Internet Engineering Task Force (IETF)
Request for Comments: 7203
Category: Standards Track
ISSN: 2070-1721

T. Takahashi
NICT
K. Landfield
McAfee
Y. Kadobayashi
NAIST

April 2014

An Incident Object Description Exchange Format (IODEF) Extension for Structured Cybersecurity Information

Abstract

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

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

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

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

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

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

2. Terminology

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

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

3. Applicability

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

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

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

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

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

4. Extension Definition

This document extends IODEF to embed structured information by introducing new classes that can be embedded consistently inside an IODEF document as element contents of the AdditionalData and RecordItem classes [RFC5070].

4.1. IANA Table for Structured Cybersecurity Information

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

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

The initial IANA table has only one entry, as follows:

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

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

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

The SpecID attributes of extension classes (Section 4.5) must allow the values of the specifications' namespace fields, but implementations are otherwise not required to support all specifications of the IANA table and may choose which specifications to support. However, at a minimum, the specification listed in the initial IANA table needs to be supported, as described in Section 5. If an implementation received data that it does not support, it may expand its functionality by looking up the IANA table or notify the sender of its inability to parse the data. Note that the lookup could be done manually or automatically, but automatic download of data from IANA’s website is not recommended, since it is not designed for mass retrieval of data by multiple devices.

4.2. Extended Data Type: XMLDATA

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

4.3. Extending IODEF

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

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

Figure 1: Incident Class

4.4. Basic Structure of the Extension Classes

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

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

Figure 2: Basic Structure

Three attributes are defined as indicated below:

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

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

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

Likewise, two elements are defined as indicated below:

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

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

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

4.5. Defining Extension Classes

This document defines eight extension classes, as described in the subsections that follow.

4.5.1. AttackPattern

An AttackPattern is an extension class to the Incident.Method.AdditionalData element with a dtype of "xml". It describes attack patterns of incidents or events. It is RECOMMENDED that the Method class [RFC5070] contain the extension elements whenever available. An AttackPattern class is structured as follows:

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

Figure 3: AttackPattern Class

This class has the following attributes:

SpecID: REQUIRED. ENUM. See Section 4.4.

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

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

Likewise, this class has the following elements:

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

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

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

4.5.2. Platform

A Platform is an extension class that identifies a software platform. It is RECOMMENDED that the AttackPattern, Vulnerability, Weakness, and System [RFC5070] classes contain the extension elements whenever available. A Platform element is structured as follows:

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

Figure 4: Platform Class

This class has the following attributes:

SpecID: REQUIRED. ENUM. See Section 4.4.

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

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

Likewise, this class has the following elements:

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

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

4.5.3. Vulnerability

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

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

Figure 5: Vulnerability Class

This class has the following attributes:

SpecID: REQUIRED. ENUM. See Section 4.4.

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

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

Likewise, this class has the following elements:

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

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

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

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

4.5.4. Scoring

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

A Scoring class is structured as follows:

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

Figure 6: Scoring Class

This class has the following attributes:

SpecID: REQUIRED. ENUM. See Section 4.4.

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

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

Likewise, this class has the following elements:

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

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

4.5.5. Weakness

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

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

Figure 7: Weakness Class

This class has the following attributes:

SpecID: REQUIRED. ENUM. See Section 4.4.

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

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

Likewise, this class has the following elements:

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

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

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

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

4.5.6. EventReport

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

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

Figure 8: EventReport Class

This class has the following attributes:

SpecID: REQUIRED. ENUM. See Section 4.4.

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

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

Likewise, this class has the following elements:

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

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

4.5.7. Verification

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

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

Figure 9: Verification Class

This class has the following attributes:

SpecID: REQUIRED. ENUM. See Section 4.4.

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

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

Likewise, this class has the following elements:

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

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

4.5.8. Remediation

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

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

Figure 10: Remediation Class

This class has the following attributes:

SpecID: REQUIRED. ENUM. See Section 4.4.

