MILE Working Group
Internet-Draft
Intended status: Standards Track
Expires: January 5, 2014

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

Jul 4, 2013

# IODEF-extension for structured cybersecurity information
# draft-ietf-mile-sci-08.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

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

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

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

This Internet-Draft will expire on January 5, 2014.

## 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 enabling such exchange, and is already in use.

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

On the other hand, there exist various other activities facilitating detailed and structured description of cybersecurity information, as listed in Section 9 . Since such structured description facilitates cybersecurity operations, it would be beneficial to embed and convey these information inside IODEF document.

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

## 2. Terminology

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

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

## 3. Applicability

To maintain cybersecurity, organization needs to exchange cybersecurity information, which includes the following information: attack pattern, platform information, vulnerability and weakness, countermeasure instruction, computer event log, and the severity.

IODEF provides a scheme to describe and exchange such information among interested parties. However, it does not define the detailed format to describe such information.

On the other hand, there already exist structured and detailed formats for describing those information and facilitating such exchange. Major of them are listed in Section 9 . By embedding them into the IODEF document, the document can convey more detailed contents to the receivers, and the document can be easily reused.

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

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

Another example is that, when an administrator wishes to check the configuration of host computers in his organization, he may send a query to host computers, which may automatically generate XML-based software configuration information upon receiving thequery by running a software and may embed that to an IODEF document, which is then sent back to the administrator.

## 4. Extension Definition

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

### 4.1. IANA Table for Structured Cybersecurity Information

This extension embeds structured cybersecurity information defined by the 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 |
| 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 using the Expert Review [RFC5226] and Specification Required [RFC5226] allocation policies as further specified in Section 7 .

The SpecID attributes of extended classes (Section 4.3) 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 by using any means defined outside the scope of this specification.

## 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. Extended Classes

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


Figure 1: Incident class

This extension defines the following seven elements.

### 4.3.1. AttackPattern

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

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

An AttackPattern class is structured as follows.

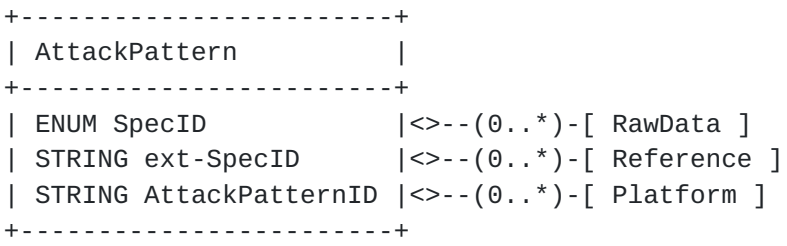


Figure 2: AttackPattern class

This class has the following attributes.

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 RawData 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. When this element is used, the value of SpecID element must be "private."

AttackPatternID: OPTIONAL. STRING. An identifier of an attack pattern to be reported. This attribute SHOULD be used whenever such identifier is available. Both RawData and Reference elements MUST NOT be used when this attribute is used, while either of them MUST be used if this attribute is omitted.

The AttackPattern class is composed of the following aggregate classes.

RawData: Zero or more. XMLDATA. 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 prefered, and the inconsistency SHOULD be logged so a human can correct the problem.

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

Platform: Zero or more. An identifier of software platform involved in the specific attack pattern, which is elaborated in Section 4.3.2 .

### 4.3.2. Platform

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

A Platform element is structured as follows.

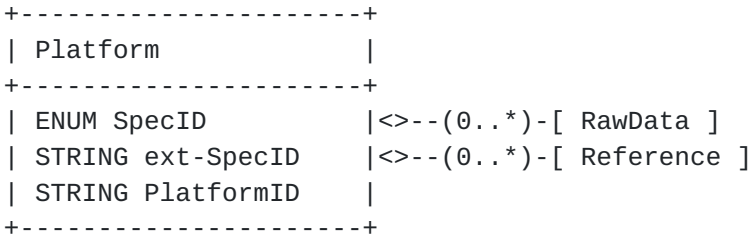


Figure 3: Platform class

This class has the following attributes.

SpecID: REQUIRED. ENUM. The meaning of this attribute is the same as that of the AttackPattern class (Section 4.3.1) .

ext-SpecID: OPTIONAL. STRING. The meaning of this attribute is the same as that of the AttackPattern class (Section 4.3.1) .

PlatformID: OPTIONAL. STRING. An identifier of a platform to be reported. This attribute SHOULD be used whenever such identifier is available. Both RawData and Reference elements MUST NOT be used when this attribute is used, while either of them MUST be used if this attribute is omitted.

This class is composed of the following aggregate classes.

RawData: Zero or more. XMLDATA. The meaning of this element is the same as that of the AttackPattern class (Section 4.3.1) .

Reference: Zero or more of iodef:Reference [RFC5070] . The meaning of this element is the same as that of the AttackPattern class (Section 4.3.1) .

### 4.3.3. Vulnerability

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

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

A Vulnerability element is structured as follows.

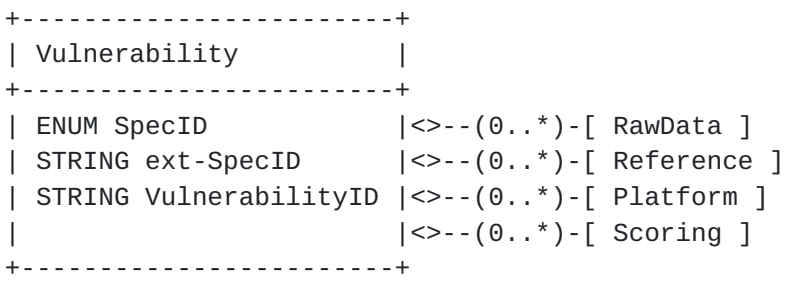


Figure 4: Vulnerability class

This class has the following attributes.

SpecID: REQUIRED. ENUM. The meaning of this attribute is the same as that of the AttackPattern class (Section 4.3.1) .

ext-SpecID: OPTIONAL. STRING. The meaning of this attribute is the same as that of the AttackPattern class (Section 4.3.1) .

