.

The attributes of the System class are:

category
Optional. ENUM. Classifies the role the host or network played in the incident. These values are maintained in the "System-category" IANA registry per Section 10.2.

1. source. The System was the source of the event.

2. target. The System was the target of the event.

3. intermediate. The System was an intermediary in the event.

4. sensor. The System was a sensor monitoring the event.

5. infrastructure. The System was an infrastructure node of IODEF document exchange.

6. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

ext-category
Optional. STRING. A means by which to extend the category attribute. See Section 5.1.1.

interface
Optional. STRING. Specifies the interface on which the event(s) on this System originated. If the Node class specifies a network rather than a host, this attribute has no meaning.

spoofed
Optional. ENUM. An indication of confidence in whether this System was the true target or attacking host. The permitted values for this attribute are shown below. The default value is "unknown".

1. unknown. The accuracy of the category attribute value is unknown.

2. yes. The category attribute value is likely incorrect. In the case of a source, the System is likely a decoy; with a target, the System was likely not the intended victim.

3. no. The category attribute value is believed to be correct.

virtual
Optional. ENUM. Indicates whether this System is a virtual or physical device. The default value is "unknown".

1. yes. The System is a virtual device.

2. no. The System is a physical device.

3. unknown. It is not known if the System is virtual.

ownership
Optional. ENUM. Describes the ownership of this System relative to the victim in the incident. These values are maintained in the "System-ownership" IANA registry per Section 10.2.

1. organization. Corporate or enterprise-owned.

2. personal. Personally-owned by an employee or affiliate of the corporation or enterprise.

3. partner. Owned by a partner of the corporation or enterprise.

4. customer. Owned by a customer of the corporation or enterprise.

5. no-relationship. Owned by an entity that has no known relationship with victim organization.

6. unknown. Ownership is unknown.

7. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

ext-ownership
Optional. STRING. A means by which to extend the ownership attribute. See Section 5.1.1.

restriction
Optional. ENUM. See Section 3.3.1.

ext-restriction

Optional. STRING. A means by which to extend the restriction attribute. See Section 5.1.1.

observable-id
Optional. ID. See Section 3.3.2.

3.18. Node Class

The Node class identifies a system, asset or network; and its location.

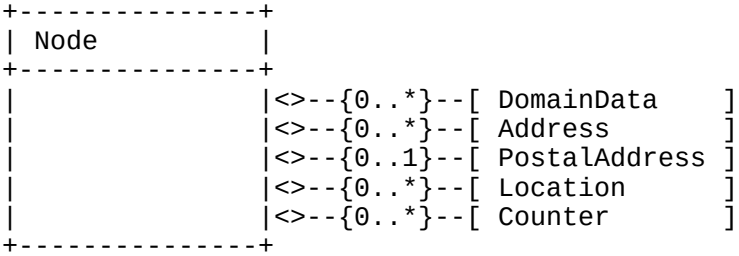

Figure 34: The Node Class

The aggregate classes of the Node class are:

DomainData
Zero or more. The domain (DNS) information associated with this Node. If an Address is not provided, at least one DomainData MUST be specified. See Section 3.19.

Address
Zero or more. The hardware, network, or application address of the Node. If a DomainData is not provided, at least one Address MUST be specified. See Section 3.18.1.

PostalAddress
Zero or one. POSTAL. The postal address of the node.

Location
Zero or more. ML_STRING. A free-form text description of the physical location of the Node. This description may provide a more detailed description of where in the PostalAddress this Node is found (e.g., room number, rack number, slot number in a chassis).

Counter
Zero or more. A counter with which to summarizes properties of this host or network. See Section 3.18.3.

The Node class has no attributes.

3.18.1. Address Class

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

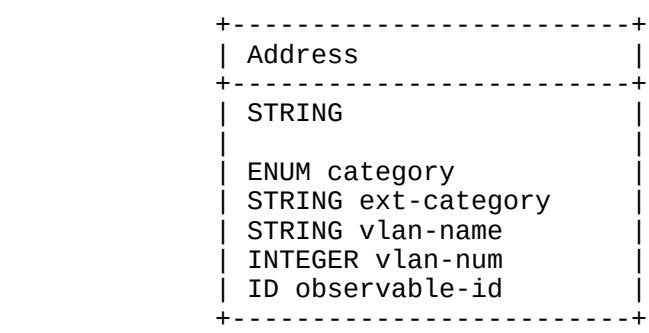

Figure 35: The Address Class

The content of the class is an address of type STRING whose semantics are determined by the category attribute.

The attributes of the Address class are:

category
: Required. ENUM. The type of address represented. The default value is "ipv6-addr". These values are maintained in the "Address-category" IANA registry per Section 10.2.

1. asn. Autonomous System Number.

2. atm. Asynchronous Transfer Mode (ATM) address.

3. e-mail. Email address, per the EMAIL data type.

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

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

6. ipv4-net-masked. A sanitized IPv4 address with significant bits per "ipv4-net" but with the character ’x’ replacing any digit(s) in the address or prefix.

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

8. ipv6-addr. IPv6 host address per Section 4 of [RFC5952].

9. ipv6-net. IPv6 network address, slash, prefix per Section 2.3 of [RFC4291].

10. ipv6-net-masked. A sanitized IPv6 address and prefix per "ipv6-net" but with the character 'x' replacing any hexadecimal digit(s) in the address or digit(s) in the prefix.

11. mac. Media Access Control (MAC) address (i.e., aa:bb:cc:dd:ee:ff).

12. site-uri. A URL or URI for a resource, per the URL data type.

13. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

ext-category
Optional. STRING. A means by which to extend the category attribute. See Section 5.1.1.

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

vlan-num
Optional. INTEGER. The number of the Virtual LAN to which the address belongs.

observable-id
Optional. ID. See Section 3.3.2.

3.18.2. NodeRole Class

The NodeRole class describes the function performed by or role of a particular system, asset or network.

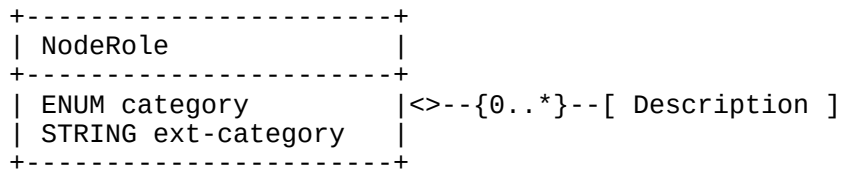

Figure 36: The NodeRole Class

The aggregate class of the NodeRole class is:

Description
: Zero or more. ML_STRING. A free-form text description of the role of the system.

The attributes of the NodeRole class are:

category
: Required. ENUM. Function or role of a node. These values are maintained in the "NodeRole-category" IANA registry per Section 10.2.

1. client. Client computer.
2. client-enterprise. Client computer on the enterprise network.
3. client-partner. Client computer on network of a partner.
4. client-remote. Client computer remotely connected to the enterprise network.
5. client-kiosk. Client computer serving as a kiosk.
6. client-mobile. Mobile device.
7. server-internal. Server with internal services.
8. server-public. Server with public services.
9. www. WWW server.
10. mail. Mail server.
11. webmail. Web mail server.
12. messaging. Messaging server (e.g., NNTP, IRC, IM).

13. streaming. Streaming-media server.

14. voice. Voice server (e.g., SIP, H.323).

15. file. File server.

16. ftp. FTP server.

17. p2p. Peer-to-peer node.

18. name. Name server (e.g., DNS, WINS).

19. directory. Directory server (e.g., LDAP, finger, whois).

20. credential. Credential server (e.g., domain controller, Kerberos).

21. print. Print server.

22. application. Application server.

23. database. Database server.

24. backup. Backup server.

25. dhcp. DHCP server.

26. assessment. Assessment server (e.g., vulnerability scanner, end-point assessment).

27. source-control. Source code control server.

28. config-management. Configuration management server.

29. monitoring. Security monitoring server (e.g., IDS).

30. infra. Infrastructure server (e.g., router, firewall, DHCP).

31. infra-firewall. Firewall.

32. infra-router. Router.

33. infra-switch. Switch.

34. camera. Camera and video system.

35. proxy. Proxy server.

36. remote-access. Remote access server.

37. log. Log server (e.g., syslog).

38. virtualization. Server running virtual machines.

39. pos. Point-of-sale device.

40. scada. Supervisory control and data acquisition (SCADA) system.

41. scada-supervisory. Supervisory system for a SCADA.

42. sinkhole. Traffic sinkhole destination.

43. honeypot. Honeypot server.

44. anonymization. Anonymization server (e.g., Tor node).

45. c2-server. Malicious command and control server.

46. malware-distribution. Server that distributes malware

47. drop-server. Server to which exfiltrated content is uploaded.

48. hop-point. Intermediary server used to get to a victim.

49. reflector. A system used in a reflector attack.

50. phishing-site. Site hosting phishing content.

51. spear-phishing-site. Site hosting spear-phishing content.

52. recruiting-site. Site to recruit.

53. fraudulent-site. Fraudulent site.

54. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

ext-category
: Optional. STRING. A means by which to extend the category attribute. See Section 5.1.1.

3.18.3. Counter Class

The Counter class summarizes multiple occurrences of an event or conveys counts or rates of various features.

The complete semantics of this class are context dependent based on the class in which it is aggregated.

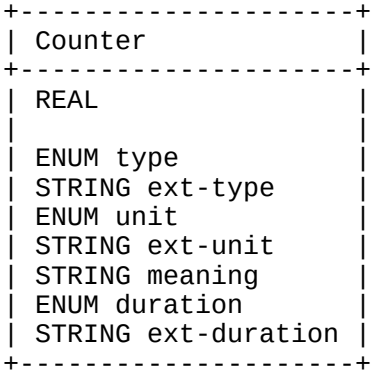

Figure 37: The Counter Class

The content of the class is a value of type REAL whose meaning and units are determined by the type and duration attributes, respectively. If the duration attribute is present, the element content is a rather. Otherwise, it is a simple counter.

The attributes of the Counter class are:

type
: Required. ENUM. Specifies the type of counter specified in the element content. These values are maintained in the "Counter-type" IANA registry per Section 10.2.

  1. count. The Counter class value is a counter.

  2. peak. The Counter class value is a peak value.

  3. average. The Counter class value is an average.

  4. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

ext-type

Optional. STRING. A means by which to extend the type attribute. See Section 5.1.1.

unit
Required. ENUM. Specifies the units of the element content. These values are maintained in the "Counter-unit" IANA registry per Section 10.2.

1. byte. Bytes transferred.

2. mbit. Megabits (Mbits) transfered.

3. packet. Packets.

4. flow. Network flow records.

5. session. Sessions.

6. alert. Notifications generated by another system (e.g., IDS or SIM).

7. message. Messages (e.g., mail messages).

8. event. Events.

9. host. Hosts.

10. site. Site.

11. organization. Organizations.

12. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

ext-unit
Optional. STRING. A means by which to extend the unit attribute. See Section 5.1.1.

meaning
Optional. STRING. A free-form text description of the metric represented by the Counter.

duration
Optional. ENUM. If present, the Counter class represents a rate. This attribute specifies unit of time over which the rate whose units are specified in the unit attribute is being conveyed. This attribute is the the denominator of the rate (where the unit

attribute specified the nominator). The possible values of this attribute are defined in the duration attribute of Section 3.12.3

ext-duration
Optional. STRING. A means by which to extend the duration attribute. See Section 5.1.1.

3.19. DomainData Class

The DomainData class describes a domain name and meta-data associated with this domain.

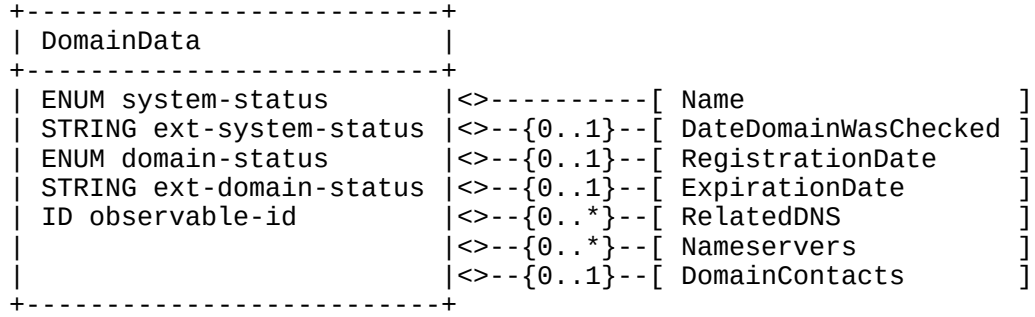

Figure 38: The DomainData Class

The aggregate classes of the DomainData class are:

Name
One. STRING. The domain name of a system.

DateDomainWasChecked
Zero or one. DATETIME. A timestamp of when the domain listed in the Name class was resolved.

RegistrationDate
Zero or one. DATETIME. A timestamp of when domain listed in Name class was registered.

ExpirationDate
Zero or one. DATETIME. A timestamp of when the domain listed in Name class is set to expire.

RelatedDNS
Zero or more. EXTENSION. Additional DNS records associated with this domain.

Nameservers

Zero or more. The name servers identified for the domain listed in Name class. See Section 3.19.1.

DomainContacts
Zero or one. Contact information for the domain listed in Name class supplied by the registrar or through a whois query.

The attributes of the DomainData class are:

system-status
Required. ENUM. Assesses the domain's involvement in the event. These values are maintained in the "DomainData-system-status" IANA registry per Section 10.2.

1. spoofed. This domain was spoofed.

2. fraudulent. This domain was operated with fraudulent intentions.

3. innocent-hacked. This domain was compromised by a third party.

4. innocent-hijacked. This domain was deliberately hijacked.

5. unknown. No categorization for this domain known.

6. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

ext-system-status
Optional. STRING. A means by which to extend the system-status attribute. See Section 5.1.1.

domain-status
Required. ENUM. Categorizes the registry status of the domain at the time the document was generated. These values and their associated descriptions are derived from Section 3.2.2 of [RFC3982]. These values are maintained in the "DomainData-domain-status" IANA registry per Section 10.2.

1. reservedDelegation. The domain is permanently inactive.

2. assignedAndActive. The domain is in a normal state.

3. assignedAndInactive. The domain has an assigned registration but the delegation is inactive.

4. assignedAndOnHold. The domain is in dispute.

5. revoked. The domain is in the process of being purged from the database.

6. transferPending. The domain is pending a change in authority.

7. registryLock. The domain is on hold by the registry.

8. registrarLock. Same as "registryLock".

9. other. The domain has a known status but it is not one of the redefined enumerated values.

10. unknown. The domain has an unknown status.

11. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

ext-domain-status
Optional. STRING. A means by which to extend the domain-status attribute. See Section 5.1.1.

observable-id
Optional. ID. See Section 3.3.2.

3.19.1. Nameservers Class

The Nameservers class describes the name servers associated with a given domain.

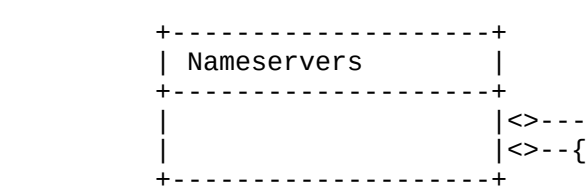

Figure 39: The Nameservers Class

The aggregate classes of the Nameservers class are:

Server
One. STRING. The domain name of the name server.

Address

One or more. The address of the name server. The value of the category attribute MUST be either "ipv4-addr" or "ipv6-addr". See Section 3.18.1.

The Nameservers class has no attributes.

3.19.2. DomainContacts Class

The DomainContacts class describes the contact information for a given domain provided either by the registrar or through a whois query.

This contact information can be explicitly described through a Contact class or a reference can be provided to a domain with identical contact information. Either a single SameDomainContact MUST be present or one or more Contact classes.

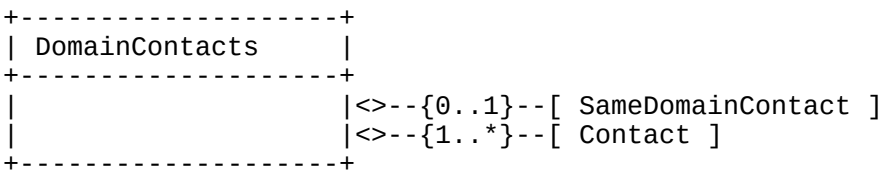

Figure 40: The DomainContacts Class

The aggregate classes of the DomainContacts class are:

SameDomainContact
: Zero or one. STRING. A domain name already cited in this document or through previous exchange that contains the identical contact information as the domain name in question. The domain contact information associated with this domain should be used instead of an explicit definition with the Contact class.

Contact
: One or more. Contact information for the domain. See Section 3.9.

The DomainContacts class has no attributes.

3.20. Service Class

The Service class describes a network service. The service is described by protocol, port, protocol header field and application providing or using the service.

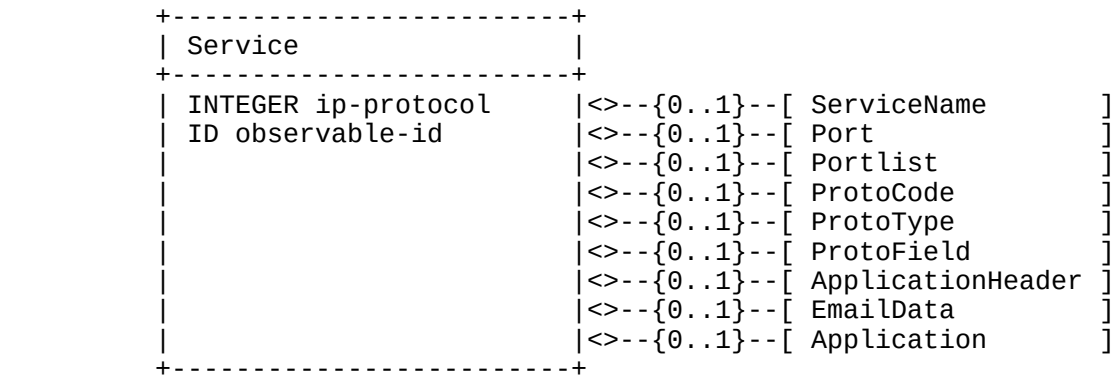

Figure 41: The Service Class

The aggregate classes of the Service class are:

ServiceName
: Zero or one. A protocol name.

Port
: Zero or one. INTEGER. A port number.

Portlist
: Zero or one. PORTLIST. A list of port numbers.

ProtoCode
: Zero or one. INTEGER. A transport layer (layer 4) protocol-specific code field (e.g., ICMP code field).

ProtoType
: Zero or one. INTEGER. A transport layer (layer 4) protocol specific type field (e.g., ICMP type field).

ProtoField
: Zero or one. INTEGER. A transport layer (layer 4) protocol specific flag field (e.g., TCP flag field).

ApplicationHeader
: Zero or one. A protocol header. See Section 3.20.2.

EmailData
: Zero or one. Headers associated with an email message. See Section 3.21.

Application

Zero or one. SOFTWARE. The application acting as either the client or server for the service.

At least one of these classes MUST be present.

When a given System classes with category="source" and another with category="target" are aggregated into a single Flow class, and each of these System classes has a Service and Portlist class, an implicit relationship between these Portlists exists. If N ports are listed for a System@category="source", and M ports are listed for System@category="target", the number of ports in N must be equal to M. Likewise, the ports MUST be listed in an identical sequence such that the n-th port in the source corresponds to the n-th port of the target. If N is greater than 1, a given instance of a Flow class MUST only have a single instance of a System@category="source" and System@category="target".

The attributes of the Service class are:

ip-protocol
Optional. INTEGER. The IANA assigned IP protocol number per [IANA.Protocols] The attribute MUST be set if a Port, Portlist, ProtoCode, ProtoType, ProtoField class is present.

observable-id
Optional. ID. See Section 3.3.2.

3.20.1. ServiceName Class

The ServiceName class identifies an application protocol. It can be described by referencing an IANA registered protocol, a URL or with free-form text.