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

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

Likewise, this class has the following elements:

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

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

5. Mandatory-to-Implement Features

Implementations compliant with this document MUST be capable of sending and receiving the extended IODEF documents that contain XML documents conforming to the specification listed in the initial IANA table described in Section 4.1 without error. The extended IODEF document is an XML document that MUST be well-formed and MUST be valid according to schemata, including extension schemata, available to the validator and applicable to the XML document. Note that the receiver can look up the namespace in the IANA table to understand what specifications the embedded XML documents follow.

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

5.1. An Example XML Document

An example IODEF document for checking an implementation's conformity with mandatory-to-implement features is provided here. The document carries Malware Metadata Exchange Format (MMDEF) metadata. Note that the metadata is generated by genMMDEF [MMDEF] with EICAR [EICAR] files. Due to the limit of 72 characters per line, some line breaks were added in this example.


```
<?xml version="1.0" encoding="UTF-8"?>
<IODEF-Document version="1.00" lang="en"
 xmlns="urn:ietf:params:xml:ns:iodef-1.0"
 xmlns:iodef="urn:ietf:params:xml:ns:iodef-1.0"
 xmlns:sci="urn:ietf:params:xml:ns:iodef-sci-1.0"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <Incident purpose="reporting">
    <IncidentID name="sci.example.com">189493</IncidentID>
    <ReportTime>2013-06-18T23:19:24+00:00</ReportTime>
    <Description>a candidate security incident</Description>
    <Assessment>
      <Impact completion="failed" type="admin" />
    </Assessment>
    <Method>
      <Description>A candidate attack event</Description>
      <AdditionalData dtype="xml">
        <sci:AttackPattern SpecID=
               "urn:ietf:params:xml:ns:mile:mmdef:1.2">
          <sci:RawData dtype="xml">
            <malwareMetaData xmlns="http://xml/metadataSharing.xsd"
             xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
             xsi:schemaLocation="http://xml/metadataSharing.xsd
             file:metadataSharing.xsd" version="1.200000" id="10000">
              <company>N/A</company>
              <author>MMDEF Generation Script</author>
              <comment>Test MMDEF v1.2 file generated using genMMDEF
              </comment>
              <timestamp>2013-03-23T15:12:50.726000</timestamp>
              <objects>
                <file id="6ce6f415d8475545be5ba114f208b0ff">
                  <md5>6ce6f415d8475545be5ba114f208b0ff</md5>
                  <sha1>da39a3ee5e6b4b0d3255bfef95601890afd80709</sha1>
                  <sha256>e3b0c44298fc1c149afbf4c8996fb92427ae41e464
                          9b934ca495991b7852b855</sha256>
```

```
                   <sha512>cf83e1357eefb8bdf1542850d66d8007d620e4050b
                           5715dc83f4a921d36ce9ce47d0d13c5d85f2b0ff83
                           18d2877eec2f63b931bd47417a81a538327af927
                           da3e</sha512>
                   <size>184</size>
                   <filename>eicar_com.zip</filename>
                   <MIMEType>application/zip</MIMEType>
                 </file>
                 <file id="44d88612fea8a8f36de82e1278abb02f">
                   <md5>44d88612fea8a8f36de82e1278abb02f</md5>
                   <sha1>3395856ce81f2b7382dee72602f798b642f14140</sha1>
                   <sha256>275a021bbfb6489e54d471899f7db9d1663fc695ec
                           2fe2a2c4538aabf651fd0f</sha256>
                   <sha512>cc805d5fab1fd71a4ab352a9c533e65fb2d5b88551
                           8f4e565e68847223b8e6b85cb48f3afad842726d99
                           239c9e36505c64b0dc9a061d9e507d833277ada3
                           36ab</sha512>
                   <size>68</size>
                   <crc32>1750191932</crc32>
                   <filename>eicar.com</filename>
                   <filenameWithinInstaller>eicar.com
                   </filenameWithinInstaller>
                 </file>
               </objects>
             <relationships>
               <relationship type="createdBy" id="1">
                 <source>
                   <ref>file[@id="6ce6f415d8475545be5ba114f208b0ff"]
                   </ref>
                 </source>
                 <target>
                   <ref>file[@id="44d88612fea8a8f36de82e1278abb02f"]
                   </ref>
                 </target>
                 <timestamp>2013-03-23T15:12:50.744000</timestamp>
                 </relationship>
               </relationships>
             </malwareMetaData>
           </sci:RawData>
         </sci:AttackPattern>
       </AdditionalData>
     </Method>
     <Contact role="creator" type="organization">
       <ContactName>sci.example.com</ContactName>
       <RegistryHandle registry="arin">sci.example-com
       </RegistryHandle>
       <Email>contact@csirt.example.com</Email>
     </Contact>
```

```
     <EventData>
       <Flow>
         <System category="source">
           <Node>
             <Address category="ipv4-addr">192.0.2.200</Address>
             <Counter type="event">57</Counter>
           </Node>
         </System>
         <System category="target">
           <Node>
             <Address category="ipv4-net">192.0.2.16/28</Address>
           </Node>
           <Service ip_protocol="4">
             <Port>80</Port>
           </Service>
         </System>
       </Flow>
       <Expectation action="block-host" />
       <Expectation action="other" />
     </EventData>
   </Incident>
 </IODEF-Document>
```