VulnerabilityID: OPTIONAL. STRING. An identifier of a vulnerability to be reported. This attribute SHOULD be used whenever such identifier is available. Both RawData and Reference elements MUST NOT be used when this attribute is used, while either of them MUST be used if this attribute is omitted.

This class is composed of the following aggregate classes.

RawData: Zero or more. XMLDATA. The meaning of this element is the same as that of the AttackPattern class (Section 4.3.1) .

Reference: Zero or more of iodef:Reference [RFC5070] . The meaning of this element is the same as that of the AttackPattern class (Section 4.3.1) .

Platform: Zero or more. An identifier of software platform affected by the vulnerability, which is elaborated in Section 4.3.2 .

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

### 4.3.4. Scoring

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

A Scoring class is structured as follows.

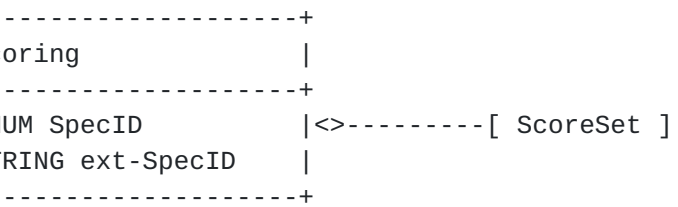

Figure 5: Scoring class

This class has two attributes.

SpecID: REQUIRED. ENUM. The meaning of this attribute is the same as that of the AttackPattern class (Section 4.3.1) .

ext-SpecID: OPTIONAL. STRING. The meaning of this attribute is the same as that of the AttackPattern class (Section 4.3.1) .

This class is composed of an aggregate class.

ScoreSet: One. XMLDATA. A complete document that is formatted according to the specification and its version identified by the SpecID/ext-SpecID. This element includes a set of score information. 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 prefered, and the inconsistency SHOULD be logged so a human can correct the problem.

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

### 4.3.5. Weakness

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

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

A Weakness element is structured as follows.

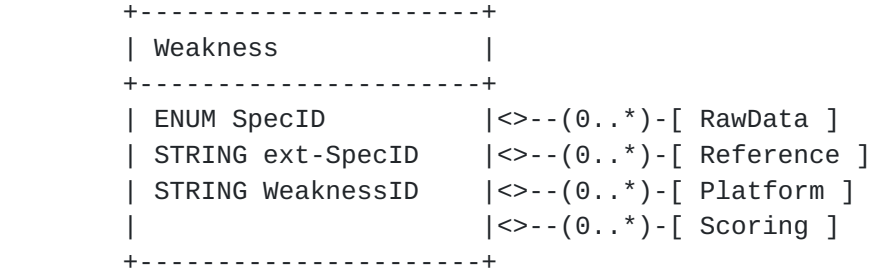

Figure 6: Weakness class

This class has the following attributes.

SpecID: REQUIRED. ENUM. The meaning of this attribute is the same as that of the AttackPattern class (Section 4.3.1) .

ext-SpecID: OPTIONAL. STRING. The meaning of this attribute is the same as that of the AttackPattern class (Section 4.3.1) .

WeaknessID: OPTIONAL. STRING. An identifier of a weakness to be reported. This attribute SHOULD be used whenever such identifier is available/ Both RawData and Reference elements MUST NOT be used when this attribute is used, while either of them MUST be used if this attribute is omitted.

This class is composed of the following aggregate classes.

RawData: Zero or more. XMLDATA. The meaning of this element is the same as that of the AttackPattern class (Section 4.3.1) .

Reference: Zero or more of iodef:Reference [RFC5070] . The meaning of this element is the same as that of the AttackPattern class (Section 4.3.1) .

Platform: Zero or more. An identifier of software platform affected by the weakness, which is elaborated in Section 4.3.2 .

Scoring: Zero or more. An indicator of the severity of the weakness, such as CWSS score, which is elaborated in Section 4.3.4 .

### 4.3.6. EventReport

An EventReport consists of an extension to the Incident.EventData.Record.RecordData.RecordItem element with a dtype of "xml". The extension embeds structured event reports.

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

An EventReport element is structured as follows.

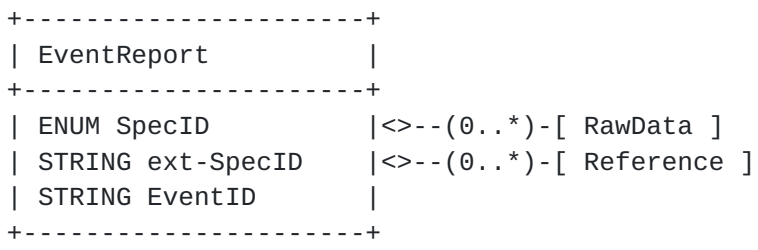


Figure 7: EventReport class

This class has the following attributes.

SpecID: REQUIRED. ENUM. The meaning of this attribute is the same as that of the AttackPattern class (Section 4.3.1) .

ext-SpecID: OPTIONAL. STRING. The meaning of this attribute is the same as that of the AttackPattern class (Section 4.3.1) .

EventID: OPTIONAL. STRING. An identifier of an event to be reported. This attribute SHOULD be used whenever such identifier is available. Both RawData and Reference elements MUST NOT be used when this attribute is used, while either of them MUST be used if this attribute is omitted.

This class is composed of three aggregate classes.

RawData: Zero or more. XMLDATA. The meaning of this element is the same as that of the AttackPattern class (Section 4.3.1) .

Reference: Zero or more of iodef:Reference [RFC5070] . The meaning of this element is the same as that of the AttackPattern class (Section 4.3.1) .

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

### 4.3.7. Verifcation

A Verification consists of an extension to the Incident.AdditionalData element with a dtype of "xml". The extension elements describes incident on vefifying incidents.

A Verification class is structured as follows.

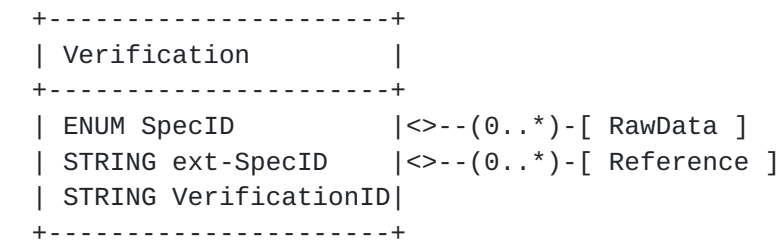


Figure 8: Verification class

This class has the following attributes.