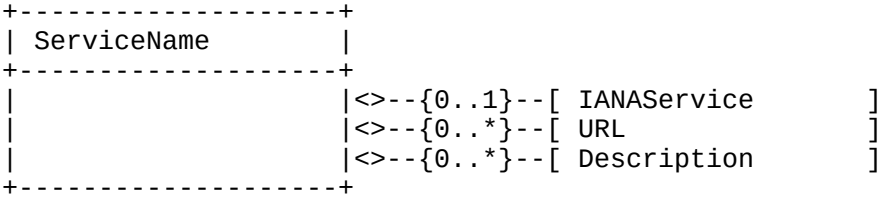

Figure 42: The ServiceName Class

The aggregate classes of the ServiceName class are:

IANAService
Zero or one. STRING. The name of the service per the "Service Name" field of the [IANA.Ports] registry.

URL
Zero or more. URL. A URL to a resource describing the service.

Description
Zero or more. ML_STRING. A free-form text description of the service.

At least one of these classes MUST be present.

The ServiceName class has no attributes.

3.20.2. ApplicationHeader Class

The ApplicationHeader class describes arbitrary fields from a protocol header and its corresponding value.

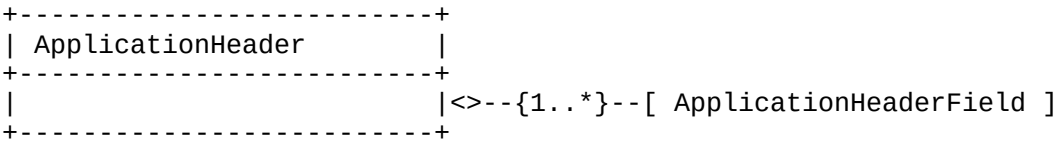

Figure 43: The ApplicationHeader Class

The aggregate class of the ApplicationHeader class is:

ApplicationHeaderField
One or more. EXTENSION. A field name and value in a protocol header. The 'name' attribute MUST be set to the field name. The field value MUST be set in the element content.

The ApplicationHeader class has no attributes.

3.21. EmailData Class

The EmailData class describes headers from an email message and cryptographic hash and signatures applied to it.

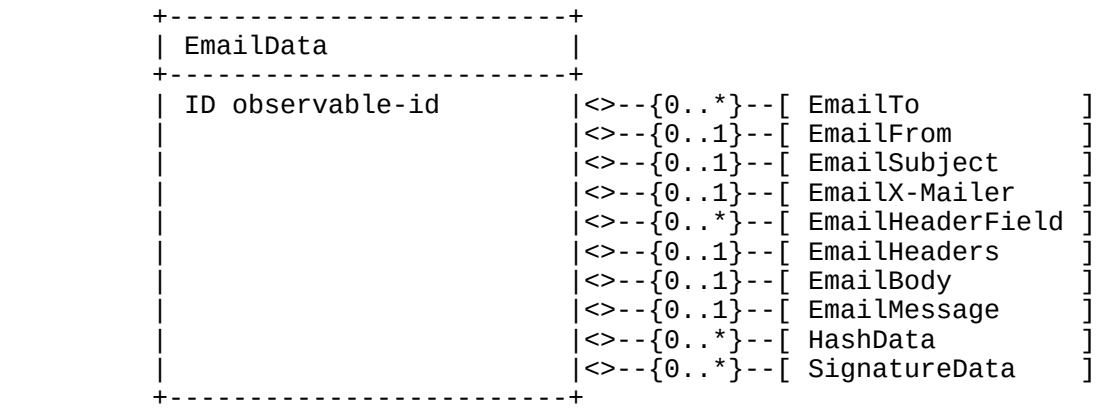

Figure 44: EmailData Class

The aggregate classes of the EmailData class are:

EmailTo
: Zero or more. EMAIL. The value of the "To:" header field (Section 3.6.3 of [RFC5322]) in an email.

EmailFrom
: Zero or one. EMAIL. The value of the "From:" header field (Section 3.6.2 of [RFC5322]) in an email.

EmailSubject
: Zero or one. STRING. The value of the "Subject:" header field in an email. See Section 3.6.4 of [RFC5322].

EmailX-Mailer
: Zero or one. STRING. The value of the "X-Mailer:" header field in an email.

EmailHeaderField
: Zero or more. EXTENSION. The header name and value of an arbitrary header field of the email message. The 'name' attribute MUST be set to header name. The header value MUST be set in the element body. The dtype attribute MUST be set to "string".

EmailHeaders
: Zero or one. STRING. The headers of an email message.

EmailBody
: Zero or one. STRING. The body of an email message.

EmailMessage

Zero or one. STRING. The headers and body of an email message.

HashData
Zero or more. Hash(es) associated with this email message. See Section 3.26.

SignatureData
Zero or more. Signature(s) associated with this email message. See Section 3.27.

The attribute of the EmailData class is:

observable-id
Optional. ID. See Section 3.3.2.

3.22. Record Class

The Record class is a container class for log and audit data that provides supportive information about the events in an incident. The source of this data will often be the output of monitoring tools. These logs substantiate the activity described in the document.

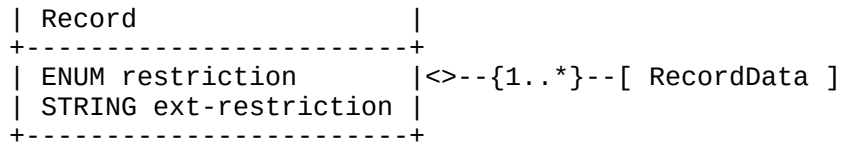

Figure 45: Record Class

The aggregate classes of the Record class are:

RecordData
One or more. Log or audit data generated by a particular tool. Separate instances of the RecordData class SHOULD be used for each type of log. See Section 3.22.1.

The attributes of the Record class are:

restriction
Optional. ENUM. See Section 3.3.1.

ext-restriction
Optional. STRING. A means by which to extend the restriction attribute. See Section 5.1.1.

3.22.1. RecordData Class

The RecordData class describes or references log or audit data from a given type of tool and provides a means to annotate the output.

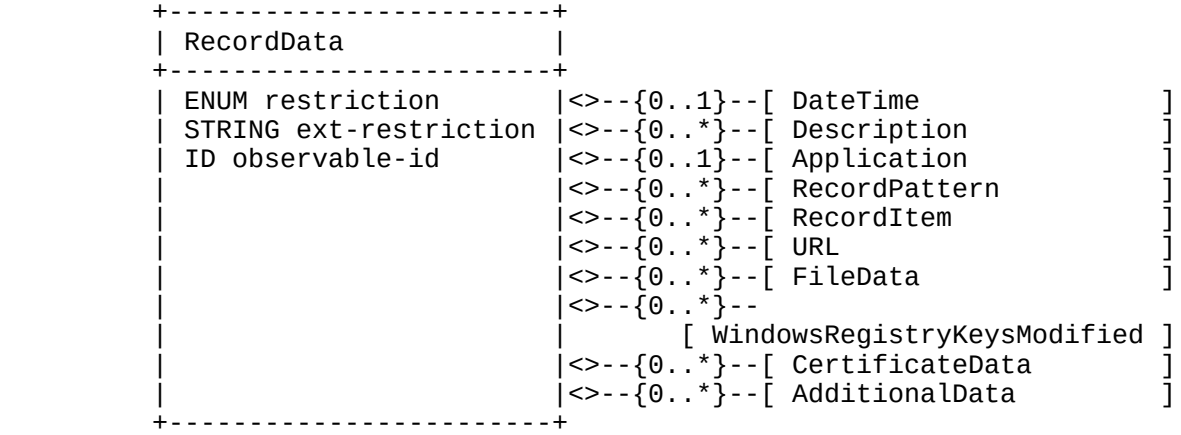

Figure 46: The RecordData Class

The aggregate classes of the RecordData class are:

DateTime
: Zero or one. DATETIME. A timestamp of the data found in the RecordItem or URL classes.

Description
: Zero or more. ML_STRING. A free-form text description of the data provided in the RecordItem or URL classes.

Application
: Zero or one. SOFTWARE. Identifies the tool used to generate the data in the RecordItem or URL classes.

RecordPattern
: Zero or more. A search string to precisely find the relevant data in the RecordItem or URL classes. See Section 3.22.2.

RecordItem
: Zero or more. EXTENSION. Log, audit, or forensic data to support the conclusions made during the course of analyzing the incident.

URL
: Zero or more. URL. A URL reference to a log or audit data.

FileData
Zero or one. The files involved in the incident. See Section 3.25.

WindowsRegistryKeysModified
Zero or more. The registry keys that were involved in the incident. See Section 3.23.

CertificateData
Zero or more. The certificates that were involved in the incident. See Section 3.24.

AdditionalData
Zero or more. EXTENSION. An extension mechanism for data not explicitly represented in the data model.

At least one of the following classes MUST be present: RecordItem, URL, FileData, WindowsRegistryKeysModified, CertificateData or AdditionalData.

The attributes of the RecordData class are:

restriction
Optional. ENUM. See Section 3.3.1.

ext-restriction
Optional. STRING. A means by which to extend the restriction attribute. See Section 5.1.1.

observable-id
Optional. ID. See Section 3.3.2.

3.22.2. RecordPattern Class

The RecordPattern class describes where in the log data provided or referenced in RecordData class relevant information can be found. It provides a way to reference subsets of information, identified by a pattern, in a large log file, audit trail, or forensic data.

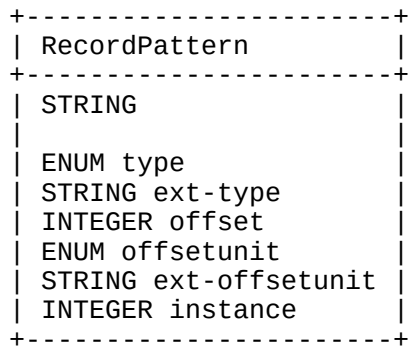

Figure 47: The RecordPattern Class

The content of the class is of type STRING and specifies a search pattern.

The attributes of the RecordPattern class are:

type
: Required. ENUM. Describes the type of pattern being specified in the element content. The default is "regex". These values are maintained in the "RecordPattern-type" IANA registry per Section 10.2.

  1. regex. regular expression as defined by POSIX Extended Regular Expressions (ERE) in Chapter 9 of [IEEE.POSIX].

  2. binary. Binhex encoded binary pattern, per the HEXBIN data type.

  3. xpath. XML Path (XPath) [W3C.XPATH]

  4. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

ext-type
: Optional. STRING. A means by which to extend the type attribute. See Section 5.1.1.

offset
: Optional. INTEGER. Amount of units (determined by the offsetunit attribute) to seek into the RecordItem data before matching the pattern.

offsetunit

Optional. ENUM. Describes the units of the offset attribute. The default is "line". These values are maintained in the "RecordPattern-offsetunit" IANA registry per Section 10.2.

1. line. Offset is a count of lines.

2. byte. Offset is a count of bytes.

3. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

ext-offsetunit
Optional. STRING. A means by which to extend the offsetunit attribute. See Section 5.1.1.

instance
Optional. INTEGER. Number of times to apply the specified pattern.

3.23. WindowsRegistryKeysModified Class

The WindowsRegistryKeysModified class describes Windows operating system registry keys and the operations that were performed on them. This class was derived from [RFC5901].

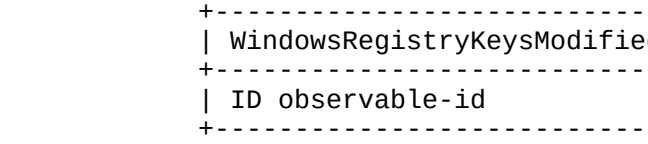

Figure 48: The WindowsRegistryKeysModified Class

The aggregate classes of the WindowsRegistryKeysModified class are:

Key
One or more. The Window registry key. See Section 3.23.1.

The attribute of the WindowsRegistryKeysModified class is:

observable-id
Optional. ID. See Section 3.3.2.

3.23.1. Key Class

The Key class describes a Windows operating system registry key name and value pair, and the operation performed on it.

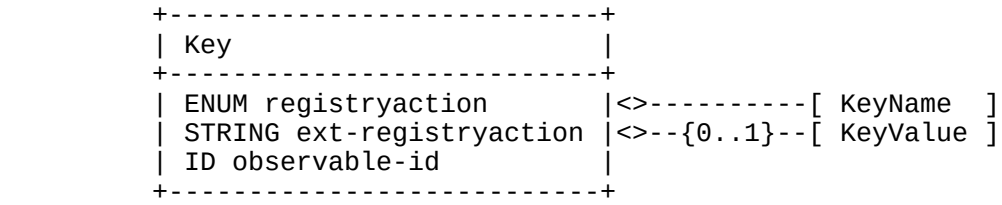

Figure 49: The Key Class

The aggregate classes of the Key class are:

KeyName
: One. STRING. The name of a Windows operating system registry key (e.g.,[HKEY_LOCAL_MACHINE\Software\Test\KeyName])

KeyValue
: Zero or one. STRING. The value of the registry key identified in the KeyName class encoded per the .reg file format [KB310516].

The attributes of the Key class are:

registryaction
: Optional. ENUM. The type of action taken on the registry key. These values are maintained in the "Key-registryaction" IANA registry per Section 10.2.

  1. add-key. Registry key added.

  2. add-value. Value added to a registry key.

  3. delete-key. Registry key deleted.

  4. delete-value. Value deleted from a registry key.

  5. modify-key. Registry key modified.

  6. modify-value. Value modified in a registry key.

  7. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

ext-registryaction
Optional. STRING. A means by which to extend the registryaction attribute. See Section 5.1.1.

observable-id
Optional. ID. See Section 3.3.2.

3.24. CertificateData Class

The CertificateData class describes X.509 certificates.

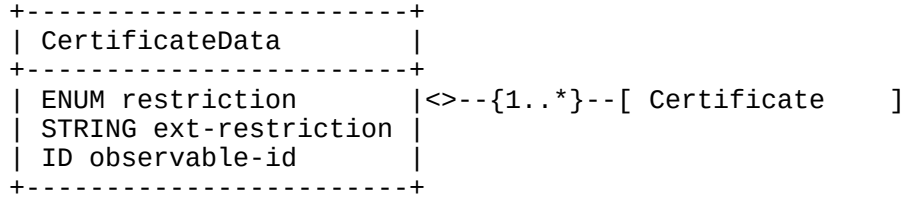

Figure 50: The CertificateData Class

The aggregate classes of the CertificateData class are:

Certificate
One or more. A description of an X.509 certificate or certificate chain. See Section 3.24.1.

The attributes of the CertificateData class are:

restriction
Optional. ENUM. See Section 3.3.1.

ext-restriction
Optional. STRING. A means by which to extend the restriction attribute. See Section 5.1.1.

observable-id
Optional. ID. See Section 3.3.2.

3.24.1. Certificate Class

The Certificate class describes a given X.509 certificate or certificate chain.

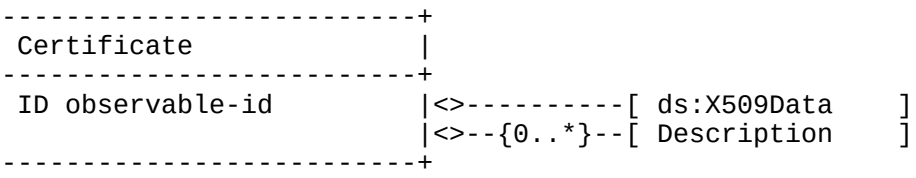

Figure 51: The Certificate Class

The aggregate classes of the Certificate class are:

ds:X509Data
: One. A given X.509 certificate or chain. See Section 4.4.4 of [W3C.XMLSIG].

Description
: Zero or more. ML_STRING. A free-form text description explaining the context of this certificate.

The attributes of the Certificate class are:

observable-id
: Optional. ID. See Section 3.3.2.

3.25. FileData Class

The FileData class describes a file or set of files.

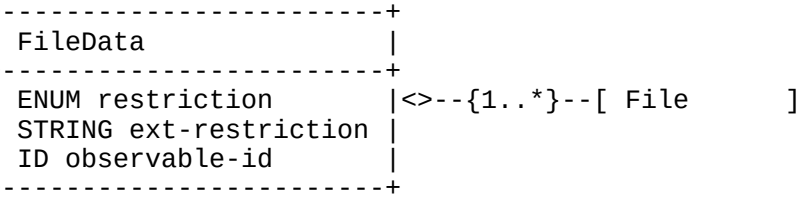

Figure 52: The FileData Class

The aggregate classes of the FileData class are:

File
: One or more. A description of a file. See Section 3.25.1.

The attributes of the FileData class are:

restriction
: Optional. ENUM. See Section 3.3.1.

ext-restriction
Optional. STRING. A means by which to extend the restriction attribute. See Section 5.1.1.

observable-id
Optional. ID. See Section 3.3.2.

3.25.1. File Class

The File class describes a file; its associated meta data; and cryptographic hashes and signatures applied to it.

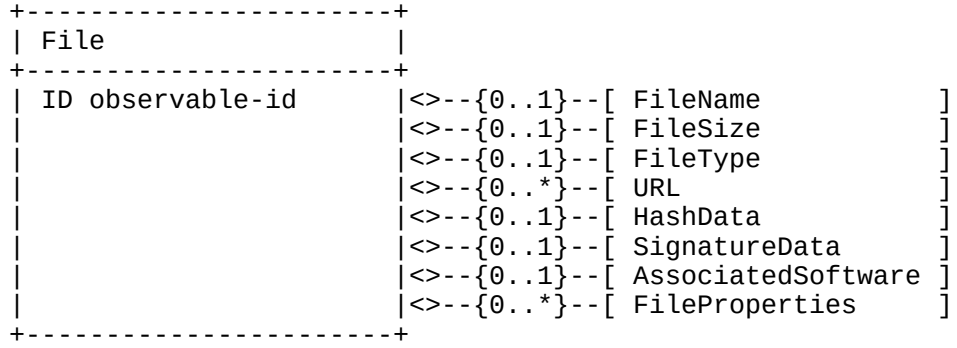

Figure 53: The File Class

The aggregate classes of the File class are:

FileName
Zero or One. STRING. The name of the file.

FileSize
Zero or One. INTEGER. The size of the file in bytes.

FileType
Zero or One. STRING. The type of file per the IANA Media Types Registry [IANA.Media]. Valid values correspond to the text in the "Template" column (e.g., "application/pdf").

URL
Zero or more. URL. A URL reference to the file.

HashData
Zero or One. Hash(es) associated with this file. See Section 3.26.

SignatureData

Zero or One. Signature(s) associated with this file. See Section 3.27.

AssociatedSoftware
Zero or One. SOFTWARE. The software application or operating system to which this file belongs or by which it can be processed.

FileProperties
Zero or more. EXTENSION. Mechanism by which to extend the data model to describe properties of the file.

The attributes of the File class are:

observable-id
Optional. ID. See Section 3.3.2.

3.26. HashData Class

The HashData class describes different types of hashes on an given object (e.g., file, part of a file, email).

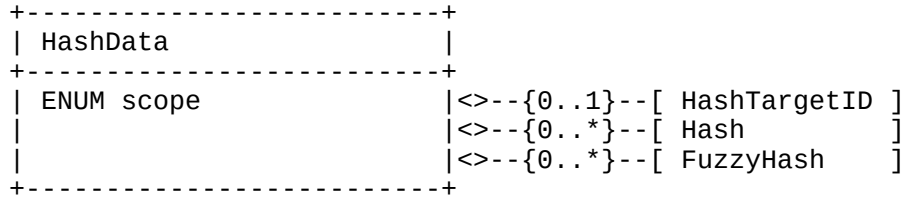

Figure 54: The HashData Class

The aggregate classes of the HashData class are:

HashTargetID
Zero or One. STRING. An identifier that references a subset of the object being hashed. The semantics of this identifier are specified by the scope attribute.

Hash
Zero or more. The hash of an object. See Section 3.26.1.

FuzzyHash
Zero or more. The fuzzy hash of an object. See Section 3.26.2.