5.2. An XML Schema for the Extension

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

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

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

<xsd:import namespace="urn:ietf:params:xml:ns:iodef-1.0" schemaLocation=
 "http://www.iana.org/assignments/xml-registry/schema/iodef-1.0.xsd"/>

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

```
      <xsd:attribute name="ext-dtype" type="xsd:string"
       use="prohibited"/>
      <xsd:attribute name="meaning" type="xsd:string"/>
      <xsd:attribute name="formatid" type="xsd:string"/>
      <xsd:attribute name="restriction" type="iodef:restriction-type"/>
    </xsd:restriction>
  </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="BasicStructure">
  <xsd:sequence>
    <xsd:choice>
      <xsd:element name="RawData" type="sci:XMLDATA"
       minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element ref="iodef:Reference" minOccurs="0"
       maxOccurs="unbounded"/>
    </xsd:choice>
  </xsd:sequence>
  <xsd:attribute name="SpecID" type="xsd:string" use="required"/>
  <xsd:attribute name="ext-SpecID" type="xsd:string"/>
  <xsd:attribute name="ContentID" type="xsd:string"/>
</xsd:complexType>

<xsd:element name="Scoring" type="sci:BasicStructure"/>
<xsd:element name="Platform" type="sci:BasicStructure"/>
<xsd:element name="EventReport" type="sci:BasicStructure"/>
<xsd:element name="Verification" type="sci:BasicStructure"/>
<xsd:element name="Remediation" type="sci:BasicStructure"/>
<xsd:element name="AttackPattern">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="sci:BasicStructure">
        <sequence>
          <xsd:element ref="sci:Platform" minOccurs="0"
           maxOccurs="unbounded"/>
        </sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
<xsd:element name="Vulnerability">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="sci:BasicStructure">
        <sequence>
          <xsd:element ref="sci:Platform" minOccurs="0"
           maxOccurs="unbounded"/>
          <xsd:element ref="sci:Scoring" minOccurs="0"
           maxOccurs="unbounded"/>
```

```
        </sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
<xsd:element name="Weakness">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="sci:BasicStructure">
        <sequence>
          <xsd:element ref="sci:Platform" minOccurs="0"
           maxOccurs="unbounded"/>
          <xsd:element ref="sci:Scoring" minOccurs="0"
           maxOccurs="unbounded"/>
        </sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

</xsd:schema>
```

6. Security Considerations

This document specifies a format for encoding a particular class of security incidents appropriate for exchange across organizations. As merely a data representation, it does not directly introduce security issues. However, it is guaranteed that parties exchanging instances of this specification will have certain concerns. For this reason, the underlying message format and transport protocol used MUST ensure the appropriate degree of confidentiality, integrity, and authenticity for the specific environment. Specific security considerations are detailed in the messaging and transport documents, where the exchange of formatted information is automated; see Sections 9 and 10 of "Real-time Inter-network Defense (RID)" [RFC6545] and Section 4 of "Transport of Real-time Inter-network Defense (RID) Messages over HTTP/TLS" [RFC6546] for a detailed overview of security requirements and considerations.

