MILE Working Group
Internet-Draft
Intended status: Standards Track
Expires: May 8, 2014

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

Nov 4, 2013

# IODEF-extension for structured cybersecurity information
## draft-ietf-mile-sci-10.txt

Abstract

This document extends the Incident Object Description Exchange Format (IODEF) defined in RFC 5070 [RFC5070] to exchange enriched cybersecurity information among cybersecurity entities and facilitate their operations. It provides the capability of embedding structured information, such as identifier- and XML-based information.

Status of this Memo



Copyright Notice

Table of Contents

## 1. Introduction

The number of cyber attacks is growing day-by-day, and 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 cybersecurity 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 occuring. Further structure within IODEF increases the machine-readability of the document thus providing a means for better automating certain cybersecurity operations.

Within the security community there exist various means for specifying structured descriptions of cybersecurity information such as [CAPEC][CCE][CCSS][CEE][CPE][CVE][CVRF][CVSS][CWE][CWSS][MAEC][OCIL][OVAL][SCAP][XCCDF]. Such structured descriptions facilitates a better understanding of an incident while enabling more streamlined automated cybersecurity operations. Because of this, it would be beneficial to embed and convey these types of information inside IODEF documents.

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

## 2. Terminology

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

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

## 3. Applicability

To maintain cybersecurity, organization needs to exchange cybersecurity information, which includes the following information:

attack pattern, platform information, vulnerability and weakness, countermeasure instruction, computer event log, and the severity. IODEF provides a scheme to describe and exchange such information among interested parties. However, it does not define the detailed formats to specify such information.

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

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

In another example, an administrator wishes to check the configuration of host computers in his organization. He could send a query to software on the host computers which could automatically generate an XML-based software configuration description, embed it in an IODEF document, and send the resulting IODEF document back to the administrator for review and additional automated uses.

## 4. Extension Definition

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

### 4.1. IANA Table for Structured Cybersecurity Information

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

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

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

```
Namespace:           http://xml/metadataSharing.xsd
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 to be managed by IANA following the allocation policy specified in Section 7.

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

### 4.2. Extended Data Type: XMLDATA

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

### 4.3. Extending IODEF

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

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

Figure 1: Incident class

### 4.4. Basic Structure of the Extension Classes

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

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

Figure 2: Basic Structure

Three attributes are defined as below.

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

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

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

Likewise, three elements are defined as below.

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

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

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

### 4.5. Defining Extension Classes

This draft defines the following seven extension classes.

#### 4.5.1. AttackPattern

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

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

Figure 3: AttackPattern class

This class has the following attributes.

SpecID: REQUIRED. ENUM. See Section 4.4.

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

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

Likewise, this class has the following elements.

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

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

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

### 4.5.2. Platform

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

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

Figure 4: Platform class

This class has the following attributes.

SpecID: REQUIRED. ENUM. See Section 4.4.

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

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

Likewise, this class has the following elements.

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

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

### 4.5.3. Vulnerability

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

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

Figure 5: Vulnerability class

This class has the following attributes.

SpecID: REQUIRED. ENUM. See Section 4.4.

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

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

Likewise, this class has the following elements.

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

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

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

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

### 4.5.4. Scoring

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

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

Figure 6: Scoring class

This class has two attributes.

SpecID: REQUIRED. ENUM. See Section 4.4.

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

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

Likewise, this class has the following elements.

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

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

### 4.5.5. Weakness

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

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

Figure 7: Weakness class

This class has the following attributes.

SpecID: REQUIRED. ENUM. See Section 4.4.

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

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

Likewise, this class has the following elements.

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

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

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

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

### 4.5.6. EventReport

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

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

Figure 8: EventReport class

This class has the following attributes.

SpecID: REQUIRED. ENUM. See Section 4.4.

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

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

Likewise, this class has the following elements.

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

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

#### 4.5.7. Verification

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

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

Figure 9: Verification class

This class has the following attributes.

SpecID: REQUIRED. ENUM. See Section 4.4.

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

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

Likewise, this class has the following elements.

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

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

#### 4.5.8. Remediation

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

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

Figure 10: Remediation class

This class has the following attributes.

SpecID: REQUIRED. ENUM. See Section 4.4.

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

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

Likewise, this class has the following elements.

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

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

## 5. Mandatory to Implement features

The implementation of this draft MUST be capable of sending and receiving the XML conforming to the specification listed in the initial IANA table described in Section 4.1 without error. The receiver MUST be capable of validating received XML documents that are embedded inside that against their schemata. Note that the receiver can look up the namespace in the IANA table to understand what specifications the embedded XML documents follows.

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

### 5.1. An Example XML

An example IODEF document for checking implementation's MTI conformity is provided here. The document carries MMDEF metadata. Note that the metadata is generated by genMMDEF [MMDEF] with EICAR [EICAR] files. Implementations of this specification must be capable of parsing the example XML since MMDEF is specified as the draft's MTI specification.