At least one instance of either Hash or FuzzyHash MUST be present.

The attribute of the HashData class is:

scope
Required. ENUM. Describes on which part of the object the hash should be applied. These values are maintained in the "HashData-scope" IANA registry per Section 10.2.

1. file-contents. A hash computed over the entire contents of a file.

2. file-pe-section. A hash computed on a given section of a Windows Portable Executable (PE) file. If set to this value, the HashTargetID class MUST identify the section being hashed. A section is identified by an ordinal number (starting at 1) corresponding to the the order in which the given section header was defined in the Section Table of the PE file header.

3. file-pe-iat. A hash computed on the Import Address Table (IAT) of a PE file. As IAT hashes are often tool dependent, if this value is set, the Application class of either the Hash or FuzzyHash classes MUST specify the tool used to generate the hash.

4. file-pe-resource. A hash computed on a given resource in a PE file. If set to this value, the HashTargetID class MUST identify the resource being hashed. A resource is identified by an ordinal number (starting at 1) corresponding to the order in which the given resource is declared in the Resource Directory of the Data Dictionary in the PE file header.

5. file-pdf-object. A hash computed on a given object in a Portable Document Format (PDF) file. If set to this value, the HashTargetID class MUST identify the object being hashed. This object is identified by its offset in the PDF file.

6. email-hash. A hash computed over the headers and body of an email message.

7. email-headers-hash. A hash computed over all of the headers of an email message.

8. email-body-hash. A hash computed over the body of an email message.

9. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

ext-scope

Optional. STRING. A means by which to extend the scope attribute. See Section 5.1.1.

3.26.1. Hash Class

The Hash class describes a cryptographic hash value; the algorithm and application used to generate it; and the canonicalization method applied to the object being hashed.

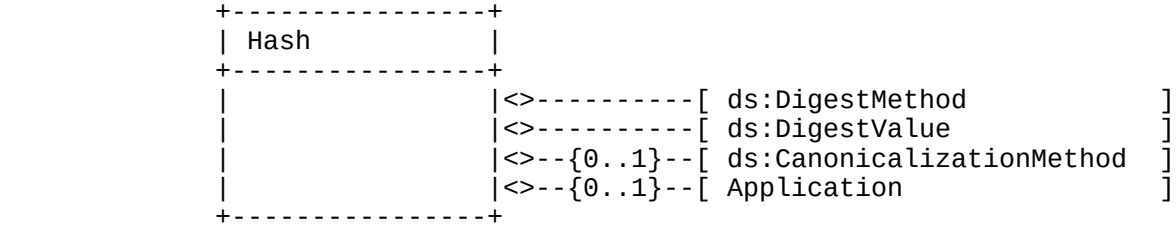

Figure 55: The Hash Class

The aggregate classes of the Hash class are:

ds:DigestMethod
One. The hash algorithm used to generate the hash. See Section 4.3.3.5 of [W3C.XMLSIG]

ds:DigestValue
One. The computed hash value. See Section 4.3.3.6 of [W3C.XMLSIG].

ds:CanonicalizationMethod
Zero or one. The canonicalization method used on the object being hashed. See Section 4.3.1 of [W3C.XMLSIG].

Application
Zero or One. SOFTWARE. The application used to calculate the hash.

The HashData class has no attributes.

3.26.2. FuzzyHash Class

The FuzzyHash class describes a fuzzy hash and the application used to generate it.

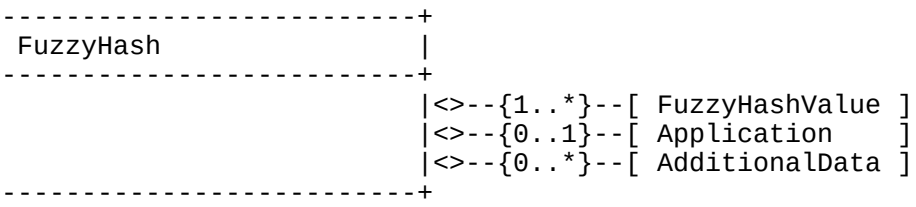

Figure 56: The FuzzyHash Class

The aggregate classes of the FuzzyHash class are:

FuzzyHashValue
: One or more. EXTENSION. The computed fuzzy hash value.

Application
: Zero or one. SOFTWARE. The application used to calculate the hash.

AdditionalData
: Zero or more. EXTENSION. Mechanism by which to extend the data model.

The FuzzyData class has no attributes.

3.27. SignatureData Class

The SignatureData class describes different types of digital signatures on an object.

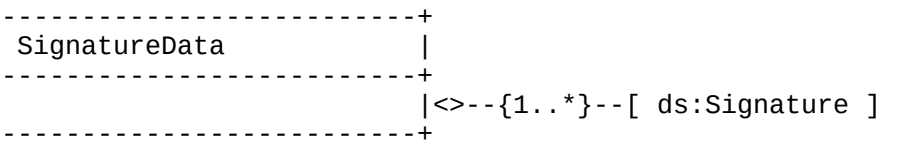

Figure 57: The SignatureData Class

The aggregate class of the SignatureData class is:

Signature
: One or more. An given signature. See Section 4.2 of [W3C.XMLSIG]

The SignatureData class has no attributes.

3.28. IndicatorData Class

The IndicatorData class describes indicators and meta-data associated with them.

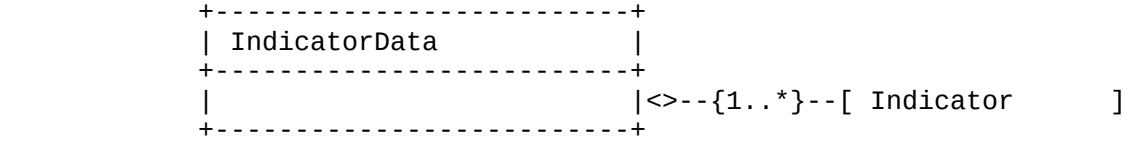

Figure 58: The IndicatorData Class

The aggregate class of the IndicatorData class is:

Indicator
One or more. A description of an indicator. See Section 3.29.

The IndicatorData class has no attributes.

3.29. Indicator Class

The Indicator class describes an indicator. An indicator consists of observable features and phenomenon that aid in the forensic or proactive detection of malicious activity; and associated meta-data. An indicator can be described outright; by referencing or composing previously defined indicators; or by referencing observables described in the incident report found in this document.

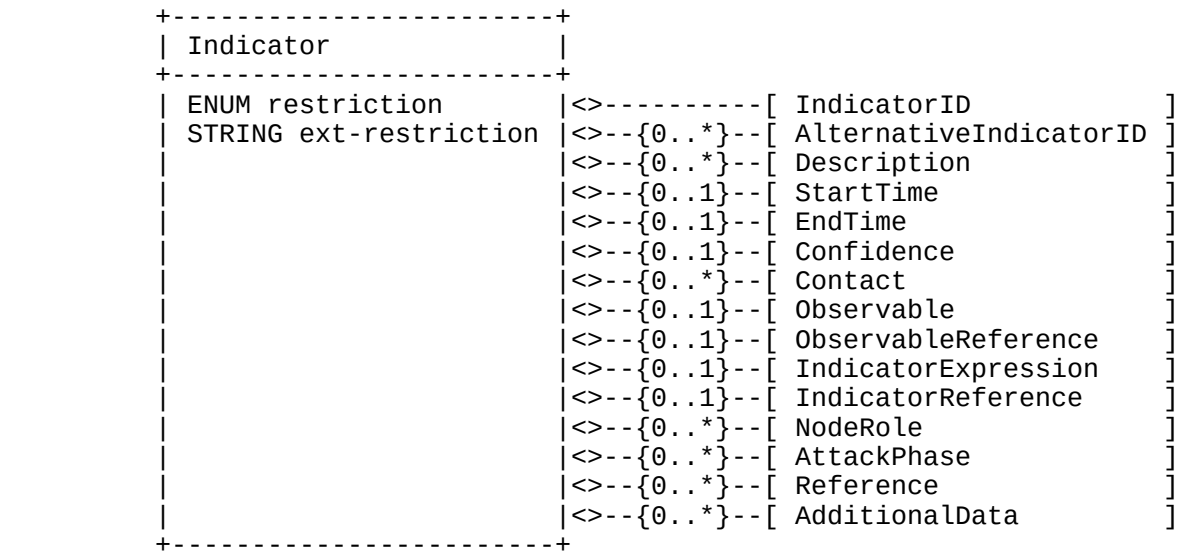

Figure 59: The Indicator Class

The aggregate classes of the Indicator class are:

IndicatorID
: One. An identifier for this indicator. See Section 3.29.1

AlternativeIndicatorID
: Zero or more. An alternative identifier for this indicator. See Section 3.29.2

Description
: Zero or more. ML_STRING. A free-form text description of the indicator.

StartTime
: Zero or one. DATETIME. A timestamp of the start of the time period during which this indicator is valid.

EndTime
: Zero or one. DATETIME. A timestamp of the end of the time period during which this indicator is valid.

Confidence
: Zero or one. An estimate of the confidence in the quality of the indicator. See Section 3.12.5.

Contact

Zero or more. Contact information for this indicator. See Section 3.9.

Observable
Zero or one. An observable feature or phenomenon of this indicator. See Section 3.29.3.

ObservableReference
Zero or one. A reference to an observable feature or phenomenon defined elsewhere in the document. See Section 3.29.6.

IndicatorExpression
Zero or one. A composition of observables. See Section 3.29.4.

IndicatorReference
Zero or one. A reference to an indicator. See Section 3.29.7.

NodeRole
Zero or more. The role of the system in the attack should this indicator be matched to it. See Section 3.18.2.

AttackPhase
Zero or more. The phase in an attack lifecycle during which this indicator might be seen. See Section 3.29.8.

Reference
Zero or more. A reference to additional information relevant to this indicator. See Section 3.11.1.

AdditionalData
Zero or more. EXTENSION. Mechanism by which to extend the data model.

The Indicator class MUST have exactly one instance of an Observable, IndicatorExpression, ObservableReference, or IndicatorReference class.

The StartTime and EndTime classes can be used to define an interval during which the indicator is valid. If both classes are present, the indicator is consider valid only during the described interval. If neither class is provided, the indicator is considered valid during any time interval. If only a StartTime is provided, the indicator is valid anytime after this timestamp. If only an EndTime is provided, the indicator is valid anytime prior to this timestamp.

The attributes of the Indicator class are:

restriction

Optional. ENUM. See Section 3.3.1.

ext-restriction
Optional. STRING. A means by which to extend the restriction attribute. See Section 5.1.1.

3.29.1. IndicatorID Class

The IndicatorID class identifies an indicator with a globally unique identifier. The combination of the name and version attributes, and the element content form this identifier. Indicators generated by given CSIRT MUST NOT reuse the same value unless they are referencing the same indicator.

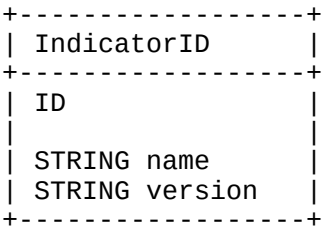

Figure 60: The IndicatorID Class

The content of the class is of type ID and specifies an identifier for an indicator.

The attributes of the IndicatorID class are:

name
Required. STRING. An identifier describing the CSIRT that created the indicator. In order to have a globally unique CSIRT name, the fully qualified domain name associated with the CSIRT MUST be used. This format is identical to the IncidentID@name attribute in Section 3.4.

version
Required. STRING. A version number of an indicator.

3.29.2. AlternativeIndicatorID Class

The AlternativeIndicatorID class lists alternative identifiers for an indicator.

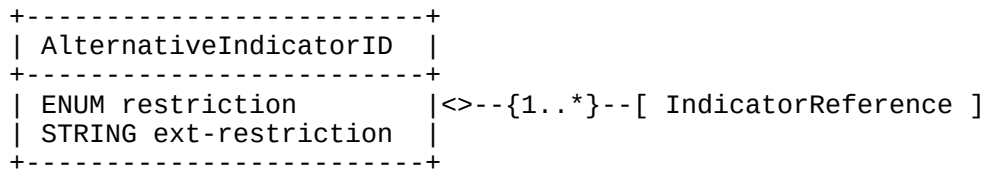

Figure 61: The AlternativeIndicatorID Class

The aggregate class of the AlternativeIndicatorID class is:

IndicatorReference
: One or more. A reference to an indicator. See Section 3.29.7

The attributes of the AlternativeIndicatorID class are:

restriction
: Optional. ENUM. See Section 3.3.1.

ext-restriction
: Optional. STRING. A means by which to extend the restriction attribute. See Section 5.1.1.

3.29.3. Observable Class

The Observable class describes a feature and phenomenon that can be observed or measured for the purposes of detecting malicious behavior.

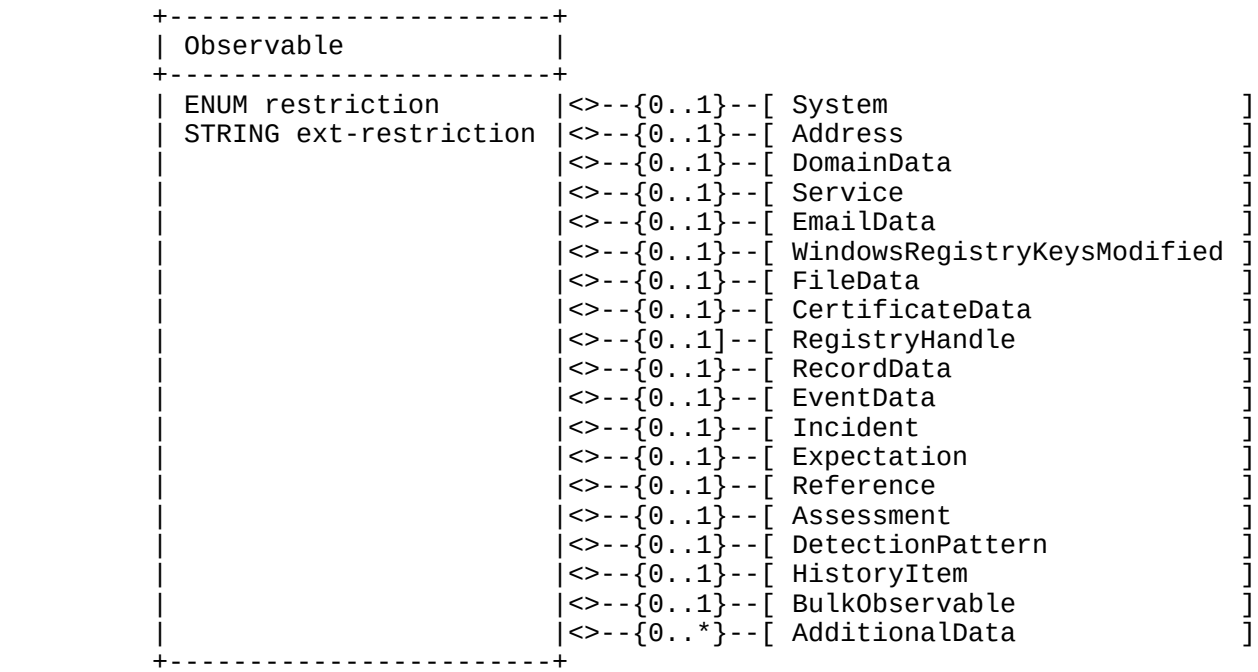

Figure 62: The Observable Class

The aggregate classes of the Observable class are:

System
: Zero or one. An System observable. See Section 3.17.

Address
: Zero or one. An Address observable. See Section 3.18.1.

DomainData
: Zero or one. A DomainData observable. See Section 3.19.

Service
: Zero or one. A Service observable. See Section 3.20.

EmailData
: Zero or one. A EmailData observable. See Section 3.21.

WindowsRegistryKeysModified
: Zero or one. A WindowsRegistryKeysModified observable. See Section 3.23.

FileData

Zero or one. A FileData observable. See Section 3.25.

CertificateData
Zero or one. A CertificateData observable. See Section 3.24.

RegistryHandle
Zero or one. A RegistryHandle observable. See Section 3.9.1.

RecordData
Zero or one. A RecordData observable. See Section 3.22.1.

EventData
Zero or one. An EventData observable. See Section 3.14.

Incident
Zero or one. An Incident observable. See Section 3.2.

Expectation
Zero or one. An Expectation observable. See Section 3.15.

Reference
Zero or one. A Reference observable. See Section 3.11.1.

Assessment
Zero or one. An Assessment observable. See Section 3.12.

DetectionPattern
Zero or one. A DetectionPattern observable. See Section 3.12.

HistoryItem
Zero or one. A HistoryItem observable. See Section 3.13.1.

BulkObservable
Zero or one. A bulk list of observables. See Section 3.29.3.1.

AdditionalData
Zero or more. EXTENSION. Mechanism by which to extend the data model.

The Observable class MUST have exactly one of the possible child classes.

The attributes of the Observable class are:

restriction
Optional. ENUM. See Section 3.3.1.

ext-restriction

Optional. STRING. A means by which to extend the restriction attribute. See Section 5.1.1.

3.29.3.1. BulkObservable Class

The BulkObservable class allows the enumeration of a single type of observables without requiring each one to be encoded individually in multiple instances of the same class.

The type attribute describes the type of observable listed in the child BulkObservableList class. The BulkObservableFormat class optionally provides additional meta-data.

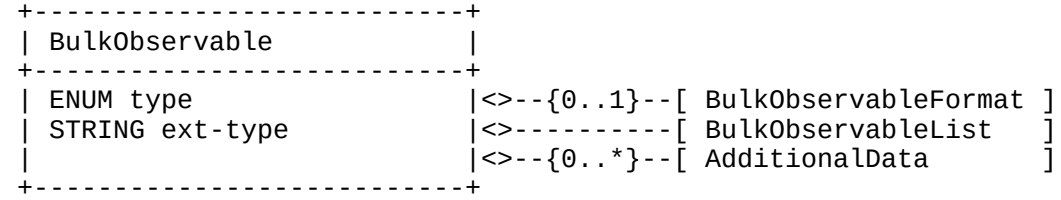

Figure 63: The BulkObservable Class

The aggregate classes of the BulkObservable class are:

BulkObservableFormat
: Zero or one. Provides additional meta-data about the observables enumerated in the BulkObservableList class. See Section 3.29.3.1.1.

BulkObservableList
: One. STRING. A list of observables, one per line. Each line is separated with either a LF character or CR-and-LF characters. The type attribute specifies which observables will be listed.

AdditionalData
: Zero or more. EXTENSION. Mechanism by which to extend the data model.

The attributes of the BulkObservable class are:

type
: Optional. ENUM. The type of the observable listed in the child ObservableList class. These values are maintained in the "BulkObservable-type" IANA registry per Section 10.2.

  1. asn. Autonomous System Number (per the Address@category attribute).

2. atm. Asynchronous Transfer Mode (ATM) address (per the Address@category attribute).

3. e-mail. Email address (per the Address@category attribute).

4. ipv4-addr. IPv4 host address in dotted-decimal notation (e.g., 192.0.2.1) (per the Address@category attribute).

5. ipv4-net. IPv4 network address in dotted-decimal notation, slash, significant bits (e.g., 192.0.2.0/24) (per the Address@category attribute).

6. ipv4-net-mask. IPv4 network address in dotted-decimal notation, slash, network mask in dotted-decimal notation (i.e., 192.0.2.0/255.255.255.0) (per the Address@category attribute).

7. ipv6-addr. IPv6 host address (e.g., 2001:DB8::3) (per the Address@category attribute).

8. ipv6-net. IPv6 network address, slash, significant bits (e.g., 2001:DB8::/32) (per the Address@category attribute).

9. ipv6-net-mask. IPv6 network address, slash, network mask (per the Address@category attribute).

10. mac. Media Access Control (MAC) address (i.e., a:b:c:d:e:f) (per the Address@category attribute).

11. site-uri. A URL or URI for a resource (per the Address@category attribute).