It is RECOMMENDED that organizations that exchange data using this document develop operating procedures that consider, at a minimum, the following areas of concern.

6.1. Transport-Specific Concerns

The underlying messaging format, IODEF, provides data markers to indicate the sensitivity level of specific classes within the structure as well as for the entire XML document. The "restriction" attribute accomplishes this with four attribute values in IODEF [RFC5070]. These values are RECOMMENDED for use at the application level, prior to transport, to protect data as appropriate. A standard mechanism to apply XML encryption using these attribute values as triggers is defined in RID [RFC6545], Section 9.1. This mechanism may be used whether or not the RID protocol [RFC6545] and its associated transport binding [RFC6546] are used in the exchange to provide object-level security on the data to prevent possible intermediary systems or middleboxes from having access to the data being exchanged. In areas where transmission security or secrecy is questionable, the application of an XML digital signature [XMLDSIG] and/or encryption on each report will counteract both of these concerns. The data markers are RECOMMENDED for use by applications for managing access controls; however, access controls and management of those controls are out of scope for this document. Options such as the usage of a standard language (e.g., eXtensible Access Control Markup Language [XACML]) for the expression of authorization policies can be used to enable source and destination systems to better coordinate and align their respective policy expressions.

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

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

6.2. Protection of Sensitive and Private Information

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

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

6.3. Application and Server Security

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

7. IANA Considerations

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

The following IODEF structured cybersecurity information extension namespace has been registered:

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

Registrant Contact: Refer to the Authors’ Addresses section of this document.

XML: None.

The following IODEF structured cybersecurity information extension XML schema has been registered:

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

Registrant Contact: Refer to the Authors’ Addresses section of this document.

XML: Refer to the XML schema in Section 5.2 of this document.

This memo creates the following registry, which is managed by IANA:

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

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

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

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

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

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

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

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

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

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

Initial registry contents: Only one entry, with the following values:

Namespace: urn:ietf:params:xml:ns:mile:mmdef:1.2

Specification Name: Malware Metadata Exchange Format

Version: 1.2

Reference URI:

<http://standards.ieee.org/develop/indconn/icsg/mmdef.html>, <http://grouper.ieee.org/groups/malware/malwg/Schema1.2/>

Applicable Classes: AttackPattern

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

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

8. Acknowledgments

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

9. References

9.1. Normative References

[MMDEF] ICSG Malware Metadata Exchange Format Working Group, "Malware Metadata Exchange Format", IEEE Standards Association, November 2011, <http://grouper.ieee.org/groups/malware/malwg/Schema1.2/>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, January 2005.

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

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.

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

[RFC6546] Trammell, B., "Transport of Real-time Inter-network Defense (RID) Messages over HTTP/TLS", RFC 6546, April 2012.

[XML1.0] Bray, T., Paoli, J., Sperberg-McQueen, C., Maler, E., and F. Yergeau, "Extensible Markup Language (XML) 1.0 (Fifth Edition)", W3C Recommendation, November 2008, <http://www.w3.org/TR/xml/>.

[XMLschemaPart1] Thompson, H., Beech, D., Maloney, M., and N. Mendelsohn, "XML Schema Part 1: Structures Second Edition", W3C Recommendation, October 2004, <http://www.w3.org/TR/xmlschema-1/>.

[XMLschemaPart2] Biron, P. and A. Malhotra, "XML Schema Part 2: Datatypes Second Edition", W3C Recommendation, October 2004, <http://www.w3.org/TR/xmlschema-2/>.

[XMLNames] Bray, T., Hollander, D., Layman, A., Tobin, R., and H. Thompson, "Namespaces in XML 1.0 (Third Edition)", W3C Recommendation, December 2009, <http://www.w3.org/TR/xml-names/>.