SpecID: REQUIRED. ENUM. The meaning of this attribute is the same as that of the AttackPattern class (Section 4.3.1) .

ext-SpecID: OPTIONAL. STRING. The meaning of this attribute is the same as that of the AttackPattern class (Section 4.3.1) .

VerificationID: OPTIONAL. STRING. An identifier of an check item to be reported. This attribute SHOULD be used whenever such identifier is available. Both RawData and Reference elements MUST NOT be used when this attribute is used, while either of them MUST be used if this attribute is omitted.

This class is composed of two aggregate classes.

RawData: Zero or more. XMLDATA. The meaning of this element is the same as that of the AttackPattern class (Section 4.3.1) .

Reference: Zero or more of iodef:Reference [RFC5070] . The meaning of this element is the same as that of the AttackPattern class (Section 4.3.1) .

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

### 4.3.8. Remediation

A Remediation consists of an extension to the Incident.AdditionalData element with a dtype of "xml". The extension elements describes incident remediation information including instructions.

It is recommended that Incident class SHOULD contain one or more of

this extension elements whenever available.

A Remediation class is structured as follows.

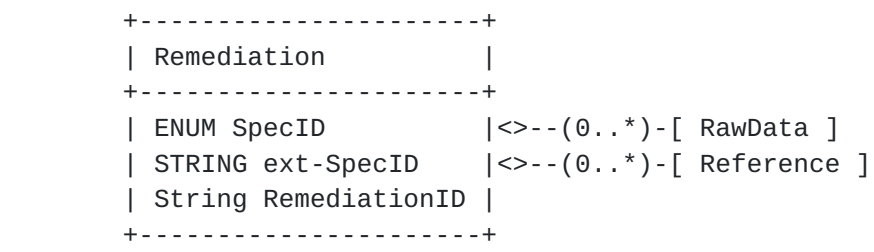


Figure 9: Remediation class

This class has the following attributes.

SpecID: REQUIRED. ENUM. The meaning of this attribute is the same as that of the AttackPattern class (Section 4.3.1) .

ext-SpecID: OPTIONAL. STRING. The meaning of this attribute is the same as that of the AttackPattern class (Section 4.3.1) .

RemediationID: OPTIONAL. STRING. An identifier of a remediation information to be reported. This attribute SHOULD be used whenever such identifier is available. Both RawData and Reference elements MUST NOT be used when this attribute is used, while either of them MUST be used if this attribute is omitted.

This class is composed of two aggregate classes.

RawData: Zero or more. XMLDATA. The meaning of this element is the same as that of the AttackPattern class (Section 4.3.1) .

Reference: Zero or more of iodef:Reference [RFC5070] . The meaning of this element is the same as that of the AttackPattern class (Section 4.3.1) .

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

## 5. 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, i.e., MMDEF version 1.2, without error.

The receiver MUST be capable of validating received XML documents that are embedeed 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.

This section provides an XML comformant to this draft, and a scehma for that.

Note that the schema of MMDEF version 1.2 is found in Section 10.

### 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>e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b9
                     34ca495991b7852b855</sha256>
             <sha512>cf83e1357eefb8bdf1542850d66d8007d620e4050b571
                     5dc83f4a921d36ce9ce47d0d13c5d85f2b0ff8318d287
                     7eec2f63b931bd47417a81a538327af927da3e
             </sha512>
             <size>184</size>
             <filename>eicar_com.zip</filename>
             <MIMEType>application/zip</MIMEType>
           </file>
           <file id="44d88612fea8a8f36de82e1278abb02f">
             <md5>44d88612fea8a8f36de82e1278abb02f</md5>
             <sha1>3395856ce81f2b7382dee72602f798b642f14140</sha1>
             <sha256>275a021bbfb6489e54d471899f7db9d1663fc695ec2fe
                     2a2c4538aabf651fd0f</sha256>
             <sha512>cc805d5fab1fd71a4ab352a9c533e65fb2d5b885518f4
                     e565e68847223b8e6b85cb48f3afad842726d99239c9e
                     36505c64b0dc9a061d9e507d833277ada336ab
             </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>
    </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 and Section 11 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:element name="ScoreSet" type="iodef-sci:XMLDATA"
         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: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="AttackPatternID" 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="VulnerabilityID" 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="WeaknessID" 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="PlatformID" 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="EventID" 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="VerificationID" 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="RemediationID" 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 the appendix of the document.

This memo creates the following registry for IANA to manage:

Name of the registry: "IODEF Structured Cyber Security Information Specifications"

Namespace details: A registry entry for a Structured Cyber Security 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 Extended Classes specified in Section 4.3 of this document. The registered SCI specification MUST only be used with the Extended 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/Applicable Classes.

Initial registry contents: none

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

The Designated Expert is expected to consult with the mile (Managed Incident Lightweight Exchange) working group or its successor if any such WG exists (e.g., via email to the working group's mailing list). The Designated Expert is expected to retrieve the 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 Mr. 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, Paul Cichonski from NIST, Panos Kampanakis from CISCO, Ivan Kirillov from MITRE, Robert Martin from MITRE, Kathleen Moriarty from EMC, Lagadec Philippe from NATO, Shuhei Yamaguchi from NICT, Anthony Rutkowski from Yaana Technology, Brian Trammel from CERT/NetSA, David Waltermire from NIST, and James Wendorf from IEEE, for their sincere discussion and feedback on this document.

## 9. Appendix I: Candidate Specifications for the IANA Table

This draft defined the structure of the IANA table in Section 4.1 . Though the management of the table is up to IANA, this appendix provides candidate entries. Note that OVAL and CVE are registered trademarks, and CAPEC, CCE, CEE, CPE, CWE, CWSS, MAEC, and OCIL are trademarks, of The MITRE Corporation.