12. domain-name. A fully qualified domain name or part of a name. (e.g., fqdn.example.com, example.com).

13. domain-to-ipv4. A fqdn-to-IPv4 address mapping specified as a comma separated list (e.g., "fqdn.example.com, 192.0.2.1").

14. domain-to-ipv6. A fqdn-to-IPv6 address mapping specified as a comma separated list (e.g., "fqdn.example.com, 2001:DB8::3").

15. domain-to-ipv4-timestamp. Same as domain-to-ipv4 but with a timestamp (in the DATETIME format) of the resolution (e.g., "fqdn.example.com, 192.0.2.1, 2015-06-11T00:38:31-06:00").

16. domain-to-ipv6-timestamp. Same as domain-to-ipv6 but with a timestamp (in the DATETIME format) of the resolution (e.g., "fqdn.example.com, 2001:DB8::3, 2015-06-11T00:38:31-06:00").

17. ipv4-port. An IPv4 address, port and protocol tuple (e.g., 192.0.2.1, 80, tcp). The protocol name corresponds to the "Keyword" column in the [IANA.Protocols] registry.

18. ipv6-port. An IPv6 address, port and protocol tuple (e.g., 2001:DB8::3, 80, tcp). The protocol name corresponds to the "Keyword" column in the [IANA.Protocols] registry.

19. windows-reg-key. A Microsoft Windows Registry key.

20. file-hash. A file hash. The format of this hash is described in the Hash class that MUST be present in a sibling BulkObservableFormat class.

21. email-x-mailer. An X-Mailer field from an email.

22. email-subject. An email subject line.

23. http-user-agent. A User Agent field from an HTTP request header (e.g., "Mozilla/5.0 (Windows NT 6.3; WOW64; rv:38.0) Gecko/20100101 Firefox/38.0").

24. http-request-uri. The Request URI from an HTTP request header.

25. mutex. The name of a system mutex.

26. file-path. A file path (e.g., "/tmp/local/file", "c:\windows\system32\file.sys")

27. user-name. A username.

28. ext-value. A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. See Section 5.1.1.

ext-type
: Optional. STRING. A means by which to extend the type attribute. See Section 5.1.1.

3.29.3.1.1. BulkObservableFormat Class

The ObservableFormat class specifies meta-data about the format of an observable enumerated in a sibling BulkObservableList class.

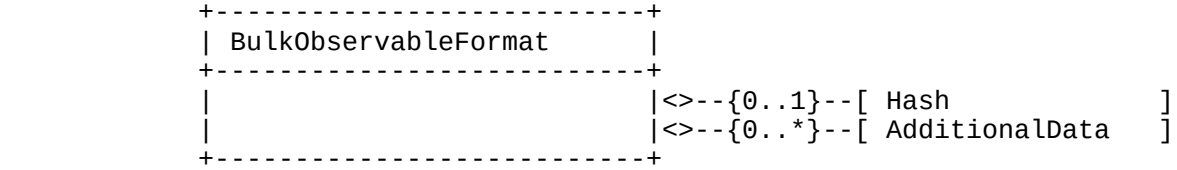

Figure 64: The BulkObservableFormat Class

The aggregate classes of the BulkObservableFormat class are:

Hash
: Zero or one. Describes the format of a hash. See Section 3.26.1.

AdditionalData
: Zero or more. EXTENSION. Mechanism by which to extend the data model.

The BulkObservableFormat class has no attributes.

Either Hash or AdditionalData MUST be present.

3.29.4. IndicatorExpression Class

The IndicatorExpression describes an expression composed of observed phenomenon or features, or indicators. Elements of the expression can be described directly, reference relevant data from other parts of a given IODEF document, or reference previously defined indicators.

All child classes of a given instance of IndicatorExpression form a boolean algebraic expression where the operator between them is determined by the operator attribute.

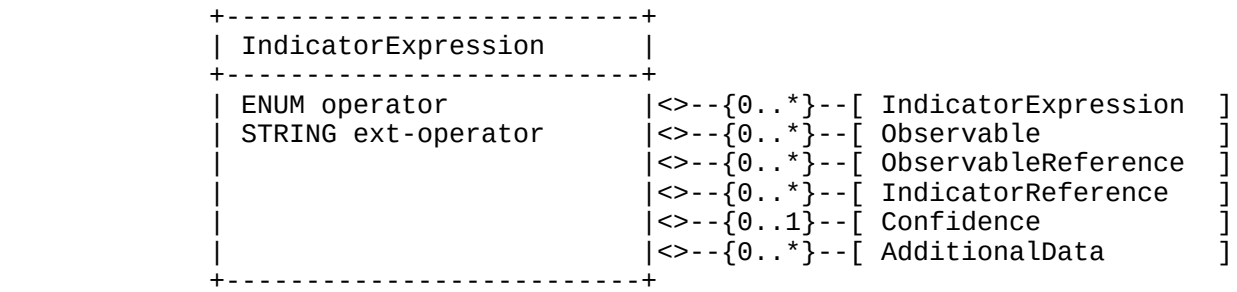

Figure 65: The IndicatorExpression Class

The aggregate classes of the IndicatorExpression class are:

IndicatorExpression
: Zero or more. An expression composed of other observables or indicators. See Section 3.29.4.

Observable
: Zero or more. A description of an observable. See Section 3.29.3.

ObservableReference
: Zero or more. A reference to an observable. See Section 3.29.6.

IndicatorReference
: Zero or more. A reference to an indicator. See Section 3.29.7.

Confidence
: Zero or one. An estimate of the confidence in the quality of the terms expressed in the expression. See Section 3.12.5.

AdditionalData
: Zero or more. EXTENSION. Mechanism by which to extend the data model.

The attributes of the IndicatorExpression class are:

operator
: Optional. ENUM. The operator to be applied between the child elements. See Section 3.29.5 for parsing guidance. The default value is "and". These values are maintained in the "IndicatorExpression-operator" IANA registry per Section 10.2.

  1. not. negation operator.

  2. and. conjunction operator.

3. or. disjunction operator.

4. xor. exclusive disjunction operator.

ext-operator
Optional. STRING. A means by which to extend the operator attribute. See Section 5.1.1.

3.29.5. Expressions with IndicatorExpression

Boolean algebraic expressions can be used to specify relationships between observables and indicator. These expressions are constructed through the use of the operator attribute and parent-child relationships in IndicatorExpressions. These expressions should be parsed as follows:

1. The operator specified by the operator attribute is applied between each of the child elements of the immediate parent IndicatorExpression element. If no operator attribute is specified, it should be assumed to be the conjunction operator (i.e., operator="and").

2. A nested IndicatorExpression element with a parent IndicatorExpression is the equivalent of a parentheses in the expression.

The following four examples in Figure 66 through Figure 70 illustrate these parsing rules:

```
1     : <IndicatorExpression>
2 [O1]:    <Observable>..</Observable>
3 [O2]:    <Observable>..</Observable>
4     : </IndicatorExpression>
```

Equivalent expression: (O1 AND O2)

Figure 66: Nested elements in an IndicatorExpression without an operator attribute specified

```
1     : <IndicatorExpression operator="or">
2 [O1]:    <Observable>..</Observable>
3 [O2]:    <Observable>..</Observable>
4     : </IndicatorExpression>
```

Equivalent expression: (O1 OR O2)

Figure 67: Nested elements in an IndicatorExpression with an operator attribute specified

```
1     : <IndicatorExpression operator="or">
2     :    <IndicatorExpression operator="or">
3 [O1]:      <Observable>..</Observable>
4 [O2]:      <Observable>..</Observable>
5     :    </IndicatorExpression>
6 [O3]:    <Observable>..</Observable>
7     : </IndicatorExpression>
```

Equivalent expression: ((O1 OR O2) OR O3)

Figure 68: Nested elements with a recursive IndicatorExpression with an operator attribute specified

```
1     : <IndicatorExpression operator="not">
2     :    <IndicatorExpression operator="and">
3 [O1]:      <Observable>..</Observable>
4 [O2]:      <Observable>..</Observable>
5     :    </IndicatorExpression>
6     : </IndicatorExpression>
```

Equivalent expression: (NOT (O1 AND O2))

Figure 69: A recursive IndicatorExpression with an operator attribute specified

```
 1                          :    <IndicatorExpression operator="or">
 2                          :      <IndicatorExpression>
 3 [O1 with low confidence] :        <Observable>..</Observable>
 4                          :        <Confidence rating="low" />
 5                          :      </IndicatorExpression>
 6                          :      <IndicatorExpression>
 7 [O2 with high confidence]:        <Observable>..</Observable>
 8                          :        <Confidence rating="high" />
 9                          :      </IndicatorExpression>
10                          :    </IndicatorExpression>
```

Equivalent expression: ((O1) OR (O2))

Figure 70: Varying confidence on particular Observables

Invalid algebraic expressions while valid XML, MUST NOT be specified.

3.29.6. ObservableReference Class

The ObservableReference describes a reference to an observable feature or phenomenon described elsewhere in the document.

The ObservableReference class has no content.

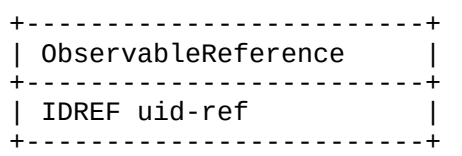

Figure 71: The ObservableReference Class

The ObservableReference class has no content.

The attribute of the ObservableReference class is:

uid-ref
: Required. IDREF. An identifier that serves as a reference to a class in the IODEF document. The referenced class will have this identifier set in its observable-id attribute.

3.29.7. IndicatorReference Class

The IndicatorReference describes a reference to an indicator. This reference may be to an indicator described in this IODEF document or in a previously exchanged IODEF document.

The IndicatorReference class has no content.

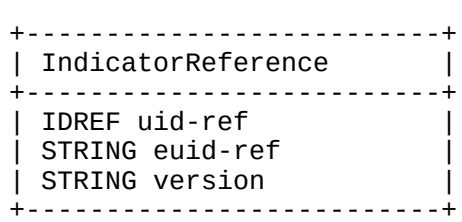

Figure 72: The IndicatorReference Class

The attributes of the IndicatorReference class are:

uid-ref
: Optional. IDREF. An identifier that references an Indicator class in the IODEF document. The referenced Indicator class will have this identifier set in its IndicatorID class.

euid-ref
: Optional. STRING. An identifier that references an IndicatorID not in this IODEF document.

version
Optional. STRING. A version number of an indicator.

Either the uid-ref or the euid-ref attribute MUST be set.

3.29.8. AttackPhase Class

The AttackPhase class describes a particular phase of an attack lifecycle.

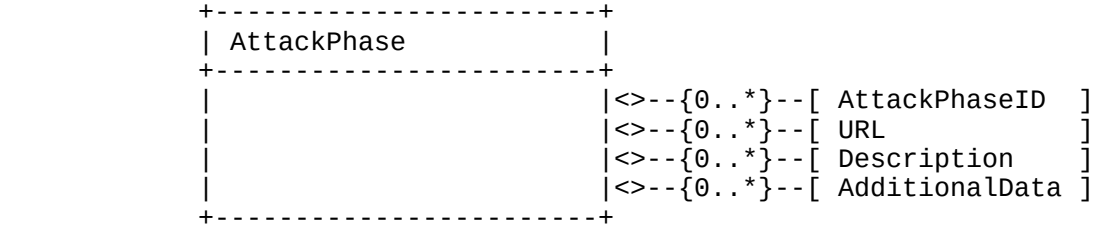

Figure 73: AttackPhase Class

The aggregate classes of the AttackPhase class are:

AttackPhaseID
Zero or more. STRING. An identifier for the phase of the attack.

URL
Zero or more. URL. A URL to a resource describing this phase of the attack.

Description
Zero or more. ML_STRING. A free-form text description of this phase of the attack.

AdditionalData
Zero or more. EXTENSION. A mechanism by which to extend the data model.

AttackPhase MUST have at least one instance of a child class.

The AttackPhase class has no attributes.

4. Processing Considerations

This section provides additional requirements and guidance on creating and processing IODEF documents.

4.1. Encoding

Every IODEF document MUST begin with an XML declaration and MUST specify the XML version used. The character encoding MUST also be explicitly specified. UTF-8 [RFC3629] SHOULD be used unless UTF-16 [RFC2781] is necessary. Encodings other than UTF-8 and UTF-16 SHOULD NOT be used. The IODEF conforms to all XML data encoding conventions and constraints.

The XML declaration with UTF-8 character encoding will read as follows:

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

Certain characters have special meaning in XML and MUST not appear in literal form. Per Section 2.4 of [W3C.XML], these characters MUST be escaped with a numeric character or entity reference.

4.2. IODEF Namespace

The IODEF schema declares a namespace of "urn:ietf:params:xml:ns:iodef-2.0" and registers it per [W3C.XMLNS]. Each IODEF document MUST include a valid reference to the IODEF schema using the "xsi:schemaLocation" attribute. An example of such a declaration would look as follows:

```
<IODEF-Document
   version="2.00" lang="en-US"
   xmlns:iodef="urn:ietf:params:xml:ns:iodef-2.0"
   xsi:schemaLocation="urn:ietf:params:xmls:schema:iodef-2.0" ...>
```

4.3. Validation

IODEF documents MUST be well-formed XML. It is RECOMMENDED that recipients validate the document against the schema described in Section 8. However, mere conformance to this schema is not sufficient for a semantically valid IODEF document. The text of Section 3 describes further formatting and constraints; some that cannot be conveniently encoded in the schema. These MUST also be considered by an IODEF implementation. Furthermore, the enumerated values present in this document are a static list that will be incomplete over time as select attributes can be extended by a corresponding IANA registry per Section 10.2. Therefore, IODEF implementations SHOULD periodically update their schema and MAY need to update their parsing algorithms to incorporate newly registered values.

4.4. Incompatibilities with v1

The IODEF data model in this document makes a number of changes to [RFC5070]. These changes were largely additive -- classes and enumerated values were added. However, some incompatibilities between [RFC5070] and this new specification were introduced. These incompatibilities are as follows:

- The IODEF-Document@version attribute is set to "2.0".
- Attributes with enumerated values can now also be extended with IANA registries.
- All iodef:MLStringType classes use xml:lang. IODEF-Document also uses xml:lang.
- The Service@ip_protocol attribute was renamed to @ip-protocol.
- The Node/NodeName class was removed in favor of representing domain names with Node/DomainData/Name class. The Node/DataTime class was also removed so that the Node/DomainData/ DateDomainWasChecked class can represent the time at which the name to address resolution occurred.
- The Node/NodeRole class was moved to System/NodeRole.
- The Reference class is now defined by [RFC7495].
- The data previously represented in the Impact class is now in the SystemImpact and IncidentCategory classes. The Impact class has been removed.
- The semantics of Counter@type are now represented in Counter@unit.
- The IODEF-Document@formatid attribute has been renamed to @format-id.
- Incident/ReportTime is no longer mandatory. However, GenerationTime is.
- The Fax class was removed and is now represented by a generic Telephone class.
- The Telephone, Email and PostalAddress classes were redefined from improved internationalization.
- The "ipv6-net-mask" value was remove from category attribute of Address.

5. Extending the IODEF

In order to support the dynamic nature of security operations, the IODEF data model will need to continue to evolve. This section discusses how new data elements can be incorporated into the IODEF. There is support to add additional enumerated values and new classes. Adding additional attributes to existing classes is not supported.

These extension mechanisms are designed so that adding new data elements is possible without requiring a modifications to this document. Extensions can be implemented publicly or privately. With proven value, well documented extensions can be incorporated into future versions of the specification.

5.1. Extending the Enumerated Values of Attributes

Additional enumerated values can be added to select attributes either through the use of specially marked attributes with the "ext-" prefix or through a set of corresponding IANA registries. The former approach allows for the extension to remain private. The latter approach is public.

5.1.1. Private Extension of Enumerated Values

The data model supports adding new enumerated values to an attribute without public registration. For each attribute that supports this extension technique, there is a corresponding attribute in the same element whose name is identical but with a prefix of "ext-". This special attribute is referred to as the extension attribute. 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 attributes.

In addition to a corresponding extension attribute, each extensible attribute has "ext-value" as one its possible enumerated values. Selection of this particular value in an extensible attribute signals that the extension attribute contains data. Otherwise, this "ext-value" value has no 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, extending the type attribute of the SystemImpact class would look as follows:

```
 <SystemImpact 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".

5.1.2. Public Extension of Enumerated Values

The data model also supports publicly extending select enumerated attributes. A new entry can be added by registering a new entry in the appropriate IANA registry. Section 10.2 provides a mapping between the extensible attributes and their corresponding registry. Section 4.3 discusses the XML Validation implications of this type of extension. All extensible attributes that support private extensions also support public extensions.

5.2. Extending Classes

Classes of the EXTENSION (iodef:ExtensionType) type can extend the data model. They provide the ability to have new atomic or XML-encoded data elements in all of the top-level classes of the Incident class and a few of the complex subordinate classes. As there are multiple instances of the extensible classes in the data model, there is discretion on where to add a new data element. It is RECOMMENDED that the extension be placed in the most closely related class to the new information.

Extensions using the atomic data types (i.e., all values of the dtype attributes other than "xml") MUST:

1. Set the element content to the desired value, and

2. Set the dtype attribute to correspond to the data type of the element content.

The following guidelines exist for extensions using XML (i.e., dtype="xml"):

1. The element content of the extensible class MUST be set to the desired value and the dtype attribute MUST be set to "xml".

2. The extension schema MUST declare a separate namespace. It is RECOMMENDED that these extensions have the prefix "iodef-". This recommendation makes readability of the document easier by allowing the reader to infer which namespaces relate to IODEF by inspection.

3. It is RECOMMENDED that extension schemas follow the naming convention of the IODEF data model. This too improves the readability of extended IODEF documents. The names of all elements SHOULD be capitalized. For elements with composed

names, a capital letter SHOULD be used for each word. Attribute names SHOULD be in lower case. Attributes with composed names SHOULD be separated by a hyphen.

4. Implementations that encounter an unrecognized element, attribute or attribute value in a supported namespace SHOULD reject the document as a syntax error.

5. There are security and performance implications in requiring implementations to dynamically download schemas at run time. Therefore, implementations MUST NOT download schemas at runtime unless the appropriate precautions are taken. Implementations also need to contend with the potential of significant network and processing issues.

6. Some adopters of the IODEF may have private schema definitions that are not publicly available. Thus implementations may encounter IODEF documents with references to private schemas that may not be resolvable. Hence, IODEF document recipients MUST be prepared for a schema definition in an IODEF document never to resolve.

The following schema and XML document excerpt provide a template for an extension schema and its use in the IODEF document.

This example schema defines a namespace of "iodef-extension1" and a single element named "newdata".

```
<xs:schema
   targetNamespace="iodef-extension1.xsd"
   xmlns:iodef-extension1="iodef-extension1.xsd"
   xmlns:xs="http://www.w3.org/2001/XMLSchema">
   attributeFormDefault="unqualified"
   elementFormDefault="qualified">
 <xs:import
      namespace="urn:ietf:params:xml:ns:iodef-2.0"
      schemaLocation=" urn:ietf:params:xml:schema:iodef-2.0"/>

   <xs:element name="newdata" type="xs:string" />
</xs:schema>
```

The following XML excerpt demonstrates the use of the above schema as an extension to the IODEF.

```
      <IODEF-Document
           version="2.00" lang="en-US"
           xmlns="urn:ietf:params:xml:ns:iodef-2.0"
           xmlns:iodef=" urn:ietf:params:xml:ns:iodef-2.0"
           xmlns:iodef-extension1="iodef-extension1.xsd"
           xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
           xsi:schemaLocation="iodef-extension1.xsd">
          <Incident purpose="reporting">
          ...
            <AdditionalData dtype="xml" meaning="xml">
              <iodef-extension1:newdata>
               Field that could not be represented elsewhere
              </iodef-extension1:newdata>
            </AdditionalData>
          </Incident>
    </IODEF-Document
```

5.3. Deconflicting Private Extensions

To disambiguate which private extension is used in an IODEF document, the data model provides a means to identify the source of an extension. Two attributes in the IODEF-Document class, private-enum-name and private-enum-id, are used to specify this attribution. Only a single private extension can be identified in a given IODEF-Document.

If an implementor has a single private extension, then only the private-enum-name attribute needs to be specified. Multiple distinct private extensions or versioning of a single extension can be attributed by also setting the corresponding private-num-id attribute.

The following XML excerpt demonstrates the specification of a private extension from "example.com" with an identifier of "13".