9.2. Informative References

[RFC3553] Mealling, M., Masinter, L., Hardie, T., and G. Klyne, "An IETF URN Sub-namespace for Registered Protocol Parameters", BCP 73, RFC 3553, June 2003.

[RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, January 2004.

[CAPEC] The MITRE Corporation, "Common Attack Pattern Enumeration and Classification (CAPEC)", <http://capec.mitre.org/>.

[CCE] National Institute of Standards and Technology, "Common Configuration Enumeration (CCE)", <http://nvd.nist.gov/cce/index.cfm>.

[CCSS] Scarfone, K. and P. Mell, "The Common Configuration Scoring System (CCSS): Metrics for Software Security Configuration Vulnerabilities", NIST Interagency Report 7502, December 2010, <http://csrc.nist.gov/publications/nistir/ir7502/nistir-7502_CCSS.pdf>.

[CEE] The MITRE Corporation, "Common Event Expression (CEE)", <http://cee.mitre.org/>.

[CPE] National Institute of Standards and Technology, "Common Platform Enumeration", June 2011, <http://scap.nist.gov/specifications/cpe/>.

[CVE] The MITRE Corporation, "Common Vulnerabilities and Exposures (CVE)", <http://cve.mitre.org/>.

[CVRF] ICASI, "The Common Vulnerability Reporting Framework (CVRF)", <http://www.icasi.org/cvrf>.

[CVSS] Mell, P., Scarfone, K., and S. Romanosky, "The Common Vulnerability Scoring System (CVSS) and Its Applicability to Federal Agency Systems", NIST Interagency Report 7435, August 2007, <http://csrc.nist.gov/publications/nistir/ir7435/NISTIR-7435.pdf>.

[CWE] The MITRE Corporation, "Common Weakness Enumeration (CWE)", <http://cwe.mitre.org/>.

[CWSS] The MITRE Corporation, "Common Weakness Scoring System (CWSS(TM))", <http://cwe.mitre.org/cwss/>.

[EICAR] EICAR - European Expert Group for IT-Security, "Anti-Malware Testfile", 2003, <http://www.eicar.org/86-0-Intended-use.html>.

[MAEC] The MITRE Corporation, "Malware Attribute Enumeration and Characterization", <http://maec.mitre.org/>.

[OCIL] Waltermire, D., Scarfone, K., and M. Casipe, "Specification for the Open Checklist Interactive Language (OCIL) Version 2.0", NIST Interagency Report 7692, April 2011, <http://csrc.nist.gov/publications/nistir/ir7692/nistir-7692.pdf>.

[OVAL] The MITRE Corporation, "Open Vulnerability and Assessment Language (OVAL)", <http://oval.mitre.org/>.

[SCAP] Waltermire, D., Quinn, S., Scarfone, K., and A. Halbardier, "The Technical Specification for the Security Content Automation Protocol (SCAP): SCAP Version 1.2", NIST Special Publication 800-126 Revision 2, September 2011, <http://csrc.nist.gov/publications/nistpubs/800-126-rev2/SP800-126r2.pdf>.

[XACML] Rissanen, E., "eXtensible Access Control Markup Language (XACML) Version 3.0", January 2013, <http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.pdf>.

[XCCDF] Waltermire, D., Schmidt, C., Scarfone, K., and N. Ziring, "Specification for the Extensible Configuration Checklist Description Format (XCCDF) version 1.2 (DRAFT)", NIST Interagency Report 7275, Revision 4, September 2011, <http://csrc.nist.gov/publications/nistir/ir7275-rev4/NISTIR-7275r4.pdf>.

[XMLDSIG] W3C Recommendation, "XML Signature Syntax and Processing (Second Edition)", June 2008, <http://www.w3.org/TR/xmldsig-core/>.

Authors' Addresses

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

Phone: +80 423 27 5862
EMail: takeshi_takahashi@nict.go.jp

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

EMail: Kent_Landfield@McAfee.com

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

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