1. CAPEC 1.6

| | |
|---|---|
| Namespace: | http://capec.mitre.org/observables |
| Specification Name: | Common Attack Pattern Enumeration and Classification |
| Version: | 1.6 |
| Reference URI: | http://capec.mitre.org/ |
| Applicable Classes: | AttackPattern |

2. CCE 5.0

| | |
|---|---|
| Namespace: | http://cce.mitre.org |
| Specification Name: | Common Configuration Enumeration |
| Version: | 5.0 |
| Reference URI: | http://cce.mitre.org/ |
| Applicable Classes: | Verification |

3. CCSS 1.0

| | |
|---|---|
| Namespace: | N/A |
| Specification Name: | Common Configuration Scoring System |
| Version: | 1.0 |
| Reference URI: | http://csrc.nist.gov/publications/PubsNISTIRs.html #NIST-IR-7502 |
| Applicable Classes: | Scoring |

## 4. CEE 1.0 alpha

Namespace: http://cee.mitre.org
Specification Name: Common Event Expression
Version: 1.0 alpha
Reference URI: http://cee.mitre.org/
Applicable Classes: EventReport

## 5. CPE 2.3 Language

Namespace: http://cpe.mitre.org/language/2.0
Specification Name: Common Platform Enumeration Reference
Version: 2.3
Reference URI: http://scap.nist.gov/specifications/cpe/, http://csrc.nist.gov/publications/PubsNISTIRs.html #NIST-IR-7695
Applicable Classes: Platform

## 6. CPE 2.3 Dictionary

Namespace: http://cpe.mitre.org/dictionary/2.0
Specification Name: Common Platform Enumeration Dictionary
Version: 2.3
Reference URI: http://scap.nist.gov/specifications/cpe/, http://csrc.nist.gov/publications/PubsNISTIRs.html #NIST-IR-7697
Applicable Classes: Platform

## 7. CVE 1.0

Namespace: http://cve.mitre.org/cve/downloads/1.0
Specification Name: Common Vulnerability and Exposures
Version: 1.0
Reference URI: http://cve.mitre.org/
Applicable Classes: Vulnerability

## 8. CVRF 1.0

Namespace: http://www.icasi.org/CVRF/schema/cvrf/1.0
Specification Name: Common Vulnerability Reporting Format
Version: 1.0
Reference URI: http://www.icasi.org/cvrf
Applicable Classes: Vulnerability

## 9. CVSS 2.0

Namespace: http://scap.nist.gov/schema/cvss-v2/1.0
Specification Name: Common Vulnerability Scoring System
Version: 2
Reference URI: http://www.first.org/cvss
Applicable Classes: Scoring

## 10. CWE 5.0

Namespace: N/A
Specification Name: Common Weakness Enumeration
Version: 5.1
Reference URI: http://cwe.mitre.org/
Applicable Classes: Weakness

## 11. CWSS 0.8

Namespace: N/A
Specification Name: Common Weakness Scoring System
Version: 0.8
Reference URI: http://cwe.mitre.org/cwss/
Applicable Classes: Scoring

## 12. MAEC 2.0

Namespace: http://maec.mitre.org/XMLSchema/maec-core-2
Specification Name: Malware Attribute Enumeration and Characterization
Version: 2.0
Reference URI: http://maec.mitre.org/
Applicable Classes: EventReport, AttackPattern

## 13. OCIL 2.0

Namespace: http://scap.nist.gov/schema/ocil/2.0
Specification Name: Open Checklist Interactive Language
Version: 2.0
Reference URI: http://scap.nist.gov/specifications/ocil/, http://csrc.nist.gov/publications/PubsNISTIRs.html #NIST-IR-7692
Applicable Classes: Verification

## 14. OVAL 5.10.1 Definitions

Namespace: http://oval.mitre.org/XMLSchema/oval-definitions-5
Specification Name: Open Vulnerability and Assessment Language
Version: 5.10.1
Reference URI: http://oval.mitre.org/
Applicable Classes: Verification

## 15. OVAL 5.10.1 Results

Namespace: http://oval.mitre.org/XMLSchema/oval-results-5
Specification Name: Open Vulnerability and Assessment Language
Version: 5.10.1
Reference URI: http://oval.mitre.org/
Applicable Classes: Verification

## 16. OVAL 5.10.1 Common

Namespace: http://oval.mitre.org/XMLSchema/oval-common-5
Specification Name: Open Vulnerability and Assessment Language
Version: 5.10.1
Reference URI: http://oval.mitre.org/
Applicable Classes: Verification

## 17. XCCDF 1.2

| | |
|---|---|
| Namespace: | http://checklists.nist.gov/xccdf/1.2 |
| Specification Name: | Extensible Configuration Checklist Description Format |
| Version: | 1.2 |
| Reference URI: | http://scap.nist.gov/specifications/xccdf/, http://csrc.nist.gov/publications/PubsNISTIRs.html #NIST-IR-7275-r4 |
| Applicable Classes: | Verification |

## 10. Appendix II: MMDEF version 1.2 scheema

This draft identified MMDEF version 1.2 as its MTI, and its schema (without annotations) is provided here.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
 targetNamespace="http://xml/metadataSharing.xsd"
 xmlns="http://xml/metadataSharing.xsd"
 elementFormDefault="qualified" version="1.8">

 <xs:simpleType name="intBetween0and100">
  <xs:restriction base="xs:integer">
   <xs:minInclusive value="0"/>
   <xs:maxInclusive value="100"/>
  </xs:restriction>
 </xs:simpleType>

 <xs:simpleType name="NoQuestionMark">
  <xs:restriction base="xs:string">
   <xs:pattern value="[^?]+"/>
  </xs:restriction>
 </xs:simpleType>

 <xs:simpleType name="IPRange">
  <xs:restriction base="xs:string">
   <xs:pattern value="[0-9\.]+-[0-9\.]+"/>
  </xs:restriction>
 </xs:simpleType>

 <xs:simpleType name="RelationshipTypeEnum">
  <xs:restriction base="xs:string">
   <xs:enumeration value="relatedTo"/>
   <xs:enumeration value="isClassifiedAs"/>
   <xs:enumeration value="hosts"/>
   <xs:enumeration value="installed"/>
```