```
      <IODEF-Document
           version="2.00" lang="en-US"
           private-enum-name="example.com"
           private-enum-id="13"
          ...
    </IODEF-Document>
```

If an unrecognized private extension is encountered in processing, the recipient MAY reject the entire document as a syntax error.

6. Internationalization Issues

Internationalization and localization is of specific concern to the IODEF as it facilitates operational coordination with a diverse set of partners. The IODEF implements internationalization by relying on XML constructs and through explicit design choices in the data model.

Since the IODEF is implemented as an XML Schema, it supports different character encodings, such as UTF-8 and UTF-16, possible with XML. Additionally, each IODEF document MUST specify the language in which its content is encoded. The language can be specified with the attribute "xml:lang" (per Section 2.12 of [W3C.XML]) in the top-level element (i.e., IODEF-Document) and letting all other elements inherit that definition. All IODEF classes with a free-form text definition (i.e., all those defined with type iodef:MLStringType) can also specify a language different from the rest of the document.

The data model supports multiple translations of free-form text. All ML_STRING (iodef:MLStringType) classes have a one-to-many cardinality to their parent. This allows the identical text translated into different languages to be encoded in different instances of the same class with a common parent. This design also enables the creation of a single document containing all the translations. The IODEF implementation SHOULD extract the appropriate language relevant to the recipient.

Related instances of a given iodef:MLStringType class that are translations of each other are identified by a common identifier set in the translation-id attribute. The example below shows three instances of a Description class expressed in three different languages. The relationship between these three instances of the Description class is conveyed by the common value of "1" in the translation-id attribute.

```
<IODEF-Document version="2.00" xml:lang="en" ...
  <Incident purpose="reporting">
    ...
    <Description translation-id="1"
                 xml:lang="en">English</Description>
    <Description translation-id="1"
                 xml:lang="de">Englisch</Description>
    <Description translation-id="1"
                 xml:lang="fr">Anglais</Description>
```

The IODEF balances internationalization support with the need for interoperability. While the IODEF supports different languages, the

data model also relies heavily on standardized enumerated attributes that can crudely approximate the contents of the document. With this approach, a CSIRT should be able to make some sense of an IODEF document it receives even if the free-form text data elements are written in a language unfamiliar to the recipient.

7. Examples

This section provides example of IODEF documents. These examples do not represent the full capabilities of the data model or the the only way to encode particular information.

7.1. Minimal Example

A document containing only the mandatory elements and attributes.

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Minimum IODEF document -->
<IODEF-Document version="2.00" xml:lang="en"
   xmlns="urn:ietf:params:xml:ns:iodef-2.0"
   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xsi:schemaLocation=
"http://www.iana.org/assignments/xmlregistry/schema/
iodef-2.0.xsd">
  <Incident purpose="reporting" restriction="private">
    <IncidentID name="csirt.example.com">492382</IncidentID>
    <GenerationTime>2015-07-18T09:00:00-05:00</GenerationTime>
    <Contact type="organization" role="creator">
      <Email>
        <EmailTo>contact@csirt.example.com</EmailTo>
      </Email>
    </Contact>
    <!-- Add more fields to make the document useful -->
  </Incident>
</IODEF-Document>
```

7.2. Indicators from a Campaign

An example of C2 domains from a given campaign.

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- A list of C2 domains associated with a campaign -->
<IODEF-Document version="2.00" xml:lang="en"
   xmlns="urn:ietf:params:xml:ns:iodef-2.0"
   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xsi:schemaLocation=
```

```
      "http://www.iana.org/assignments/xml-registry/schema/
       iodef-2.0.xsd">
     <Incident purpose="watch" restriction="green">
       <IncidentID name="csirt.example.com">897923</IncidentID>
         <RelatedActivity>
           <ThreatActor>
             <ThreatActorID>
             TA-12-AGGRESSIVE-BUTTERFLY
             </ThreatActorID>
             <Description>Aggressive Butterfly</Description>
           </ThreatActor>
           <Campaign>
             <CampaignID>C-2015-59405</CampaignID>
             <Description>Orange Giraffe</Description>
           </Campaign>
         </RelatedActivity>
         <GenerationTime>2015-10-02T11:18:00-05:00</GenerationTime>
         <Description>Summarizes the Indicators of Compromise
           for the Orange Giraffe campaign of the Aggressive
           Butterfly crime gang.
         </Description>
         <Assessment>
           <BusinessImpact type="breach-proprietary"/>
         </Assessment>
         <Contact type="organization" role="creator">
           <ContactName>CSIRT for example.com</ContactName>
           <Email>
             <EmailTo>contact@csirt.example.com</EmailTo>
           </Email>
         </Contact>
         <IndicatorData>
           <Indicator>
             <IndicatorID name="csirt.example.com" version="1">
             G90823490
             </IndicatorID>
             <Description>C2 domains</Description>
             <StartTime>2014-12-02T11:18:00-05:00</StartTime>
             <Observable>
               <BulkObservable type="fqdn">
               <BulkObservableList>
                 kj290023j09r34.example.com
                 09ijk23jfj0k8.example.net
                 klknjwfjiowjefr923.example.org
                 oimireik79msd.example.org
               </BulkObservableList>
             </BulkObservable>
           </Observable>
         </Indicator>
```

```
      </IndicatorData>
    </Incident>
  </IODEF-Document>
```

8. The IODEF Data Model (XML Schema)

```
<?xml version="1.0"?>
<xs:schema xmlns="urn:ietf:params:xml:ns:iodef-2.0"
           xmlns:iodef="urn:ietf:params:xml:ns:iodef-2.0"
           xmlns:enum="urn:ietf:params:xml:ns:iodef-enum-1.0"
           xmlns:sci="urn:ietf:params:xml:ns:iodef-sci-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:iodef-2.0"
           elementFormDefault="qualified"
           attributeFormDefault="unqualified">
  <xs:import namespace="http://www.w3.org/2000/09/xmldsig#"
             schemaLocation="http://www.w3.org/TR/2002/
REC-xmldsig-core-20020212/xmldsig-core-schema.xsd"/>
  <xs:import namespace="urn:ietf:params:xml:ns:iodef-enum-1.0"
             schemaLocation="http://www.iana.org/assignments/
xml-registry/schema/iodef-enum-1.0.xsd"/>
  <xs:import namespace="urn:ietf:params:xml:ns:iodef-sci-1.0"
             schemaLocation="http://www.iana.org/assignments/
xml-registry/schema/iodef-sci-1.0.xsd"/>
  <xs:import namespace="http://www.w3.org/XML/1998/namespace"
             schemaLocation="http://www.w3c.org/2001/xml.xsd"/>
  <xs:annotation>
    <xs:documentation>
       Incident Object Description Exchange Format v2.0
    </xs:documentation>
  </xs:annotation>
  <!--
   ================================================================
   == IODEF-Document class                                       ==
   ================================================================
  -->
  <xs:element name="IODEF-Document">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="iodef:Incident" maxOccurs="unbounded"/>
        <xs:element ref="iodef:AdditionalData"
                    minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:attribute name="version" type="xs:string" fixed="2.00"/>
      <xs:attribute ref="xml:lang"/>
      <xs:attribute name="format-id" type="xs:string" use="optional"/>
      <xs:attribute name="private-enum-name"
```

```
                    type="xs:string" use="optional"/>
      <xs:attribute name="private-enum-id"
                    type="xs:string" use="optional"/>
    </xs:complexType>
  </xs:element>
  <!--
   ================================================================
   == Incident class                                             ==
   ================================================================
  -->
  <xs:element name="Incident">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="iodef:IncidentID"/>
        <xs:element ref="iodef:AlternativeID" minOccurs="0"/>
        <xs:element ref="iodef:RelatedActivity"
                    minOccurs="0" maxOccurs="unbounded"/>
        <xs:element ref="iodef:DetectTime" minOccurs="0"/>
        <xs:element ref="iodef:StartTime" minOccurs="0"/>
        <xs:element ref="iodef:EndTime" minOccurs="0"/>
        <xs:element ref="iodef:RecoveryTime" minOccurs="0"/>
        <xs:element ref="iodef:ReportTime" minOccurs="0"/>
        <xs:element ref="iodef:GenerationTime"/>
        <xs:element ref="iodef:Description"
                    minOccurs="0" maxOccurs="unbounded"/>
        <xs:element ref="iodef:Discovery"
                    minOccurs="0" maxOccurs="unbounded"/>
        <xs:element ref="iodef:Assessment"
                    minOccurs="0" maxOccurs="unbounded"/>
        <xs:element ref="iodef:Method"
                    minOccurs="0" maxOccurs="unbounded"/>
        <xs:element ref="iodef:Contact" maxOccurs="unbounded"/>
        <xs:element ref="iodef:EventData"
                    minOccurs="0" maxOccurs="unbounded"/>
        <xs:element ref="iodef:IndicatorData" minOccurs="0"/>
        <xs:element ref="iodef:History" minOccurs="0"/>
        <xs:element ref="iodef:AdditionalData"
                    minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:attribute name="purpose"
                    type="incident-purpose-type" use="required"/>
      <xs:attribute name="ext-purpose"
                    type="xs:string" use="optional"/>
      <xs:attribute name="status" type="incident-status-type"/>
      <xs:attribute name="ext-status"
                    type="xs:string" use="optional"/>
      <xs:attribute ref="xml:lang"/>
      <xs:attribute name="restriction"
```

```
                    type="iodef:restriction-type" default="private"
                    use="optional"/>
      <xs:attribute name="ext-restriction"
                    type="xs:string" use="optional"/>
      <xs:attribute name="observable-id" type="xs:ID" use="optional"/>
    </xs:complexType>
  </xs:element>
  <xs:simpleType name="incident-purpose-type">
    <xs:restriction base="xs:NMTOKEN">
      <xs:enumeration value="traceback"/>
      <xs:enumeration value="mitigation"/>
      <xs:enumeration value="reporting"/>
      <xs:enumeration value="watch"/>
      <xs:enumeration value="other"/>
      <xs:enumeration value="ext-value"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="incident-status-type">
    <xs:restriction base="xs:NMTOKEN">
      <xs:enumeration value="new"/>
      <xs:enumeration value="in-progress"/>
      <xs:enumeration value="forwarded"/>
      <xs:enumeration value="resolved"/>
      <xs:enumeration value="future"/>
      <xs:enumeration value="ext-value"/>
    </xs:restriction>
  </xs:simpleType>
  <!--
   ========================================================================
   ==  IncidentID class                                                  ==
   ========================================================================
  -->
  <xs:element name="IncidentID" type="iodef:IncidentIDType"/>
  <xs:complexType name="IncidentIDType">
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute name="name" type="xs:string" use="required"/>
        <xs:attribute name="instance"
                      type="xs:string" use="optional"/>
        <xs:attribute name="restriction"
                      type="iodef:restriction-type" use="optional"/>
        <xs:attribute name="ext-restriction"
                      type="xs:string" use="optional"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
  <!--
   ========================================================================
```

```
     ==  AlternativeID class                                          ==
     ======================================================================
    -->
    <xs:element name="AlternativeID">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:IncidentID" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="restriction"
                      type="iodef:restriction-type" use="optional"/>
        <xs:attribute name="ext-restriction"
                      type="xs:string" use="optional"/>
      </xs:complexType>
    </xs:element>
    <!--
     ======================================================================
     ==  RelatedActivity class                                           ==
     ======================================================================
    -->
    <xs:element name="RelatedActivity">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:IncidentID"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:URL"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:ThreatActor"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:Campaign"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:IndicatorID"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:Confidence" minOccurs="0"/>
          <xs:element ref="iodef:Description"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:AdditionalData"
                      minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="restriction"
                      type="iodef:restriction-type" use="optional"/>
        <xs:attribute name="ext-restriction"
                      type="xs:string" use="optional"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="ThreatActor">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:ThreatActorID"
```

```
                    minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:URL" maxOccurs="unbounded"/>
          <xs:element ref="iodef:Description"
                    minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:AdditionalData"
                    minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="restriction"
                    type="iodef:restriction-type" use="optional"/>
        <xs:attribute name="ext-restriction"
                    type="xs:string" use="optional"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="ThreatActorID" type="xs:string"/>
    <xs:element name="Campaign">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:CampaignID"
                    minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:URL"
                    minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:Description"
                    minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:AdditionalData"
                    minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="restriction"
                    type="iodef:restriction-type" use="optional"/>
        <xs:attribute name="ext-restriction"
                    type="xs:string" use="optional"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="CampaignID" type="xs:string"/>
    <!--
     ========================================================================
     ==   Contact class                                                    ==
     ========================================================================
    -->
    <xs:element name="Contact">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:ContactName"
                    minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:ContactTitle"
                    minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:Description"
                    minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:RegistryHandle"
```

```
                    minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:PostalAddress"
                    minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:Email"
                    minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:Telephone"
                    minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:Timezone" minOccurs="0"/>
          <xs:element ref="iodef:Contact"
                    minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:AdditionalData"
                    minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="role"
                    type="contact-role-type" use="required"/>
        <xs:attribute name="ext-role"
                    type="xs:string" use="optional"/>
        <xs:attribute name="type"
                    type="contact-type-type" use="required"/>
        <xs:attribute name="ext-type"
                    type="xs:string" use="optional"/>
        <xs:attribute name="restriction"
                    type="iodef:restriction-type" use="optional"/>
        <xs:attribute name="ext-restriction"
                    type="xs:string" use="optional"/>
      </xs:complexType>
    </xs:element>
    <xs:simpleType name="contact-role-type">
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="creator"/>
        <xs:enumeration value="reporter"/>
        <xs:enumeration value="admin"/>
        <xs:enumeration value="tech"/>
        <xs:enumeration value="provider"/>
        <xs:enumeration value="user"/>
        <xs:enumeration value="billing"/>
        <xs:enumeration value="legal"/>
        <xs:enumeration value="abuse"/>
        <xs:enumeration value="irt"/>
        <xs:enumeration value="cc"/>
        <xs:enumeration value="cc-irt"/>
        <xs:enumeration value="leo"/>
        <xs:enumeration value="vendor"/>
        <xs:enumeration value="vendor-services"/>
        <xs:enumeration value="victim"/>
        <xs:enumeration value="victim-notified"/>
        <xs:enumeration value="ext-value"/>
      </xs:restriction>
```

```
    </xs:simpleType>
    <xs:simpleType name="contact-type-type">
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="person"/>
        <xs:enumeration value="organization"/>
        <xs:enumeration value="ext-value"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:element name="ContactName" type="iodef:MLStringType"/>
    <xs:element name="ContactTitle" type="iodef:MLStringType"/>
    <xs:element name="RegistryHandle">
      <xs:complexType>
        <xs:simpleContent>
          <xs:extension base="xs:string">
            <xs:attribute name="registry"
                          type="registryhandle-registry-type"/>
            <xs:attribute name="ext-registry"
                          type="xs:string" use="optional"/>
          </xs:extension>
        </xs:simpleContent>
      </xs:complexType>
    </xs:element>
    <xs:simpleType name="registryhandle-registry-type">
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="internic"/>
        <xs:enumeration value="apnic"/>
        <xs:enumeration value="arin"/>
        <xs:enumeration value="lacnic"/>
        <xs:enumeration value="ripe"/>
        <xs:enumeration value="afrinic"/>
        <xs:enumeration value="local"/>
        <xs:enumeration value="ext-value"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:element name="PostalAddress">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:PAddress"/>
          <xs:element ref="iodef:Description"
                      minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="type"
                      type="postaladdress-type-type" use="optional"/>
        <xs:attribute name="ext-type" type="xs:string" use="optional"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="PAddress" type="iodef:MLStringType"/>
    <xs:simpleType name="postaladdress-type-type">
```

```
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="street"/>
        <xs:enumeration value="mailing"/>
        <xs:enumeration value="ext-value"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:element name="Telephone">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:TelephoneNumber"/>
          <xs:element ref="iodef:Description"
                      minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="type"
                      type="telephone-type-type" use="optional"/>
        <xs:attribute name="ext-type" type="xs:string" use="optional"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="TelephoneNumber" type="xs:string"/>
    <xs:simpleType name="telephone-type-type">
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="wired"/>
        <xs:enumeration value="mobile"/>
        <xs:enumeration value="fax"/>
        <xs:enumeration value="hotline"/>
        <xs:enumeration value="ext-value"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:element name="Email">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:EmailTo"/>
          <xs:element ref="iodef:Description"
                      minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="type"
                      type="email-type-type" use="optional"/>
        <xs:attribute name="ext-type" type="xs:string" use="optional"/>
      </xs:complexType>
    </xs:element>
    <xs:simpleType name="email-type-type">
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="direct"/>
        <xs:enumeration value="hotline"/>
        <xs:enumeration value="ext-value"/>
      </xs:restriction>
    </xs:simpleType>
    <!--
```

```
     ============================================================
     ==  Time-based classes                                    ==
     ============================================================
    -->
    <xs:element name="DateTime" type="xs:dateTime"/>
    <xs:element name="ReportTime" type="xs:dateTime"/>
    <xs:element name="DetectTime" type="xs:dateTime"/>
    <xs:element name="StartTime" type="xs:dateTime"/>
    <xs:element name="EndTime" type="xs:dateTime"/>
    <xs:element name="RecoveryTime" type="xs:dateTime"/>
    <xs:element name="GenerationTime" type="xs:dateTime"/>
    <xs:element name="Timezone" type="iodef:TimezoneType"/>
    <!--
     ============================================================
     ==  History class                                         ==
     ============================================================
    -->
    <xs:element name="History">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:HistoryItem" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="restriction"
                      type="iodef:restriction-type" use="optional"/>
        <xs:attribute name="ext-restriction"
                      type="xs:string" use="optional"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="HistoryItem">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:DateTime"/>
          <xs:element ref="iodef:IncidentID" minOccurs="0"/>
          <xs:element ref="iodef:Contact" minOccurs="0"/>
          <xs:element ref="iodef:Description"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:DefinedCOA"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:AdditionalData"
                      minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="action"
                      type="iodef:action-type" use="required"/>
        <xs:attribute name="ext-action"
                      type="xs:string" use="optional"/>
        <xs:attribute name="restriction"
                      type="iodef:restriction-type" use="optional"/>
        <xs:attribute name="ext-restriction"
```

```
                    type="xs:string" use="optional"/>
        <xs:attribute name="observable-id" type="xs:ID" use="optional"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="DefinedCOA" type="xs:string"/>
    <!--
     ========================================================================
     ==  Expectation class                                                 ==
     ========================================================================
    -->
    <xs:element name="Expectation">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:Description"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:DefinedCOA"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:StartTime" minOccurs="0"/>
          <xs:element ref="iodef:EndTime" minOccurs="0"/>
          <xs:element ref="iodef:Contact" minOccurs="0"/>
        </xs:sequence>
        <xs:attribute name="action"
                      type="iodef:action-type" default="other"/>
        <xs:attribute name="ext-action"
                      type="xs:string" use="optional"/>
        <xs:attribute name="severity" type="iodef:severity-type"/>
        <xs:attribute name="restriction"
                      type="iodef:restriction-type" use="optional"/>
        <xs:attribute name="ext-restriction"
                      type="xs:string" use="optional"/>
        <xs:attribute name="observable-id" type="xs:ID" use="optional"/>
      </xs:complexType>
    </xs:element>
    <!--
     ========================================================================
     ==  Discovery class                                                   ==
     ========================================================================
    -->
    <xs:element name="Discovery">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:Description"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:Contact"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:DetectionPattern"
                      minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
```