```
<?xml version="1.0" encoding="UTF-8"?>
<IODEF-Document version="1.00" lang="en"
 xmlns="urn:ietf:params:xml:ns:iodef-1.0"
 xmlns:iodef="urn:ietf:params:xml:ns:iodef-1.0"
 xmlns:iodef-sci="urn:ietf:params:xml:ns:iodef-sci-1.0"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <Incident purpose="reporting">
    <IncidentID name="iodef-sci.example.com">189493</IncidentID>
    <ReportTime>2013-06-18T23:19:24+00:00</ReportTime>
    <Description>a candidate security incident</Description>
    <Assessment>
      <Impact completion="failed" type="admin" />
    </Assessment>
    <Method>
      <Description>A candidate attack event</Description>
      <AdditionalData dtype="xml">
        <iodef-sci:AttackPattern
         SpecID="http://xml/metadataSharing.xsd">
```

```
          <iodef-sci:RawData dtype="xml">
            <malwareMetaData xmlns="http://xml/metadataSharing.xsd"
             xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
             xsi:schemaLocation="http://xml/metadataSharing.xsd
             file:metadataSharing.xsd" version="1.200000" id="10000">
              <company>N/A</company>
              <author>MMDEF Generation Script</author>
              <comment>Test MMDEF v1.2 file generated using genMMDEF
              </comment>
              <timestamp>2013-03-23T15:12:50.726000</timestamp>
              <objects>
                <file id="6ce6f415d8475545be5ba114f208b0ff">
                  <md5>6ce6f415d8475545be5ba114f208b0ff</md5>
                  <sha1>da39a3ee5e6b4b0d3255bfef95601890afd80709</sha1>
                  <sha256>e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca4
                          95991b7852b855</sha256>
                  <sha512>cf83e1357eefb8bdf1542850d66d8007d620e4050b5715dc83
                          f4a921d36ce9ce47d0d13c5d85f2b0ff8318d2877eec2f63b9
                          31bd47417a81a538327af927da3e</sha512>
                  <size>184</size>
                  <filename>eicar_com.zip</filename>
                  <MIMEType>application/zip</MIMEType>
                </file>
                <file id="44d88612fea8a8f36de82e1278abb02f">
                  <md5>44d88612fea8a8f36de82e1278abb02f</md5>
                  <sha1>3395856ce81f2b7382dee72602f798b642f14140</sha1>
                  <sha256>275a021bbfb6489e54d471899f7db9d1663fc695ec2fe2a2c4
                          538aabf651fd0f</sha256>
                  <sha512>cc805d5fab1fd71a4ab352a9c533e65fb2d5b885518f4e565e
                          68847223b8e6b85cb48f3afad842726d99239c9e36505c64b0
                          dc9a061d9e507d833277ada336ab</sha512>
                  <size>68</size>
                  <crc32>1750191932</crc32>
                  <filename>eicar.com</filename>
                  <filenameWithinInstaller>eicar.com
                  </filenameWithinInstaller>
                </file>
              </objects>
            <relationships>
              <relationship type="createdBy" id="1">
                <source>
                  <ref>file[@id="6ce6f415d8475545be5ba114f208b0ff"]</ref>
                </source>
                <target>
                  <ref>file[@id="44d88612fea8a8f36de82e1278abb02f"]</ref>
                </target>
                <timestamp>2013-03-23T15:12:50.744000</timestamp>
                </relationship>
```

```
              </relationships>
            </malwareMetaData>
          </iodef-sci:RawData>
        </iodef-sci:AttackPattern>
      </AdditionalData>
    </Method>
    <Contact role="creator" type="organization">
      <ContactName>iodef-sci.example.com</ContactName>
      <RegistryHandle registry="arin">iodef-sci.example-com
      </RegistryHandle>
      <Email>contact@csirt.example.com</Email>
    </Contact>
    <EventData>
      <Flow>
        <System category="source">
          <Node>
            <Address category="ipv4-addr">192.0.2.200</Address>
            <Counter type="event">57</Counter>
          </Node>
        </System>
        <System category="target">
          <Node>
            <Address category="ipv4-net">192.0.2.16/28</Address>
          </Node>
          <Service ip_protocol="4">
            <Port>80</Port>
          </Service>
        </System>
      </Flow>
      <Expectation action="block-host" />
      <Expectation action="other" />
    </EventData>
  </Incident>
</IODEF-Document>
```


## 5.2. An XML Schema for the Extension

An XML Schema describing the elements defined in this draft is given here. Any XMLs compliant to this draft including the ones in Section 5.1 should be verified against this schema by automated tools.


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

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

```
 elementFormDefault="qualified" attributeFormDefault="unqualified">

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

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

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

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

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

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

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

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

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

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

  <xsd:element name="Verification">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:choice>
```

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

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

</xsd:schema>
```

## 6. Security Considerations

This document specifies a format for encoding a particular class of security incidents appropriate for exchange across organizations. As merely a data representation, it does not directly introduce security issues. However, it is guaranteed that parties exchanging instances of this specification will have certain concerns. For this reason, the underlying message format and transport protocol used MUST ensure the appropriate degree of confidentiality, integrity, and authenticity for the specific environment.