```
   <xs:enumeration value="isParentOf"/>
   <xs:enumeration value="causesToInstall"/>
   <xs:enumeration value="downloads"/>
   <xs:enumeration value="runs"/>
   <xs:enumeration value="usesCNC"/>
   <xs:enumeration value="isNameServerOf"/>
   <xs:enumeration value="resolvesTo"/>
   <xs:enumeration value="verifiedBy"/>
   <xs:enumeration value="isServerOfService"/>
   <xs:enumeration value="hasAssociatedConfiguration"/>
   <xs:enumeration value="operatedByEntity"/>
   <xs:enumeration value="downloadedFrom"/>
   <xs:enumeration value="contactedBy"/>
   <xs:enumeration value="partOfPackage"/>
   <xs:enumeration value="sourcedFrom"/>
   <xs:enumeration value="createdBy"/>
   <xs:enumeration value="hasSignature"/>
   <xs:enumeration value="hasTaggant"/>
  </xs:restriction>
 </xs:simpleType>

 <xs:simpleType name="ClassificationTypeEnum">
  <xs:restriction base="xs:string">
   <xs:enumeration value="clean"/>
   <xs:enumeration value="dirty"/>
   <xs:enumeration value="unknown"/>
   <xs:enumeration value="unwanted"/>
   <xs:enumeration value="neutral"/>
  </xs:restriction>
 </xs:simpleType>

 <xs:simpleType name="LocationTypeEnum">
  <xs:restriction base="xs:string">
   <xs:enumeration value="countryCodeISO3166-2"/>
   <xs:enumeration value="countryCodeISO3166-3"/>
   <xs:enumeration value="countryCodeFIPS"/>
   <xs:enumeration value="city"/>
   <xs:enumeration value="region"/>
   <xs:enumeration value="isp"/>
  </xs:restriction>
 </xs:simpleType>

 <xs:simpleType name="VolumeUnitsEnum">
  <xs:restriction base="xs:string">
   <xs:enumeration value="numberUsersAffected"/>
   <xs:enumeration value="numberMachinesAffected"/>
   <xs:enumeration value="numberSeenInSpam"/>
   <xs:enumeration value="numberSeenInMalwareSamples"/>
```

```
   <xs:enumeration value="numberOfWebsitesHosting"/>
   <xs:enumeration value="numberOfWebsitesRedirecting"/>
  </xs:restriction>
 </xs:simpleType>

 <xs:simpleType name="IPTypeEnum">
  <xs:restriction base="xs:string">
   <xs:enumeration value="ipv4"/>
   <xs:enumeration value="ipv6"/>
  </xs:restriction>
 </xs:simpleType>

 <xs:simpleType name="RegionTypeEnum">
  <xs:restriction base="xs:string">
   <xs:enumeration value="NorthAmerica"/>
   <xs:enumeration value="SouthAmerica"/>
   <xs:enumeration value="CentralAmerica"/>
   <xs:enumeration value="Europe"/>
   <xs:enumeration value="Africa"/>
   <xs:enumeration value="APAC"/>
  </xs:restriction>
 </xs:simpleType>

 <xs:simpleType name="OriginTypeEnum">
  <xs:restriction base="xs:string">
   <xs:enumeration value="user"/>
   <xs:enumeration value="desktop"/>
   <xs:enumeration value="lan"/>
   <xs:enumeration value="gateway"/>
   <xs:enumeration value="isp"/>
   <xs:enumeration value="honeypot"/>
   <xs:enumeration value="collection"/>
   <xs:enumeration value="spam"/>
   <xs:enumeration value="wan"/>
   <xs:enumeration value="internal"/>
   <xs:enumeration value="partner"/>
   <xs:enumeration value="unknown"/>
  </xs:restriction>
 </xs:simpleType>

 <xs:simpleType name="PropertyTypeEnum">
  <xs:restriction base="xs:string">
   <xs:enumeration value="filename"/>
   <xs:enumeration value="filepath"/>
   <xs:enumeration value="locationUrl"/>
   <xs:enumeration value="isKernel"/>
   <xs:enumeration value="isParasitic"/>
   <xs:enumeration value="isStealth"/>
```

```
   <xs:enumeration value="isPolymorphic"/>
   <xs:enumeration value="isVirus"/>
   <xs:enumeration value="isNonReplicating"/>
   <xs:enumeration value="isDamaged"/>
   <xs:enumeration value="registryValueData"/>
   <xs:enumeration value="urlParameterString"/>
   <xs:enumeration value="postData"/>
   <xs:enumeration value="registrant"/>
   <xs:enumeration value="registrationDate"/>
   <xs:enumeration value="ownerAddress"/>
   <xs:enumeration value="adminContact"/>
   <xs:enumeration value="technicalContact"/>
   <xs:enumeration value="nameServer"/>
   <xs:enumeration value="countryCodeISO3166-2"/>
   <xs:enumeration value="countryCodeISO3166-3"/>
   <xs:enumeration value="countryCodeFIPS"/>
   <xs:enumeration value="city"/>
   <xs:enumeration value="region"/>
   <xs:enumeration value="isp"/>
   <xs:enumeration value="httpMethod"/>
   <xs:enumeration value="referrer"/>
   <xs:enumeration value="operatingSystem"/>
   <xs:enumeration value="userAgent"/>
   <xs:enumeration value="browser"/>
   <xs:enumeration value="comment"/>
  </xs:restriction>
 </xs:simpleType>

 <xs:element name="malwareMetaData">
  <xs:complexType>
   <xs:sequence>
    <xs:element name="company" type="xs:string"/>
    <xs:element name="author" type="xs:string"/>
    <xs:element name="comment" type="xs:string"/>
    <xs:element name="timestamp" type="xs:dateTime"/>
    <xs:element name="objects" minOccurs="0">
     <xs:complexType>
      <xs:sequence>
       <xs:element name="file" type="fileObject" minOccurs="0"
        maxOccurs="unbounded"/>
       <xs:element name="uri" type="uriObject" minOccurs="0"
        maxOccurs="unbounded"/>
       <xs:element name="domain" type="domainObject" minOccurs="0"
        maxOccurs="unbounded"/>
       <xs:element name="registry" type="registryObject" minOccurs="0"
        maxOccurs="unbounded"/>
       <xs:element name="ip" type="IPObject" minOccurs="0"
        maxOccurs="unbounded"/>
```