```
        <xs:attribute name="source"
                      type="discovery-source-type" use="optional"
                      default="unknown"/>
        <xs:attribute name="ext-source"
                      type="xs:string" use="optional"/>
        <xs:attribute name="restriction"
                      type="iodef:restriction-type" use="optional"/>
        <xs:attribute name="ext-restriction"
                      type="xs:string" use="optional"/>
      </xs:complexType>
    </xs:element>
    <xs:simpleType name="discovery-source-type">
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="nidps"/>
        <xs:enumeration value="hips"/>
        <xs:enumeration value="siem"/>
        <xs:enumeration value="av"/>
        <xs:enumeration value="third-party-monitoring"/>
        <xs:enumeration value="incident"/>
        <xs:enumeration value="os-log"/>
        <xs:enumeration value="application-log"/>
        <xs:enumeration value="device-log"/>
        <xs:enumeration value="network-flow"/>
        <xs:enumeration value="passive-dns"/>
        <xs:enumeration value="investigation"/>
        <xs:enumeration value="audit"/>
        <xs:enumeration value="internal-notification"/>
        <xs:enumeration value="external-notification"/>
        <xs:enumeration value="leo"/>
        <xs:enumeration value="partner"/>
        <xs:enumeration value="actor"/>
        <xs:enumeration value="unknown"/>
        <xs:enumeration value="ext-value"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:element name="DetectionPattern">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:Application"/>
          <xs:element ref="iodef:Description"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element name="DetectionConfiguration"
                      type="xs:string"
                      minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="restriction"
                      type="iodef:restriction-type" use="optional"/>
        <xs:attribute name="ext-restriction"
```

```
                    type="xs:string" use="optional"/>
        <xs:attribute name="observable-id" type="xs:ID" use="optional"/>
      </xs:complexType>
    </xs:element>
    <!--
     ================================================================
     ==  Method class                                               ==
     ================================================================
    -->
    <xs:element name="Method">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:Reference"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:Description"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="sci:AttackPattern"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="sci:Vulnerability"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="sci:Weakness"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:AdditionalData"
                      minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="restriction"
                      type="iodef:restriction-type" use="optional"/>
        <xs:attribute name="ext-restriction"
                      type="xs:string" use="optional"/>
      </xs:complexType>
    </xs:element>
    <!--
     ================================================================
     ==  Reference class                                            ==
     ================================================================
    -->
    <xs:element name="Reference">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="enum:ReferenceName" minOccurs="0"/>
          <xs:element ref="iodef:URL"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:Description"
                      minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="observable-id" type="xs:ID" use="optional"/>
      </xs:complexType>
    </xs:element>
```

```
    <!--
     ========================================================================
     ==  Assessment class                                                  ==
     ========================================================================
    -->
    <xs:element name="Assessment">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:IncidentCategory"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:choice maxOccurs="unbounded">
            <xs:element ref="iodef:SystemImpact"/>
            <xs:element ref="iodef:BusinessImpact"/>
            <xs:element ref="iodef:TimeImpact"/>
            <xs:element ref="iodef:MonetaryImpact"/>
            <xs:element ref="iodef:IntendedImpact"/>
          </xs:choice>
          <xs:element ref="iodef:Counter"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:MitigatingFactor"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:Cause"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:Confidence" minOccurs="0"/>
          <xs:element ref="iodef:AdditionalData"
                      minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="occurrence">
          <xs:simpleType>
            <xs:restriction base="xs:NMTOKEN">
              <xs:enumeration value="actual"/>
              <xs:enumeration value="potential"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:attribute>
        <xs:attribute name="restriction"
                      type="iodef:restriction-type" use="optional"/>
        <xs:attribute name="ext-restriction"
                      type="xs:string" use="optional"/>
        <xs:attribute name="observable-id" type="xs:ID" use="optional"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="IncidentCategory" type="iodef:MLStringType"/>
    <xs:element name="BusinessImpact" type="iodef:BusinessImpactType"/>
    <xs:element name="IntendedImpact" type="iodef:BusinessImpactType"/>
    <xs:element name="MitigatingFactor" type="iodef:MLStringType"/>
    <xs:element name="Cause" type="iodef:MLStringType"/>
    <xs:element name="SystemImpact">
```

```
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:Description"
                      minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="severity"
                      type="iodef:severity-type" use="optional"/>
        <xs:attribute name="completion"
                      type="iodef:systemimpact-completion-type"
                      use="optional"/>
        <xs:attribute name="type"
                      type="systemimpact-type-type"
                      use="optional" default="unknown"/>
        <xs:attribute name="ext-type" type="xs:string" use="optional"/>
      </xs:complexType>
    </xs:element>
    <xs:simpleType name="systemimpact-completion-type">
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="failed"/>
        <xs:enumeration value="succeeded"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="systemimpact-type-type">
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="takeover-account"/>
        <xs:enumeration value="takeover-service"/>
        <xs:enumeration value="takeover-system"/>
        <xs:enumeration value="cps-manipulation"/>
        <xs:enumeration value="cps-damage"/>
        <xs:enumeration value="availability-data"/>
        <xs:enumeration value="availability-account"/>
        <xs:enumeration value="availability-service"/>
        <xs:enumeration value="availability-system"/>
        <xs:enumeration value="damaged-system"/>
        <xs:enumeration value="damaged-data"/>
        <xs:enumeration value="breach-proprietary"/>
        <xs:enumeration value="breach-privacy"/>
        <xs:enumeration value="breach-credential"/>
        <xs:enumeration value="breach-configuration"/>
        <xs:enumeration value="integrity-data"/>
        <xs:enumeration value="integrity-configuration"/>
        <xs:enumeration value="integrity-hardware"/>
        <xs:enumeration value="traffic-redirection"/>
        <xs:enumeration value="monitoring-traffic"/>
        <xs:enumeration value="monitoring-host"/>
        <xs:enumeration value="policy"/>
        <xs:enumeration value="unknown"/>
        <xs:enumeration value="ext-value"/>
```

```
      </xs:restriction>
    </xs:simpleType>
    <xs:complexType name="BusinessImpactType">
      <xs:sequence>
        <xs:element ref="iodef:Description"
                    minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:attribute name="severity"
                    type="businessimpact-severity-type" use="optional"/>
      <xs:attribute name="ext-severity"
                    type="xs:string" use="optional"/>
      <xs:attribute name="type"
                    type="businessimpact-type-type"
                    use="optional" default="unknown"/>
      <xs:attribute name="ext-type" type="xs:string" use="optional"/>
    </xs:complexType>
    <xs:simpleType name="businessimpact-severity-type">
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="none"/>
        <xs:enumeration value="low"/>
        <xs:enumeration value="medium"/>
        <xs:enumeration value="high"/>
        <xs:enumeration value="unknown"/>
        <xs:enumeration value="ext-value"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="businessimpact-type-type">
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="breach-proprietary"/>
        <xs:enumeration value="breach-privacy"/>
        <xs:enumeration value="breach-credential"/>
        <xs:enumeration value="loss-of-integrity"/>
        <xs:enumeration value="loss-of-service"/>
        <xs:enumeration value="theft-financial"/>
        <xs:enumeration value="theft-service"/>
        <xs:enumeration value="degraded-reputation"/>
        <xs:enumeration value="asset-damage"/>
        <xs:enumeration value="asset-manipulation"/>
        <xs:enumeration value="legal"/>
        <xs:enumeration value="extortion"/>
        <xs:enumeration value="unknown"/>
        <xs:enumeration value="ext-value"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:element name="TimeImpact">
      <xs:complexType>
        <xs:simpleContent>
          <xs:extension base="iodef:PositiveFloatType">
```

```
            <xs:attribute name="severity" type="iodef:severity-type"/>
            <xs:attribute name="metric"
                          type="timeimpact-metric-type" use="required"/>
            <xs:attribute name="ext-metric"
                          type="xs:string" use="optional"/>
            <xs:attribute name="duration" type="iodef:duration-type"/>
            <xs:attribute name="ext-duration"
                          type="xs:string" use="optional"/>
          </xs:extension>
        </xs:simpleContent>
      </xs:complexType>
    </xs:element>
    <xs:simpleType name="timeimpact-metric-type">
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="labor"/>
        <xs:enumeration value="elapsed"/>
        <xs:enumeration value="downtime"/>
        <xs:enumeration value="ext-value"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:element name="MonetaryImpact">
      <xs:complexType>
        <xs:simpleContent>
          <xs:extension base="iodef:PositiveFloatType">
            <xs:attribute name="severity" type="iodef:severity-type"/>
            <xs:attribute name="currency" type="xs:string"/>
          </xs:extension>
        </xs:simpleContent>
      </xs:complexType>
    </xs:element>
    <xs:element name="Confidence">
      <xs:complexType>
        <xs:attribute name="rating"
                      type="confidence-rating-type" use="required"/>
        <xs:attribute name="ext-rating"
                      type="xs:string" use="optional"/>
      </xs:complexType>
    </xs:element>
    <xs:simpleType name="confidence-rating-type">
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="low"/>
        <xs:enumeration value="medium"/>
        <xs:enumeration value="high"/>
        <xs:enumeration value="numeric"/>
        <xs:enumeration value="unknown"/>
        <xs:enumeration value="ext-value"/>
      </xs:restriction>
    </xs:simpleType>
```

```
    <!--
     ========================================================================
     == EventData class                                                    ==
     ========================================================================
    -->
    <xs:element name="EventData">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:Description"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:DetectTime" minOccurs="0"/>
          <xs:element ref="iodef:StartTime" minOccurs="0"/>
          <xs:element ref="iodef:EndTime" minOccurs="0"/>
          <xs:element ref="iodef:RecoveryTime" minOccurs="0"/>
          <xs:element ref="iodef:ReportTime" minOccurs="0"/>
          <xs:element ref="iodef:Contact"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:Discovery"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:Assessment" minOccurs="0"/>
          <xs:element ref="iodef:Method"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:Flow"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:Expectation"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:Record" minOccurs="0"/>
          <xs:element ref="iodef:EventData"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:AdditionalData"
                      minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="restriction"
                      type="iodef:restriction-type" use="optional"/>
        <xs:attribute name="ext-restriction"
                      type="xs:string" use="optional"/>
        <xs:attribute name="observable-id" type="xs:ID" use="optional"/>
      </xs:complexType>
    </xs:element>
    <!--
     ========================================================================
     ==  Flow class                                                        ==
     ========================================================================
    -->
    <xs:element name="Flow">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:System" maxOccurs="unbounded"/>
```

```
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!--
   ================================================================
   ==  System class                                               ==
   ================================================================
  -->
  <xs:element name="System">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="iodef:Node"/>
        <xs:element ref="iodef:NodeRole"
                    minOccurs="0" maxOccurs="unbounded"/>
        <xs:element ref="iodef:Service"
                    minOccurs="0" maxOccurs="unbounded"/>
        <xs:element ref="iodef:OperatingSystem"
                    minOccurs="0" maxOccurs="unbounded"/>
        <xs:element ref="iodef:Counter"
                    minOccurs="0" maxOccurs="unbounded"/>
        <xs:element name="AssetID"
                    type="xs:string"
                    minOccurs="0" maxOccurs="unbounded"/>
        <xs:element ref="iodef:Description"
                    minOccurs="0" maxOccurs="unbounded"/>
        <xs:element ref="iodef:AdditionalData"
                    minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:attribute name="category" type="system-category-type"/>
      <xs:attribute name="ext-category"
                    type="xs:string" use="optional"/>
      <xs:attribute name="interface" type="xs:string"/>
      <xs:attribute name="spoofed"
                    type="yes-no-unknown-type" default="unknown"/>
      <xs:attribute name="virtual"
                    type="yes-no-unknown-type" use="optional"
                    default="unknown"/>
      <xs:attribute name="ownership" type="system-ownership-type"
                    use="optional"/>
      <xs:attribute name="ext-ownership"
                    type="xs:string" use="optional"/>
      <xs:attribute name="restriction"
                    type="iodef:restriction-type" use="optional"/>
      <xs:attribute name="ext-restriction"
                    type="xs:string" use="optional"/>
      <xs:attribute name="observable-id" type="xs:ID" use="optional"/>
    </xs:complexType>
  </xs:element>
```

```
    <xs:element name="OperatingSystem" type="iodef:SoftwareType"/>
    <xs:simpleType name="system-category-type">
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="source"/>
        <xs:enumeration value="target"/>
        <xs:enumeration value="intermediate"/>
        <xs:enumeration value="sensor"/>
        <xs:enumeration value="infrastructure"/>
        <xs:enumeration value="ext-value"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="system-ownership-type">
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="organization"/>
        <xs:enumeration value="personal"/>
        <xs:enumeration value="partner"/>
        <xs:enumeration value="customer"/>
        <xs:enumeration value="no-relationship"/>
        <xs:enumeration value="unknown"/>
        <xs:enumeration value="ext-value"/>
      </xs:restriction>
    </xs:simpleType>
    <!--
     ================================================================
     == Node class                                                 ==
     ================================================================
    -->
    <xs:element name="Node">
      <xs:complexType>
        <xs:sequence>
          <xs:choice maxOccurs="unbounded">
            <xs:element ref="iodef:DomainData"
                        minOccurs="0" maxOccurs="unbounded"/>
            <xs:element ref="iodef:Address"
                        minOccurs="0" maxOccurs="unbounded"/>
          </xs:choice>
          <xs:element ref="iodef:PostalAddress" minOccurs="0"/>
          <xs:element ref="iodef:Location"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:Counter"
                      minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="Address">
      <xs:complexType>
        <xs:simpleContent>
          <xs:extension base="xs:string">
```

```
            <xs:attribute name="category"
                          type="address-category-type"
                          default="ipv6-addr"/>
            <xs:attribute name="ext-category"
                          type="xs:string" use="optional"/>
            <xs:attribute name="vlan-name" type="xs:string"/>
            <xs:attribute name="vlan-num" type="xs:integer"/>
            <xs:attribute name="observable-id"
                          type="xs:ID" use="optional"/>
          </xs:extension>
        </xs:simpleContent>
      </xs:complexType>
    </xs:element>
    <xs:simpleType name="address-category-type">
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="asn"/>
        <xs:enumeration value="atm"/>
        <xs:enumeration value="e-mail"/>
        <xs:enumeration value="mac"/>
        <xs:enumeration value="ipv4-addr"/>
        <xs:enumeration value="ipv4-net"/>
        <xs:enumeration value="ipv4-net-masked"/>
        <xs:enumeration value="ipv4-net-mask"/>
        <xs:enumeration value="ipv6-addr"/>
        <xs:enumeration value="ipv6-net"/>
        <xs:enumeration value="ipv6-net-masked"/>
        <xs:enumeration value="site-uri"/>
        <xs:enumeration value="ext-value"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:element name="Location" type="iodef:MLStringType"/>
    <xs:element name="NodeRole">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:Description"
                      minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="category"
                      type="noderole-category-type" use="required"/>
        <xs:attribute name="ext-category"
                      type="xs:string" use="optional"/>
      </xs:complexType>
    </xs:element>
    <xs:simpleType name="noderole-category-type">
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="client"/>
        <xs:enumeration value="client-enterprise"/>
        <xs:enumeration value="client-partner"/>
```

```
        <xs:enumeration value="client-remote"/>
        <xs:enumeration value="client-kiosk"/>
        <xs:enumeration value="client-mobile"/>
        <xs:enumeration value="server-internal"/>
        <xs:enumeration value="server-public"/>
        <xs:enumeration value="www"/>
        <xs:enumeration value="mail"/>
        <xs:enumeration value="webmail"/>
        <xs:enumeration value="messaging"/>
        <xs:enumeration value="streaming"/>
        <xs:enumeration value="voice"/>
        <xs:enumeration value="file"/>
        <xs:enumeration value="ftp"/>
        <xs:enumeration value="p2p"/>
        <xs:enumeration value="name"/>
        <xs:enumeration value="directory"/>
        <xs:enumeration value="credential"/>
        <xs:enumeration value="print"/>
        <xs:enumeration value="application"/>
        <xs:enumeration value="database"/>
        <xs:enumeration value="backup"/>
        <xs:enumeration value="dhcp"/>
        <xs:enumeration value="assessment"/>
        <xs:enumeration value="source-control"/>
        <xs:enumeration value="config-management"/>
        <xs:enumeration value="monitoring"/>
        <xs:enumeration value="infra"/>
        <xs:enumeration value="infra-firewall"/>
        <xs:enumeration value="infra-router"/>
        <xs:enumeration value="infra-switch"/>
        <xs:enumeration value="camera"/>
        <xs:enumeration value="proxy"/>
        <xs:enumeration value="remote-access"/>
        <xs:enumeration value="log"/>
        <xs:enumeration value="virtualization"/>
        <xs:enumeration value="pos"/>
        <xs:enumeration value="scada"/>
        <xs:enumeration value="scada-supervisory"/>
        <xs:enumeration value="sinkhole"/>
        <xs:enumeration value="honeypot"/>
        <xs:enumeration value="anonymization"/>
        <xs:enumeration value="c2-server"/>
        <xs:enumeration value="malware-distribution"/>
        <xs:enumeration value="drop-server"/>
        <xs:enumeration value="hop-point"/>
        <xs:enumeration value="reflector"/>
        <xs:enumeration value="phishing-site"/>
        <xs:enumeration value="spear-phishing-site"/>
```

```
        <xs:enumeration value="recruiting-site"/>
        <xs:enumeration value="fraudulent-site"/>
        <xs:enumeration value="ext-value"/>
      </xs:restriction>
    </xs:simpleType>
    <!--
     ========================================================================
     ==  Service Class                                                     ==
     ========================================================================
    -->
    <xs:element name="Service">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:ServiceName" minOccurs="0"/>
          <xs:element ref="iodef:Port" minOccurs="0"/>
          <xs:element ref="iodef:Portlist" minOccurs="0"/>
          <xs:element ref="iodef:ProtoType" minOccurs="0"/>
          <xs:element ref="iodef:ProtoCode" minOccurs="0"/>
          <xs:element ref="iodef:ProtoField" minOccurs="0"/>
          <xs:element ref="iodef:ApplicationHeader" minOccurs="0"/>
          <xs:element ref="iodef:EmailData" minOccurs="0"/>
          <xs:element ref="iodef:Application" minOccurs="0"/>
        </xs:sequence>
        <xs:attribute name="ip-protocol"
                      type="xs:integer" use="optional"/>
        <xs:attribute name="observable-id" type="xs:ID" use="optional"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="Port" type="xs:integer"/>
    <xs:element name="Portlist" type="iodef:PortlistType"/>
    <xs:element name="ProtoType" type="xs:integer"/>
    <xs:element name="ProtoCode" type="xs:integer"/>
    <xs:element name="ProtoField" type="xs:integer"/>
    <xs:element name="ApplicationHeader">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:ApplicationHeaderField"
                      maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="ApplicationHeaderField"
                type="iodef:ExtensionType"/>
    <xs:element name="ServiceName">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:IANAService"
                      minOccurs="0"/>
```

```
          <xs:element ref="iodef:URL"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:Description"
                      minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="IANAService" type="xs:string"/>
    <xs:element name="Application" type="iodef:SoftwareType"/>
    <!--
     ========================================================================
     ==  Counter class                                                     ==
     ========================================================================
    -->
    <xs:element name="Counter">
      <xs:complexType>
        <xs:simpleContent>
          <xs:extension base="xs:float">
            <xs:attribute name="type"
                          type="counter-type-type" use="required"/>
            <xs:attribute name="ext-type"
                          type="xs:string" use="optional"/>
            <xs:attribute name="unit"
                          type="counter-unit-type" use="required"/>
            <xs:attribute name="ext-unit"
                          type="xs:string" use="optional"/>
            <xs:attribute name="meaning"
                          type="xs:string" use="optional"/>
            <xs:attribute name="duration" type="iodef:duration-type"/>
            <xs:attribute name="ext-duration"
                          type="xs:string" use="optional"/>
          </xs:extension>
        </xs:simpleContent>
      </xs:complexType>
    </xs:element>
    <xs:simpleType name="counter-type-type">
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="counter"/>
        <xs:enumeration value="rate"/>
        <xs:enumeration value="average"/>
        <xs:enumeration value="ext-value"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="counter-unit-type">
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="byte"/>
        <xs:enumeration value="mbit"/>
        <xs:enumeration value="packet"/>
```