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

concern.

## 6.1. Transport-Specific Concerns

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

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

# 7. IANA Considerations

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

Registration request for the IODEF structured cybersecurity information extension namespace:

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

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

XML: None

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

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

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

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

This memo creates the following registry for IANA to manage:

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

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

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

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

Namespace: A URI [RFC3986] that is the XML namespace name used by the registered SCI specification.

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

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

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

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

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

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

Namespace: http://xml/metadataSharing.xsd

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 WG exists (e.g., via email to the working group's mailing list). The Designated Expert is expected to retrieve the SCI specification from the provided URI in order to check the public availability of the specification and verify the correctness of the URI. An important responsibility of the Designated Expert is to ensure that the registered Applicable Classes are appropriate for the registered SCI specification.

## 8. Acknowledgment

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

## 9. References

### 9.1. Normative References

[MMDEF] IEEE ICSG Malware Metadata Exchange Format Working Group, "Malware Metadata Exchange Format".

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

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

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

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

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

[RFC6046] Moriarty, K. and B. Trammell, "Transport of Real-time Inter-network Defense (RID) Messages", RFC 6046, November 2010.

[XML1.0] Bray, T., Maler, E., Paoli, J., Sperberg-McQueen, C., and F. Yergeau, "Extensible Markup Language (XML) 1.0 (Fifth Edition)", W3C Recommendation, November 2008.

[XMLschemaPart1] Thompson, H., Beech, D., Maloney, M., and N. Mendelsohn, "XML Schema Part 1: Structures Second Edition", W3C Recommendation, October 2004.

[XMLschemaPart2] Biron, P. and A. Malhotra, "XML Schema Part 2: Datatypes Second Edition", W3C Recommendation, October 2004.

[XMLNames] Bray, T., Hollander, D., Layman, A., Tobin, R., and H. Thomson, ""Namespaces in XML (Third Edition)", W3C Recommendation, December 2009.

### 9.2. Informative References

[RFC3339] Klyne, G., Ed. and C. Newman, "Date and Time on the Internet: Timestamps", RFC 3339, July 2002.

[RFC3552] Rescorla, E. and B. Korver, "Guidelines for Writing RFC Text on Security Considerations", BCP 72, RFC 3552, July 2003.

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

[RFC5322] Resnick, P., Ed., "Internet Message Format", RFC 5322, October 2008.

[RFC6116] Bradner, S., Conroy, L., and K. Fujiwara, "The E.164 to Uniform Resource Identifiers (URI) Dynamic Delegation Discovery System (DDDS) Application (ENUM)", RFC 6116, March 2011.

[CAPEC] The MITRE Corporation, "Common Attack Pattern Enumeration and Classification (CAPEC)".

[CCE] The MITRE Corporation, "Common Configuration Enumeration (CCE)".

[CCSS] Scarfone, K. and P. Mell, "The Common Configuration Scoring System (CCSS)", NIST Interagency Report 7502, December 2010.

[CEE] The MITRE Corporation, "Common Event Expression (CEE)".

[CPE] National Institute of Standards and Technology, "Common Platform Enumeration", June 2011.

[CVE] The MITRE Corporation, "Common Vulnerability and Exposures (CVE)".

[CVRF] ICASI, "Common Vulnerability Reporting Framework (CVRF)".

[CVSS] Peter Mell, Karen Scarfone, and Sasha Romanosky, "The Common Vulnerability Scoring System (CVSS) and Its Applicability to Federal Agency Systems".

[CWE] The MITRE Corporation, "Common Weakness Enumeration (CWE)".

[CWSS] The MITRE Corporation, "Common Weakness Scoring System (CWSS)".

[EICAR] European Expert Group for IT-Security, "Anti-Malware Testfile", 2003.

[MAEC] The MITRE Corporation, "Malware Attribute Enumeration and Characterization".

[OCIL] David Waltermire and Karen Scarfone and Maria Casipe, "The Open Checklist Interactive Language (OCIL) Version 2.0", April 2011.

[OVAL] The MITRE Corporation, "Open Vulnerability and Assessment Language (OVAL)".

[SCAP] Waltermire, D., Quinn, S., Scarfone, K., and A. Halbardier, "The Technical Specification for the Security Content Automation Protocol (SCAP): SCAP Version 1.2", NIST Special Publication 800-126 Revision 2, September 2011.

[XCCDF] David Waltermire and Charles Schmidt and Karen Scarfone and Neal Ziring, "Specification for the Extensible Configuration Checklist Description Format (XCCDF) version 1.2 (DRAFT)", July 2011.

## Authors' Addresses

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

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

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

Email: Kent_Landfield@McAfee.com

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

Phone: +1 888 282 0870
Email: thomas.millar@us-cert.gov

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

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