```
      <xs:element name="asn" type="ASNObject" minOccurs="0"
       maxOccurs="unbounded"/>
      <xs:element name="entity" type="entityObject" minOccurs="0"
       maxOccurs="unbounded"/>
      <xs:element name="classification" type="classificationObject"
       minOccurs="0" maxOccurs="unbounded"/>
      <xs:element maxOccurs="unbounded" minOccurs="0"
       name="softwarePackage" type="softwarePackageObject"/>
      <xs:element maxOccurs="unbounded" minOccurs="0"
       name="digitalSignature" type="digitalSignatureObject"/>
      <xs:element maxOccurs="unbounded" minOccurs="0" name="taggant"
       type="taggantObject"/>
     </xs:sequence>
    </xs:complexType>
   </xs:element>

   <xs:element name="objectProperties" minOccurs="0">
    <xs:complexType>
     <xs:sequence>
      <xs:element name="objectProperty" type="objectProperty"
       maxOccurs="unbounded"/>
     </xs:sequence>
    </xs:complexType>
   </xs:element>

   <xs:element name="relationships" minOccurs="0">
    <xs:complexType>
     <xs:sequence>
      <xs:element name="relationship" type="relationship"
       maxOccurs="unbounded"/>
     </xs:sequence>
    </xs:complexType>
   </xs:element>

   <xs:element name="fieldData" minOccurs="0">
    <xs:complexType>
     <xs:sequence>
      <xs:element name="fieldDataEntry" type="fieldDataEntry"
       maxOccurs="unbounded"/>
     </xs:sequence>
    </xs:complexType>
   </xs:element>
  </xs:sequence>
  <xs:attribute use="required" name="version" type="xs:decimal"
   fixed="1.2"/>
  <xs:attribute use="required" name="id" type="xs:string"/>
 </xs:complexType>
</xs:element>
```

```
 <xs:complexType name="fileObject">
  <xs:sequence>
   <xs:element name="md5" type="xs:hexBinary" minOccurs="1"/>
   <xs:element name="sha1" type="xs:hexBinary" minOccurs="0"/>
   <xs:element name="sha256" type="xs:hexBinary" minOccurs="0"/>
   <xs:element name="sha512" type="xs:hexBinary" minOccurs="0"/>
   <xs:element name="size" type="xs:int" minOccurs="0"/>
   <xs:element name="crc32" type="xs:string" minOccurs="0"/>
   <xs:element name="fileType" type="xs:string" minOccurs="0"
    maxOccurs="unbounded"/>
   <xs:element name="extraHash" minOccurs="0" maxOccurs="unbounded">
    <xs:complexType>
     <xs:simpleContent>
      <xs:extension base="xs:string">
       <xs:attribute name="type" use="required" type="xs:string"/>
      </xs:extension>
     </xs:simpleContent>
    </xs:complexType>
   </xs:element>

   <xs:element maxOccurs="unbounded" minOccurs="0" name="filename"
    type="xs:string"/>
   <xs:element maxOccurs="unbounded" minOccurs="0"
    name="normalizedNativePath" type="xs:string"/>
   <xs:element maxOccurs="unbounded" minOccurs="0"
    name="filenameWithinInstaller" type="xs:string"/>
   <xs:element maxOccurs="unbounded" minOccurs="0"
    name="folderWithinInstaller" type="xs:string"/>
   <xs:element minOccurs="0" name="vendor" type="xs:string"/>
   <xs:element maxOccurs="unbounded" minOccurs="0"
    name="internalName" type="xs:string"/>
   <xs:element maxOccurs="unbounded" minOccurs="0" name="language"
    type="xs:string"/>
   <xs:element maxOccurs="1" minOccurs="0" name="productName"
    type="xs:string"/>
   <xs:element minOccurs="0" name="fileVersion" type="xs:string"/>
   <xs:element minOccurs="0" name="productVersion" type="xs:string"/>
   <xs:element minOccurs="0" name="developmentEnvironment"
    type="xs:string"/>
   <xs:element minOccurs="0" name="checksum" type="xs:hexBinary"/>
   <xs:element minOccurs="0" name="architecture" type="xs:string"/>
   <xs:element minOccurs="0" name="buildTimeDateStamp" type="xs:dateTime"/>
   <xs:element minOccurs="0" name="compilerVersion" type="xs:string"/>
   <xs:element minOccurs="0" name="linkerVersion" type="xs:float"/>
   <xs:element minOccurs="0" name="minOSVersionCPE" type="xs:string"/>
   <xs:element maxOccurs="1" minOccurs="0" name="numberOfSections"
    type="xs:int"/>
   <xs:element minOccurs="0" name="MIMEType" type="xs:string"/>
```

```
   <xs:element minOccurs="0" name="requiredPrivilege" type="xs:string"/>
   <xs:element minOccurs="0" name="digitalSignature"
    type="digitalSignatureObject"/>
   <xs:element maxOccurs="1" minOccurs="0" name="taggant"
    type="taggantObject"/>
  </xs:sequence>
  <xs:attribute name="id" use="required" type="xs:hexBinary"/>
 </xs:complexType>

 <xs:complexType name="registryObject">
  <xs:sequence>
   <xs:element name="key" type="xs:string"/>
   <xs:element name="valueName" type="xs:string" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="id" use="required" type="xs:string"/>
 </xs:complexType>

 <xs:complexType name="entityObject">
  <xs:sequence>
   <xs:element name="name" type="xs:string"/>
  </xs:sequence>
  <xs:attribute name="id" use="required" type="xs:string"/>
 </xs:complexType>

 <xs:complexType name="uriObject">
  <xs:sequence>
   <xs:element name="uriString" type="NoQuestionMark"/>
   <xs:element name="protocol" type="xs:string" minOccurs="0"/>
   <xs:element name="hostname" type="xs:string" minOccurs="0"/>
   <xs:element name="domain" type="xs:string" minOccurs="0"/>
   <xs:element name="port" type="xs:int" minOccurs="0"/>
   <xs:element name="path" type="xs:string" minOccurs="0"/>
   <xs:element name="ipProtocol" type="xs:string" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="id" use="required" type="NoQuestionMark"/>
 </xs:complexType>