```
        <xs:enumeration value="flow"/>
        <xs:enumeration value="session"/>
        <xs:enumeration value="event"/>
        <xs:enumeration value="alert"/>
        <xs:enumeration value="message"/>
        <xs:enumeration value="host"/>
        <xs:enumeration value="site"/>
        <xs:enumeration value="organization"/>
        <xs:enumeration value="ext-value"/>
      </xs:restriction>
    </xs:simpleType>
    <!--
     ================================================================
     ==  EmailData class                                           ==
     ================================================================
    -->
    <xs:element name="EmailData">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:EmailTo"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:EmailFrom" minOccurs="0"/>
          <xs:element ref="iodef:EmailSubject" minOccurs="0"/>
          <xs:element ref="iodef:EmailX-Mailer" minOccurs="0"/>
          <xs:element ref="iodef:EmailHeaderField"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:EmailHeaders" minOccurs="0"/>
          <xs:element ref="iodef:EmailBody" minOccurs="0"/>
          <xs:element ref="iodef:EmailMessage" minOccurs="0"/>
          <xs:element ref="iodef:HashData"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="SignatureData"
                      minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="observable-id" type="xs:ID" use="optional"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="EmailTo" type="xs:string"/>
    <xs:element name="EmailFrom" type="xs:string"/>
    <xs:element name="EmailSubject" type="xs:string"/>
    <xs:element name="EmailX-Mailer" type="xs:string"/>
    <xs:element name="EmailHeaderField" type="iodef:ExtensionType"/>
    <xs:element name="EmailHeaders" type="xs:string"/>
    <xs:element name="EmailBody" type="xs:string"/>
    <xs:element name="EmailMessage" type="xs:string"/>
    <!--
     ================================================================
     ==   DomainData class                                         ==
```

```
     ============================================================================
    -->
    <xs:element name="DomainData">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:Name"/>
          <xs:element ref="iodef:DateDomainWasChecked"
                      minOccurs="0"/>
          <xs:element ref="iodef:RegistrationDate"
                      minOccurs="0"/>
          <xs:element ref="iodef:ExpirationDate"
                      minOccurs="0"/>
          <xs:element ref="iodef:RelatedDNS"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:Nameservers"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:DomainContacts"
                      minOccurs="0"/>
        </xs:sequence>
        <xs:attribute name="system-status"
                      type="domaindata-system-status-type"/>
        <xs:attribute name="ext-system-status"
                      type="xs:string" use="optional"/>
        <xs:attribute name="domain-status"
                      type="domaindata-domain-status-type"/>
        <xs:attribute name="ext-domain-status"
                      type="xs:string" use="optional"/>
        <xs:attribute name="observable-id" type="xs:ID" use="optional"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="Name" type="xs:string"/>
    <xs:element name="DateDomainWasChecked" type="xs:dateTime"/>
    <xs:element name="RegistrationDate" type="xs:dateTime"/>
    <xs:element name="ExpirationDate" type="xs:dateTime"/>
    <xs:simpleType name="domaindata-system-status-type">
      <xs:restriction base="xs:string">
        <xs:enumeration value="spoofed"/>
        <xs:enumeration value="fraudulent"/>
        <xs:enumeration value="innocent-hacked"/>
        <xs:enumeration value="innocent-hijacked"/>
        <xs:enumeration value="unknown"/>
        <xs:enumeration value="ext-value"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="domaindata-domain-status-type">
      <xs:restriction base="xs:string">
        <xs:enumeration value="reservedDelegation"/>
        <xs:enumeration value="assignedAndActive"/>
```

```
        <xs:enumeration value="assignedAndInactive"/>
        <xs:enumeration value="assignedAndOnHold"/>
        <xs:enumeration value="revoked"/>
        <xs:enumeration value="transferPending"/>
        <xs:enumeration value="registryLock"/>
        <xs:enumeration value="registrarLock"/>
        <xs:enumeration value="other"/>
        <xs:enumeration value="unknown"/>
        <xs:enumeration value="ext-value"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:element name="RelatedDNS" type="iodef:ExtensionType"/>
    <xs:element name="Nameservers">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:Server"/>
          <xs:element ref="iodef:Address" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="Server" type="xs:string"/>
    <xs:element name="DomainContacts">
      <xs:complexType>
        <xs:choice>
          <xs:element ref="iodef:SameDomainContact"/>
          <xs:element ref="iodef:Contact"
                      minOccurs="1" maxOccurs="unbounded"/>
        </xs:choice>
      </xs:complexType>
    </xs:element>
    <xs:element name="SameDomainContact" type="xs:string"/>
    <!--
     ========================================================================
     ==  Record class                                                      ==
     ========================================================================
    -->
    <xs:element name="Record">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:RecordData" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="restriction"
                      type="iodef:restriction-type" use="optional"/>
        <xs:attribute name="ext-restriction"
                      type="xs:string" use="optional"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="RecordData">
```

```
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:DateTime" minOccurs="0"/>
          <xs:element ref="iodef:Description"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:Application" minOccurs="0"/>
          <xs:element ref="iodef:RecordPattern"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:RecordItem"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:URL"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:FileData"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:WindowsRegistryKeysModified"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:CertificateData"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:AdditionalData"
                      minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="restriction"
                      type="iodef:restriction-type" use="optional"/>
        <xs:attribute name="ext-restriction"
                      type="xs:string" use="optional"/>
        <xs:attribute name="observable-id" type="xs:ID" use="optional"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="RecordPattern">
      <xs:complexType>
        <xs:simpleContent>
          <xs:extension base="xs:string">
            <xs:attribute name="type"
                          type="recordpattern-type-type"
                          use="required"/>
            <xs:attribute name="ext-type"
                          type="xs:string" use="optional"/>
            <xs:attribute name="offset"
                          type="xs:integer" use="optional"/>
            <xs:attribute name="offsetunit"
                          type="recordpattern-offsetunit-type"
                          use="optional" default="line"/>
            <xs:attribute name="ext-offsetunit"
                          type="xs:string" use="optional"/>
            <xs:attribute name="instance"
                          type="xs:integer" use="optional"/>
          </xs:extension>
        </xs:simpleContent>
```

```
      </xs:complexType>
    </xs:element>
    <xs:simpleType name="recordpattern-type-type">
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="regex"/>
        <xs:enumeration value="binary"/>
        <xs:enumeration value="xpath"/>
        <xs:enumeration value="ext-value"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="recordpattern-offsetunit-type">
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="line"/>
        <xs:enumeration value="byte"/>
        <xs:enumeration value="ext-value"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:element name="RecordItem" type="iodef:ExtensionType"/>
    <!--
     ========================================================================
     ==  WindowsRegistryKeysModified Class                                 ==
     ========================================================================
    -->
    <xs:element name="WindowsRegistryKeysModified">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:Key" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="observable-id" type="xs:ID" use="optional"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="Key">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:KeyName"/>
          <xs:element ref="iodef:Value" minOccurs="0"/>
        </xs:sequence>
        <xs:attribute name="registryaction"
                      type="key-registryaction-type"/>
        <xs:attribute name="ext-registryaction"
                      type="xs:string" use="optional"/>
        <xs:attribute name="observable-id" type="xs:ID" use="optional"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="KeyName" type="xs:string"/>
    <xs:element name="Value" type="xs:string"/>
    <xs:simpleType name="key-registryaction-type">
      <xs:restriction base="xs:NMTOKEN">
```

```
        <xs:enumeration value="add-key"/>
        <xs:enumeration value="add-value"/>
        <xs:enumeration value="delete-key"/>
        <xs:enumeration value="delete-value"/>
        <xs:enumeration value="modify-key"/>
        <xs:enumeration value="modify-value"/>
        <xs:enumeration value="ext-value"/>
      </xs:restriction>
    </xs:simpleType>
    <!--
    ========================================================================
    ==  FileData Class                                                    ==
    ========================================================================
    -->
    <xs:element name="FileData">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:File"
                      minOccurs="1" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="restriction"
                      type="iodef:restriction-type" use="optional"/>
        <xs:attribute name="ext-restriction"
                      type="xs:string" use="optional"/>
        <xs:attribute name="observable-id" type="xs:ID" use="optional"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="File">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:FileName" minOccurs="0"/>
          <xs:element ref="iodef:FileSize" minOccurs="0"/>
          <xs:element ref="FileType" minOccurs="0"/>
          <xs:element ref="iodef:URL"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:HashData" minOccurs="0"/>
          <xs:element ref="iodef:SignatureData" minOccurs="0"/>
          <xs:element ref="iodef:AssociatedSoftware" minOccurs="0"/>
          <xs:element ref="iodef:FileProperties"
                      minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="observable-id" type="xs:ID" use="optional"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="FileName" type="xs:string"/>
    <xs:element name="FileSize" type="xs:integer"/>
    <xs:element name="FileType" type="xs:string"/>
    <xs:element name="AssociatedSoftware" type="iodef:SoftwareType"/>
```

```
    <xs:element name="FileProperties" type="iodef:ExtensionType"/>
    <!--
    ========================================================================
    ==  HashData Class                                                    ==
    ========================================================================
    -->
    <xs:element name="HashData">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:HashTargetID" minOccurs="0"/>
          <xs:element ref="iodef:Hash"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:FuzzyHash"
                      minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="scope"
                      type="hashdata-scope-type" use="required"/>
        <xs:attribute name="ext-scope" type="xs:string" use="optional"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="HashTargetID" type="xs:string"/>
    <xs:simpleType name="hashdata-scope-type">
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="file-contents"/>
        <xs:enumeration value="file-pe-section"/>
        <xs:enumeration value="file-pe-iat"/>
        <xs:enumeration value="file-pe-resource"/>
        <xs:enumeration value="file-pdf-object"/>
        <xs:enumeration value="email-hash"/>
        <xs:enumeration value="email-headers-hash"/>
        <xs:enumeration value="email-body-hash"/>
        <xs:enumeration value="ext-value"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:element name="Hash">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="ds:DigestMethod"/>
          <xs:element ref="ds:DigestValue"/>
          <xs:element ref="ds:CanonicalizationMethod"
                      minOccurs="0"/>
          <xs:element ref="iodef:Application" minOccurs="0"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="FuzzyHash">
      <xs:complexType>
        <xs:sequence>
```

```
          <xs:element ref="iodef:FuzzyHashValue"
                      maxOccurs="unbounded"/>
          <xs:element ref="iodef:Application" minOccurs="0"/>
          <xs:element ref="iodef:AdditionalData"
                      minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="FuzzyHashValue" type="iodef:ExtensionType"/>
    <!--
     ========================================================================
     ==  SignatureData Class                                               ==
     ========================================================================
    -->
    <xs:element name="SignatureData">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="ds:Signature" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <!--
     ========================================================================
     ==  CertificateData                                                   ==
     ========================================================================
    -->
    <xs:element name="CertificateData">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:Certificate" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="restriction"
                      type="iodef:restriction-type" use="optional"/>
        <xs:attribute name="ext-restriction"
                      type="xs:string" use="optional"/>
        <xs:attribute name="observable-id" type="xs:ID" use="optional"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="Certificate">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="ds:X509Data"/>
          <xs:element ref="iodef:Description"
                      minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="observable-id" type="xs:ID" use="optional"/>
      </xs:complexType>
    </xs:element>
```

```
    <!--
     ========================================================================
     == IndicatorData Class                                                ==
     ========================================================================
    -->
    <xs:element name="IndicatorData">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:Indicator"
                      minOccurs="1" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="Indicator">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:IndicatorID"/>
          <xs:element ref="iodef:AlternativeIndicatorID"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:Description"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:StartTime" minOccurs="0"/>
          <xs:element ref="iodef:EndTime" minOccurs="0"/>
          <xs:element ref="iodef:Confidence" minOccurs="0"/>
          <xs:element ref="iodef:Contact"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:choice>
            <xs:element ref="iodef:Observable"/>
            <xs:element ref="iodef:ObservableReference"/>
            <xs:element ref="iodef:IndicatorExpression"/>
            <xs:element ref="iodef:IndicatorReference"/>
          </xs:choice>
          <xs:element ref="iodef:NodeRole"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:AttackPhase"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:Reference"
                      minOccurs="0" maxOccurs="unbounded"/>
          <xs:element ref="iodef:AdditionalData"
                      minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="restriction"
                      type="iodef:restriction-type" use="optional"/>
        <xs:attribute name="ext-restriction"
                      type="xs:string" use="optional"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="IndicatorID">
```

```
      <xs:complexType>
        <xs:simpleContent>
          <xs:extension base="xs:ID">
            <xs:attribute name="name" type="xs:string" use="required"/>
            <xs:attribute name="version"
                          type="xs:string" use="required"/>
          </xs:extension>
        </xs:simpleContent>
      </xs:complexType>
    </xs:element>
    <xs:element name="AlternativeIndicatorID">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:IndicatorID" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="restriction"
                      type="iodef:restriction-type" use="optional"/>
        <xs:attribute name="ext-restriction"
                      type="xs:string" use="optional"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="Observable">
      <xs:complexType>
        <xs:choice>
          <xs:element ref="iodef:System" minOccurs="0"/>
          <xs:element ref="iodef:Address" minOccurs="0"/>
          <xs:element ref="iodef:DomainData" minOccurs="0"/>
          <xs:element ref="iodef:Service" minOccurs="0"/>
          <xs:element ref="iodef:EmailData" minOccurs="0"/>
          <xs:element ref="iodef:WindowsRegistryKeysModified"
                           minOccurs="0"/>
          <xs:element ref="iodef:FileData" minOccurs="0"/>
          <xs:element ref="iodef:CertificateData" minOccurs="0"/>
          <xs:element ref="iodef:RegistryHandle" minOccurs="0"/>
          <xs:element ref="iodef:RecordData" minOccurs="0"/>
          <xs:element ref="iodef:EventData" minOccurs="0"/>
          <xs:element ref="iodef:Incident" minOccurs="0"/>
          <xs:element ref="iodef:Expectation" minOccurs="0"/>
          <xs:element ref="iodef:Reference" minOccurs="0"/>
          <xs:element ref="iodef:Assessment" minOccurs="0"/>
          <xs:element ref="iodef:DetectionPattern" minOccurs="0"/>
          <xs:element ref="iodef:HistoryItem" minOccurs="0"/>
          <xs:element ref="iodef:BulkObservable" minOccurs="0"/>
          <xs:element ref="iodef:AdditionalData"
                      minOccurs="0" maxOccurs="unbounded"/>
        </xs:choice>
        <xs:attribute name="restriction"
                      type="iodef:restriction-type" use="optional"/>
```

```
        <xs:attribute name="ext-restriction"
                      type="xs:string" use="optional"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="BulkObservable">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:BulkObservableFormat" minOccurs="0"/>
          <xs:element name="BulkObservableList"/>
          <xs:element ref="iodef:AdditionalData"
                      minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="type"
                      type="bulkobservable-type-type" use="required"/>
        <xs:attribute name="ext-type" type="xs:string" use="optional"/>
      </xs:complexType>
    </xs:element>
    <xs:simpleType name="bulkobservable-type-type">
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="asn"/>
        <xs:enumeration value="atm"/>
        <xs:enumeration value="e-mail"/>
        <xs:enumeration value="ipv4-addr"/>
        <xs:enumeration value="ipv4-net"/>
        <xs:enumeration value="ipv4-net-mask"/>
        <xs:enumeration value="ipv6-addr"/>
        <xs:enumeration value="ipv6-net"/>
        <xs:enumeration value="ipv6-net-mask"/>
        <xs:enumeration value="mac"/>
        <xs:enumeration value="site-uri"/>
        <xs:enumeration value="domain-name"/>
        <xs:enumeration value="domain-to-ipv4"/>
        <xs:enumeration value="domain-to-ipv6"/>
        <xs:enumeration value="domain-to-ipv4-timestamp"/>
        <xs:enumeration value="domain-to-ipv6-timestamp"/>
        <xs:enumeration value="ipv4-port"/>
        <xs:enumeration value="ipv6-port"/>
        <xs:enumeration value="windows-reg-key"/>
        <xs:enumeration value="file-hash"/>
        <xs:enumeration value="email-x-mailer"/>
        <xs:enumeration value="email-subject"/>
        <xs:enumeration value="http-user-agent"/>
        <xs:enumeration value="http-request-uri"/>
        <xs:enumeration value="mutex"/>
        <xs:enumeration value="file-path"/>
        <xs:enumeration value="user-name"/>
      </xs:restriction>
    </xs:simpleType>
```

```
    <xs:element name="BulkObservableFormat">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="iodef:Hash" minOccurs="0"/>
          <xs:element ref="iodef:AdditionalData"
                      minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="BulkObservableList" type="xs:string"/>
    <xs:element name="IndicatorExpression">
      <xs:complexType>
        <xs:sequence maxOccurs="unbounded">
          <xs:choice>
            <xs:element ref="iodef:IndicatorExpression"/>
            <xs:element ref="iodef:Observable"/>
            <xs:element ref="iodef:ObservableReference"/>
            <xs:element ref="iodef:IndicatorReference"/>
          </xs:choice>
          <xs:element ref="iodef:Confidence" minOccurs="0"/>
          <xs:element ref="iodef:AdditionalData"
                      minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="operator"
                      type="indicatorexpression-operator-type"
                      use="optional" default="and"/>
        <xs:attribute name="ext-operator"
                      type="xs:string" use="optional"/>
      </xs:complexType>
    </xs:element>
    <xs:simpleType name="indicatorexpression-operator-type">
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="not"/>
        <xs:enumeration value="and"/>
        <xs:enumeration value="or"/>
        <xs:enumeration value="xor"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:element name="ObservableReference">
      <xs:complexType>
        <xs:attribute name="uid-ref" type="xs:IDREF" use="required"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="IndicatorReference">
      <xs:complexType>
        <xs:attribute name="uid-ref" type="xs:IDREF" use="optional"/>
        <xs:attribute name="euid-ref" type="xs:string" use="optional"/>
        <xs:attribute name="version" type="xs:string" use="optional"/>
```

```
    </xs:complexType>
  </xs:element>
  <xs:element name="AttackPhase">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="iodef:AttackPhaseID"
                    minOccurs="0" maxOccurs="unbounded"/>
        <xs:element ref="iodef:URL" maxOccurs="unbounded"/>
        <xs:element ref="iodef:Description"
                    minOccurs="0" maxOccurs="unbounded"/>
        <xs:element ref="iodef:AdditionalData"
                    minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="AttackPhaseID" type="xs:string"/>
  <!--
   ========================================================================
   == Miscellaneous Classes                                              ==
   ========================================================================
  -->
  <xs:element name="AdditionalData" type="iodef:ExtensionType"/>
  <xs:element name="Description" type="iodef:MLStringType"/>
  <xs:element name="URL" type="xs:anyURI"/>
  <!--
   ========================================================================
   == IODEF Data Types                                                   ==
   ========================================================================
  -->
  <xs:simpleType name="PositiveFloatType">
    <xs:restriction base="xs:float">
      <xs:minExclusive value="0"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="MLStringType">
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute name="translation-id"
                      type="xs:string" use="optional"/>
        <xs:attribute ref="xml:lang"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
  <xs:simpleType name="PortlistType">
    <xs:restriction base="xs:string">
      <xs:pattern value="\d+(\-\d+)?(,\d+(\-\d+)?)*"/>
    </xs:restriction>
  </xs:simpleType>
```

```
    <xs:simpleType name="TimezoneType">
      <xs:restriction base="xs:string">
        <xs:pattern
         value="Z|[\+\-](0[0-9]|1[0-4]):[0-5][0-9]"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:complexType name="ExtensionType" mixed="true">
      <xs:sequence>
        <xs:any namespace="##any" processContents="lax"
                minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>