 <xs:complexType name="IPObject">
  <xs:sequence>
   <xs:element name="startAddress" type="IPAddress"/>
   <xs:element name="endAddress" type="IPAddress"/>
  </xs:sequence>
  <xs:attribute name="id" use="required" type="IPRange"/>
 </xs:complexType>

 <xs:complexType name="IPAddress">
  <xs:simpleContent>
   <xs:extension base="xs:string">
```

```
    <xs:attribute name="type" type="IPTypeEnum" use="required"/>
   </xs:extension>
  </xs:simpleContent>
 </xs:complexType>

 <xs:complexType name="domainObject">
  <xs:sequence>
   <xs:element name="domain" type="xs:string"/>
  </xs:sequence>
  <xs:attribute name="id" use="required" type="xs:string"/>
 </xs:complexType>

 <xs:complexType name="ASNObject">
  <xs:sequence>
   <xs:element name="as-number" type="xs:int"/>
  </xs:sequence>
  <xs:attribute name="id" use="required" type="xs:int"/>
 </xs:complexType>

 <xs:complexType name="classificationObject">
  <xs:sequence>
   <xs:element name="classificationName" type="xs:string"> </xs:element>
   <xs:element name="companyName" type="xs:string"/>
   <xs:element name="category" minOccurs="0" type="xs:string"/>
   <xs:element name="classificationDetails" minOccurs="0">
    <xs:complexType>
     <xs:sequence>
      <xs:element name="definitionVersion" type="xs:string" minOccurs="0"/>
      <xs:element name="detectionAddedTimeStamp" type="xs:dateTime"
       minOccurs="0"/>
      <xs:element name="detectionShippedTimeStamp" type="xs:dateTime"
       minOccurs="0"/>
      <xs:element name="product" type="xs:string" minOccurs="0"/>
      <xs:element name="productVersion" type="xs:string" minOccurs="0"/>
     </xs:sequence>
    </xs:complexType>
   </xs:element>

  </xs:sequence>
  <xs:attribute name="id" type="xs:string" use="required"/>
  <xs:attribute name="type" type="ClassificationTypeEnum" use="required"/>
 </xs:complexType>

 <xs:complexType name="fieldDataEntry">
  <xs:sequence>
   <xs:element name="references">
    <xs:complexType>
     <xs:sequence>
```

```
      <xs:element name="ref" type="reference" minOccurs="1"
       maxOccurs="unbounded"/>
     </xs:sequence>
    </xs:complexType>
   </xs:element>
   <xs:element name="startDate" type="xs:dateTime"/>
   <xs:element name="endDate" type="xs:dateTime"/>
   <xs:element name="firstSeenDate" type="xs:dateTime" minOccurs="0"/>
   <xs:element name="origin" type="OriginTypeEnum"/>
   <xs:element name="commonality" type="intBetween0and100" minOccurs="0"/>
   <xs:element name="volume" minOccurs="0" maxOccurs="unbounded">
    <xs:complexType>
     <xs:simpleContent>
      <xs:extension base="xs:int">
       <xs:attribute name="units" type="VolumeUnitsEnum" use="required"/>
      </xs:extension>
     </xs:simpleContent>
    </xs:complexType>
   </xs:element>
   <xs:element name="importance" type="intBetween0and100" minOccurs="0"/>
   <xs:element name="location" minOccurs="0">
    <xs:complexType>
     <xs:simpleContent>
      <xs:extension base="xs:string">
       <xs:attribute name="type" type="LocationTypeEnum"/>
      </xs:extension>
     </xs:simpleContent>
    </xs:complexType>
   </xs:element>
  </xs:sequence>
 </xs:complexType>

 <xs:complexType name="reference">
  <xs:simpleContent>
   <xs:extension base="xs:string"> </xs:extension>
  </xs:simpleContent>
 </xs:complexType>

 <xs:complexType name="property">
  <xs:simpleContent>
   <xs:extension base="xs:string">
    <xs:attribute name="type" type="PropertyTypeEnum" use="required"/>
   </xs:extension>
  </xs:simpleContent>
 </xs:complexType>

 <xs:complexType name="objectProperty">
  <xs:sequence>
```

```
   <xs:element name="references">
    <xs:complexType>
     <xs:sequence>
      <xs:element name="ref" type="reference" minOccurs="1"
       maxOccurs="unbounded"/>
     </xs:sequence>
    </xs:complexType>
   </xs:element>
   <xs:element name="timestamp" type="xs:dateTime"/>
   <xs:element name="property" type="property" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="id" type="xs:anySimpleType" use="optional"/>
 </xs:complexType>

 <xs:complexType name="relationship">
  <xs:sequence>
   <xs:element name="source">
    <xs:complexType>
     <xs:sequence>
      <xs:element name="ref" minOccurs="1" maxOccurs="unbounded"
       type="reference"/>
     </xs:sequence>
    </xs:complexType>
   </xs:element>

   <xs:element name="target">
    <xs:complexType>
     <xs:sequence>
      <xs:element name="ref" minOccurs="1" maxOccurs="unbounded"
       type="reference"/>
     </xs:sequence>
    </xs:complexType>
   </xs:element>

   <xs:element name="timestamp" type="xs:dateTime"/>
  </xs:sequence>

  <xs:attribute name="type" type="RelationshipTypeEnum" use="required"/>
  <xs:attribute name="id" type="xs:anySimpleType" use="optional"/>
 </xs:complexType>

 <xs:complexType name="softwarePackageObject">
  <xs:sequence>
   <xs:element minOccurs="1" name="vendor" type="xs:string"/>
   <xs:element minOccurs="0" name="productgroup" type="xs:string"/>
   <xs:element minOccurs="1" name="product" type="xs:string"/>
   <xs:element minOccurs="0" name="version" type="xs:string"/>
   <xs:element minOccurs="0" name="update" type="xs:string"/>
```