      <xs:attribute name="name" type="xs:string" use="optional"/>
      <xs:attribute name="dtype"
                    type="iodef:dtype-type" use="required"/>
      <xs:attribute name="ext-dtype" type="xs:string" use="optional"/>
      <xs:attribute name="meaning" type="xs:string" use="optional"/>
      <xs:attribute name="formatid" type="xs:string" use="optional"/>
      <xs:attribute name="restriction"
                    type="iodef:restriction-type" use="optional"/>
      <xs:attribute name="ext-restriction"
                    type="xs:string" use="optional"/>
      <xs:attribute name="observable-id" type="xs:ID" use="optional"/>
    </xs:complexType>
    <xs:complexType name="SoftwareType">
      <xs:sequence>
        <xs:element ref="iodef:SoftwareReference" minOccurs="0"/>
        <xs:element ref="iodef:URL"
                    minOccurs="0" maxOccurs="unbounded"/>
        <xs:element ref="iodef:Description"
                    minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
    <xs:element name="SoftwareReference">
      <xs:complexType>
        <xs:sequence>
          <xs:any namespace="##any" processContents="lax"
                  minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="spec-name"
                      type="softwarereference-spec-name-type"
                      use="required"/>
        <xs:attribute name="ext-spec-name"
                      type="xs:string" use="optional"/>
        <xs:attribute name="dtype"
                      type="softwarereference-dtype-type"
                      use="optional"/>
        <xs:attribute name="ext-dtype" type="xs:string" use="optional"/>
```

```
      </xs:complexType>
    </xs:element>
    <xs:simpleType name="softwarereference-spec-name-type">
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="custom"/>
        <xs:enumeration value="cpe"/>
        <xs:enumeration value="swid"/>
        <xs:enumeration value="ext-value"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="softwarereference-dtype-type">
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="bytes"/>
        <xs:enumeration value="integer"/>
        <xs:enumeration value="real"/>
        <xs:enumeration value="string"/>
        <xs:enumeration value="xml"/>
        <xs:enumeration value="ext-value"/>
      </xs:restriction>
    </xs:simpleType>
    <!--
     ========================================================================
     == Global attribute type declarations                                 ==
     ========================================================================
    -->
    <xs:simpleType name="yes-no-unknown-type">
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="yes"/>
        <xs:enumeration value="no"/>
        <xs:enumeration value="unknown"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="restriction-type">
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="default"/>
        <xs:enumeration value="public"/>
        <xs:enumeration value="partner"/>
        <xs:enumeration value="need-to-know"/>
        <xs:enumeration value="private"/>
        <xs:enumeration value="white"/>
        <xs:enumeration value="green"/>
        <xs:enumeration value="amber"/>
        <xs:enumeration value="red"/>
        <xs:enumeration value="ext-value"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="severity-type">
      <xs:restriction base="xs:NMTOKEN">
```

```
        <xs:enumeration value="low"/>
        <xs:enumeration value="medium"/>
        <xs:enumeration value="high"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="duration-type">
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="second"/>
        <xs:enumeration value="minute"/>
        <xs:enumeration value="hour"/>
        <xs:enumeration value="day"/>
        <xs:enumeration value="month"/>
        <xs:enumeration value="quarter"/>
        <xs:enumeration value="year"/>
        <xs:enumeration value="ext-value"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="action-type">
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="nothing"/>
        <xs:enumeration value="contact-source-site"/>
        <xs:enumeration value="contact-target-site"/>
        <xs:enumeration value="contact-sender"/>
        <xs:enumeration value="investigate"/>
        <xs:enumeration value="block-host"/>
        <xs:enumeration value="block-network"/>
        <xs:enumeration value="block-port"/>
        <xs:enumeration value="rate-limit-host"/>
        <xs:enumeration value="rate-limit-network"/>
        <xs:enumeration value="rate-limit-port"/>
        <xs:enumeration value="redirect-traffic"/>
        <xs:enumeration value="honeypot"/>
        <xs:enumeration value="upgrade-software"/>
        <xs:enumeration value="rebuild-asset"/>
        <xs:enumeration value="harden-asset"/>
        <xs:enumeration value="remediate-other"/>
        <xs:enumeration value="status-triage"/>
        <xs:enumeration value="status-new-info"/>
        <xs:enumeration value="watch-and-report"/>
        <xs:enumeration value="defined-coa"/>
        <xs:enumeration value="other"/>
        <xs:enumeration value="ext-value"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="dtype-type">
      <xs:restriction base="xs:NMTOKEN">
        <xs:enumeration value="boolean"/>
        <xs:enumeration value="byte"/>
```

```
        <xs:enumeration value="bytes"/>
        <xs:enumeration value="character"/>
        <xs:enumeration value="date-time"/>
        <xs:enumeration value="integer"/>
        <xs:enumeration value="ntpstamp"/>
        <xs:enumeration value="portlist"/>
        <xs:enumeration value="real"/>
        <xs:enumeration value="string"/>
        <xs:enumeration value="file"/>
        <xs:enumeration value="path"/>
        <xs:enumeration value="frame"/>
        <xs:enumeration value="packet"/>
        <xs:enumeration value="ipv4-packet"/>
        <xs:enumeration value="ipv6-packet"/>
        <xs:enumeration value="url"/>
        <xs:enumeration value="csv"/>
        <xs:enumeration value="winreg"/>
        <xs:enumeration value="xml"/>
        <xs:enumeration value="ext-value"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:schema>
```

9. Security Considerations

The IODEF data model does not directly introduce security or privacy issues. However, as the data encoded by the IODEF might be considered sensitive by the parties exchanging it or by those described by it, care needs to be taken to ensure appropriate handling during the document construction, exchange, processing, archiving, subsequent retrieval and analysis.

9.1. Security

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 [RFC6545] and its associated transport binding IODEF/RID over HTTP/TLS [RFC6546] provide such security.

An IODEF implementation may act on the data in the document. These actions might be explicitly requested in the document or the result of analytical logic that triggered on data in the document. For this reason, care must be taken by IODEF implementations to properly authenticate the sender and receiver of the document. The sender needs confidence that sensitive information and timely requests for action are sent to the correct recipient. The recipient may

interpret the contents of the document differently based on who sent it; or vary actions based on the sender. While the sender of the document may explicitly convey confidence in the data in a granular way using the Confidence class, the recipient is free to ignore or refine this information to make its own assessment. Ambiguous Confidence elements (where it is unclear to which of a set of other elements the Confidence element relates) in a document MUST be ignored by the recipient.

Certain classes may require out-of-band coordination to agree upon their semantics (e.g., Confidence@rating="low" or DefinedCOA). This coordination MUST occur prior to operational data exchange to prevent the incorrect interpretation of these select data elements. When parsing these data elements, implementations should validate, when possible, that they conform to the agreed upon semantics. These semantics may need to be periodically reevaluated.

Executable content of various forms could be embedded into the IODEF document directly or through an extension. Implementation MUST handle this content with care to prevent unintentional automated execution. The following classes are explicitly intended to represent content that might be executable:

- All classes of type iodef:ExtensionType and the RecordPattern class can represent arbitrary binary strings such as legitimate software programs or malware.

- The EmailMessage and EmailBody classes can represent email attachments that can contain arbitrary content.

- The DetectionPattern class could specify a machine-readable configuration that directs the execution of the corresponding tool.

Per Section 4.3, IODEF implementations will need to periodically consult the IANA registries specified in Section 10.2 to discover newly registered enumerated attribute values. These implementations MUST communicate with IANA in a way that ensures the integrity of the values and the authenticity of the source. HTTPS over TLS [RFC2818][RFC5246] provides such security.

9.2. Privacy

The IODEF contains numerous fields that are identifiers which could be linked to an individual or organization. IODEF documents may contain sensitive information about these identified parties; and repeated document exchanges about the same and related parties may

enable the correlation of data about them. Likewise, a party may report on another to a third party without their knowledge.

When creating an IODEF document, careful consideration must be given to what information is shared. Personal identifiers and attributable sensitive information should only be shared when necessary.

When exchanging documents, transport security MUST provide document-level confidentiality. XML element-level confidentiality can also be provided by using [W3C.XMLENC].

In order to suggest data processing and handling guidelines of the encoded information, the IODEF allows a document sender to convey a privacy policy using the restriction attribute. The various instances of this attribute allow different data elements of the document to be covered by dissimilar policies. While flexible, it must be stressed that this approach only serves as a guideline from the sender, as the recipient is free to ignore it.

Although outside of the scope of an IODEF implementation, the contents of IODEF documents and any derived analysis should be archived with at appropriate confidentiality controls. Likewise, access to retrieve and analyze this data should be restricted to authorized users.

10. IANA Considerations

This document registers a namespace, an XML schema, and a number of registries that map to enumerated values defined in the data model. It also defines an expert review process for IODEF-related XML registry entries.

10.1. Namespace and Schema

This document uses URNs to describe an XML namespace and schema conforming to a registry mechanism described in [RFC3688]

Registration for the IODEF namespace:

- URI: urn:ietf:params:xml:ns:iodef-2.0
- Registrant Contact: See the first author of the "Author’s Address" section of this document.
- XML: None. Namespace URIs do not represent an XML specification.

Registration for the IODEF XML schema:

- URI: urn:ietf:params:xml:schema:iodef-2.0
- Registrant Contact: See the first author of the "Author's Address" section of this document.
- XML: See Section 8 of this document.

10.2. Enumerated Value Registries

This document creates 34 identically structured registries to be managed by IANA:

- Name of the parent registry: "Incident Object Description Exchange Format v2 (IODEF)"
- URL of the registry: http://www.iana.org/assignments/iodef2
- Namespace format: A registry entry consists of:
  - Value. A value for a given IODEF attribute. It MUST conform to the formatting specified by the IODEF ENUM data type which is implemented as an "xs:NMTOKEN" type per Section 3.3.4 of [W3C.SCHEMA.DTYPES]. The value SHOULD conform to the convention specified in Section 5.2.
  - Description. A short description of the enumerated value.
  - Reference. An optional list of URIs to further describe the value.
- Allocation policy: Expert Review per [RFC5226]. This reviewer will ensure that the requested registry entry conforms to the prescribed formatting. The reviewer will also ensure that the entry is an appropriate value for the attribute per the information model (Section 3).

The registries to be created are named in the "Registry Name" column of Table 1. Each registry is initially populated with values and descriptions that come from an attribute specified in the IODEF schema (Section 8) whose description is found in a sub-section of the information model (Section 3). The initial values for the Value and Description fields of a given registry are listed in the "IV (Value)" and "IV (Description)" columns respectively. The "IV (Value)" points to a given schema type per Section 8. Each enumerated value in the schema gets a corresponding entry in a given registry. The "IV (Description)" points to a section in the text of this document that describes each enumerated value. The initial value of the Reference

field of every registry entry described below should be this document.

Registry Name	IV (Value)	IV (Description)
Restriction	iodef-restriction-type	Section 3.3.1
Incident-purpose	incident-purpose-type	Section 3.2
Incident-status	incident-status-type	Section 3.2
Contact-role	contact-role-type	Section 3.9
Contact-type	contact-type-type	Section 3.9
RegistryHandle-registry	registryhandle-registry-type	Section 3.9.1
PostalAddress-type	postaladdress-type-type	Section 3.9.2
Telephone-type	telephone-type-type	Section 3.9.4
Email-type	email-type-type	Section 3.9.3
Expectation-action	action-type	Section 3.15
Discovery-source	discovery-source-type	Section 3.10
SystemImpact-type	systemimpact-type-type	Section 3.12.1
BusinessImpact-severity	businessimpact-severity-type	Section 3.12.2
BusinessImpact-type	businessimpact-type-type	Section 3.12.2
TimeImpact-metric	timeimpact-metric-type	Section 3.12.3
TimeImpact-duration	duration-type	Section 3.12.3
Confidence-rating	confidence-rating-type	Section 3.12.5

NodeRole-category	noderole-category-type	Section 3.18.2
System-category	system-category-type	Section 3.17
System-ownership	system-ownership-type	Section 3.17
Address-category	address-category-type	Section 3.18.1
Counter-type	counter-type-type	Section 3.18.3
Counter-unit	counter-unit-type	Section 3.18.3
DomainData-system-status	domaindata-system-status-type	Section 3.19
DomainData-domain-status	domaindata-domain-status-type	Section 3.19
RecordPattern-type	recordpattern-type-type	Section 3.22.2
RecordPattern-offsetunit	recordpattern-offsetunit-type	Section 3.22.2
Key-registryaction	key-registryaction-type	Section 3.23.1
HashData-scope	hashdata-scope-type	Section 3.26
BulkObservable-type	bulkobservable-type-type	Section 3.29.3.1
IndicatorExpression-operator	indicatorexpression-operator-type	Section 3.29.4
ExtensionType-dtype	dtype-type	Section 2.16
SoftwareReference-spec-id	softwarereference-spec-id-type	Section 2.15.1
SoftwareReference-dtype	softwarereference-dtype-type	Section 2.15.1

Table 1: IANA Enumerated Value Registries

10.3. Expert Review of IODEF-Related XML Registry Entries

IODEF class extensions, per Section 5.2, could register their namespaces and schemas with the IANA XML Namespace ("ns", http://www.iana.org/assignments/xml-registry/xml-registry.xhtml#ns) and Schema registries ("schema", http://www.iana.org/assignments/xml-registry/xml-registry.xhtml#schema) described in [RFC3688]. In addition to any reviews required by IANA, 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] to ensure compatibility with IODEF and other existing IODEF extensions.

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

This document obsoletes [RFC6685].

11. Acknowledgments

Thanks to Paul Stockler for his editorial leadership in the transition of RFC5070bis to this document.

Thanks to Kathleen Moriarty, Brian Trammel, Alexey Melnikov, Takeshi Takahashi, David Waltermire and Sean Turner as the MILE working group chairs, secretary or area directors for providing feedback and coordination of this document.

Thanks to the following individuals (listed alphabetically) who provided feedback during the meetings, on the mailing list or through implementation experience: Jerome Athias, David Black, Eric Burger, Toma Cejka, Patrick Curry, John Field, Christopher Harrington, Chris Inacio, Panos Kampanakis, David Misell, Daisuke Miyamoto, Adam Montville, Robert Moskowitz, Lagadec Philippe, Tony Rutkowski, Mio Suzuki and Nik Teague.

12. References

12.1. Normative References

[W3C.XML] World Wide Web Consortium, "Extensible Markup Language (XML) 1.0 (Fifth Edition)", W3C Recommendation , November 2008, <http://www.w3.org/TR/2008/REC-xml-20081126/>.

[W3C.SCHEMA]
World Wide Web Consortium, "XML XML Schema Part 1: Structures Second Edition", W3C Recommendation , October 2004, <http://www.w3.org/TR/xmlschema-1/>.

[W3C.SCHEMA.DTYPES]
World Wide Web Consortium, "XML Schema Part 2: Datatypes Second Edition", W3C Recommendation , October 2004, <http://www.w3.org/TR/xmlschema-2/>.

[W3C.XMLNS]
World Wide Web Consortium, "Namespaces in XML 1.0 (Third Edition)", W3C Recommendation , December 2009, <http://www.w3.org/TR/2009/REC-xml-names-20091208/>.

[W3C.XPATH]
World Wide Web Consortium, "XML Path Language (XPath) 3.1", W3C Candidate Recommendation , December 2015, <https://www.w3.org/TR/xpath-3/>.

[W3C.XMLSIG]
World Wide Web Consortium, "XML Signature Syntax and Processing 2.0", W3C Recommendation , June 2008, <http://www.w3.org/TR/xmldsig-core/>.

[IEEE.POSIX]
Institute of Electrical and Electronics Engineers, "Information Technology - Portable Operating System Interface (POSIX) - Part 1: Base Definitions", IEEE 1003.1, June 2001.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", RFC 2119, March 1997.

[RFC5646] Philips, A. and M. Davis, "Tags for Identifying of Languages", RFC 5646, September 2009.

[RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifiers (URI): Generic Syntax", RFC 3986, January 2005‘.

[RFC4519] Sciberras, A., "Schema for User Applications", RFC 4519, June 2006.

[RFC5322] Resnick, P., "Internet Message Format", RFC 5322, October 2008.

[RFC6531] Yao, J. and W. Mao, "SMTP Extension for Internationalized Email", RFC 6531, February 2012.

[RFC7495] Montville, A. and D. Black, "IODEF Enumeration Reference Format", RFC 7495, January 2015.

[RFC7203] Takahashi, T., Landfield, K., and Y. Kadobayashi, "An Incident Object Description Exchange Format (IODEF) Extension for Structured Cybersecurity Information", RFC 7203, April 2014.

[ISO4217] International Organization for Standardization, "International Standard: Codes for the representation of currencies and funds, ISO 4217:2001", ISO 4217:2001, August 2001.

[RFC3688] Mealling, M., "The IETF XML Registry", RFC 3688, January 2004.

[IANA.Ports]
Internet Assigned Numbers Authority, "Service Name and Transport Protocol Port Number Registry", January 2014, <http://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.txt>.

[IANA.Protocols]
Internet Assigned Numbers Authority, "Assigned Internet Protocol Numbers", January 2014, <http://www.iana.org/assignments/protocol-numbers/protocol-numbers.txt>.

[RFC3629] Yergeau, F., "UTF-8, a transformation format of ISO 10646", RFC 3629, November 2003.

[RFC2781] Hoffman, P. and F. Yergeau, "UTF-16, an encoding of ISO 10646", RFC 2781, February 2000.

[IANA.Media]
Internet Assigned Numbers Authority, "Media Types", March 2015, <http://www.iana.org/assignments/media-types/media-types.xhtml>.

[NIST.CPE]
The National Institute of Standards and Technology, "Common Platform Enumeration", 2014, <http://scap.nist.gov/specifications/cpe/>.

[ISO19770]
International Organization for Standardization, "Information technology -- Software asset management -- Part 2: Software identification tag, ISO/IEC 19770-2:2015", ISO 19770-2:2015, October 2015.

[E.164] ITU Telecommunication Standardization Sector, "The International Public Telecommunication Numbering Plan", ITU-T Recommendation E.164 (02/05), February 2005.

[RFC5952] Kawamura, S. and M. Kawashima, "A Recommendation for IPv6 Address Text Representation", RFC 5952, August 2010.

[RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 4291, February 2006.

12.2. Informative References

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

[RFC6685] Trammell, B., "Expert Review for Incident Object Description Exchange Format (IODEF) Extensions in IANA XML Registry", RFC 6685, July 2012.

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

[RFC6546] Trammell, B., "Transport of Real-time Inter-network Defense (RID) Messages over HTTP/TLS", RFC 6546, April 2012.

[RFC5901] Cain, P. and D. Jevans, "Extensions to the IODEF-Document Class for Reporting Phishing", RFC 5901, July 2010.

[NIST800.61rev2]
Cichonski, P., Millar, T., Grance, T., and K. Scarfone, "NIST Special Publication 800-61 Revision 2: Computer Security Incident Handling Guide", January 2012, <http://csrc.nist.gov/publications/nistpubs/800-61rev2/SP800-61rev2.pdf>.

[RFC3982] Newton, A. and M. Sanz, "IRIS: A Domain Registry (dreg) Type for the Internet Registry Information Service (IRIS)", RFC 3982, January 2005.

[KB310516]
Microsoft Corporation, "How to add, modify, or delete registry subkeys and values by using a registration entries (.reg) file", December 2007.

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

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", RFC 5226, May 2008.

[W3C.XMLENC]
World Wide Web Consortium, "XML Encryption Syntax and Processing Version 1.1", W3C Recommendation , April 2013, <https://www.w3.org/TR/xmlenc-core1/>.

[RFC2818] Rescorla, E., "HTTP Over TLS", RFC 2818, May 2000.

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, August 2008.

Author's Address

Roman Danyliw
CERT - Carnegie Mellon University
4500 Fifth Avenue
Pittsburgh, PA
USA

EMail: rdd@cert.org