```
   <xs:element minOccurs="0" name="edition" type="xs:string"/>
   <xs:element minOccurs="0" name="language" type="xs:string"/>
   <xs:element minOccurs="0" name="CPEname">
    <xs:complexType>
     <xs:simpleContent>
      <xs:extension base="xs:string">
       <xs:attribute name="cpeVersion" type="xs:string"/>
      </xs:extension>
     </xs:simpleContent>
    </xs:complexType>
   </xs:element>
  </xs:sequence>
  <xs:attribute name="id" type="xs:string" use="required"/>
 </xs:complexType>
 <xs:complexType name="digitalSignatureObject">
  <xs:sequence>
   <xs:element minOccurs="1" name="certificateIssuer" type="xs:string"/>
   <xs:element minOccurs="0" name="certificateSubject" type="xs:string"/>
   <xs:element minOccurs="1" name="certificateValidity" type="xs:boolean"/>
   <xs:element minOccurs="0" name="certificateRevocationTimestamp"
    type="xs:dateTime"/>
   <xs:element minOccurs="0" name="signingTimestamp">
    <xs:complexType>
     <xs:simpleContent>
      <xs:extension base="xs:dateTime">
       <xs:attribute name="valid" type="xs:boolean"/>
      </xs:extension>
     </xs:simpleContent>
    </xs:complexType>
   </xs:element>
  </xs:sequence>
  <xs:attribute name="id" type="xs:string" use="required"/>
  <xs:attribute name="type">
   <xs:simpleType>
    <xs:restriction base="xs:string">
     <xs:enumeration value="CatalogSigned"/>
     <xs:enumeration value="CodeSigned"/>
    </xs:restriction>
   </xs:simpleType>
  </xs:attribute>
 </xs:complexType>
 <xs:complexType name="taggantObject">
  <xs:sequence>
   <xs:element minOccurs="1" name="vendorID" type="xs:string"/>
   <xs:element minOccurs="0" name="taggantValidity" type="xs:boolean"/>
   <xs:element minOccurs="0" name="signingTimestamp">
    <xs:complexType>
     <xs:simpleContent>
```

```
      <xs:extension base="xs:dateTime">
       <xs:attribute name="valid" type="xs:boolean"/>
      </xs:extension>
     </xs:simpleContent>
    </xs:complexType>
   </xs:element>
  </xs:sequence>
  <xs:attribute name="id" type="xs:string" use="required"/>
 </xs:complexType>
</xs:schema>
```

## 11. Appendix III: An Example XML using Candidate Specifications

This section provides an example of an incident encoded in the IODEF. This do not necessarily represent the only way to encode a particular incident. This example reports an attack to a CSIRT and is extended from the example described in [RFC5070] . It uses identifiers whose dictionary follows CVE 1.0 schema, and it aembeds XML following CEE 0.6. Though these specifications are not listed in the IANA table at this moment, this section shows the example to demonstrate the usability of the draft.

```
<?xml version="1.0" encoding="UTF-8"?>
<IODEF-Document version="1.00" lang="en"
 xmlns="urn:ietf:params:xml:ns:iodef-1.0"
 xmlns:iodef="urn:ietf:params:xml:ns:iodef-1.0"
 xmlns:iodef-sci="urn:ietf:params:xml:ns:iodef-sci-1.0"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <Incident purpose="reporting">
    <IncidentID name="csirt.example.com">189493</IncidentID>
    <ReportTime>2001-09-13T23:19:24+00:00</ReportTime>
    <Description>Incident report in company xx</Description>
    <Assessment>
      <Impact completion="failed" type="admin"/>
    </Assessment>
    <Method>
      <Description>An identifier of a vulnerability is embedded
      </Description>
      <AdditionalData dtype="xml">
        <iodef-sci:Vulnerability
         SpecID="http://cve.mitre.org/cve/downloads/1.0"
         VulnerabilityID="CVE-2010-0483"/>
      </AdditionalData>
    </Method>
    <Contact role="creator" type="organization">
```

```
      <ContactName>Example.com CSIRT</ContactName>
      <RegistryHandle registry="arin">example-com</RegistryHandle>
      <Email>contact@csirt.example.com</Email>
    </Contact>
        <EventData>
      <Flow>
        <System category="source">
          <Node>
            <Address category="ipv4-addr">192.0.2.200</Address>
            <Counter type="event">57</Counter>
          </Node>
        </System>
        <System category="target">
          <Node>
            <Address category="ipv4-net">192.0.2.16/28</Address>
          </Node>
          <Service ip_protocol="6">
            <Port>80</Port>
          </Service>
          <AdditionalData dtype="xml">
            <iodef-sci:Platform
             SpecID="http://cpe.mitre.org/dictionary/2.0"
             PlatformID="cpe://microsoft:windows:xp:pro:sp2"/>
          </AdditionalData>
        </System>
      </Flow>
      <Expectation action="block-host" />
      <Expectation action="other"/>
      <!-- <RecordItem> has an excerpt from a log -->
      <Record>
        <RecordData>
          <DateTime>2001-09-13T18:11:21+02:00</DateTime>
          <Description>a Web-server event record</Description>
          <RecordItem dtype="xml">
            <iodef-sci:EventReport SpecID="http://cee.mitre.org">
              <iodef-sci:RawData dtype="xml">
                <cee:cee xmlns="http://cee.mitre.org/1.0/profile/"
                 xmlns:cee="http://cee.mitre.org/1.0/">
                  <cee:event>
                    <host>system.example.com</host>
                    <pname>auth</pname>
                    <time>2011-12-20T12:38:05.123456-05:00</time>
                    <appname>application</appname>
                    <pid>123</pid>
                    <sev>10</sev>
                    <action>login</action>
                    <domain>app</domain>
                    <object>account</object>
```

```
                    <service>web</service>
                    <status>success</status>
                  </cee:event>
                </cee:cee>
              </iodef-sci:RawData>
            </iodef-sci:EventReport>
          </RecordItem>
        </RecordData>
      </Record>
    </EventData>
    <History>
      <!-- Contact was previously made with the source network owner -->
      <HistoryItem action="contact-source-site">
        <DateTime>2001-09-14T08:19:01+00:00</DateTime>
        <Description>Notification sent to
         constituency-contact@192.0.2.200</Description>
      </HistoryItem>
    </History>
  </Incident>
</IODEF-Document>
```

## 12. References

### 12.1. 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.

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

[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.

## 12.2. 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)".

[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