Network Working Group
Internet-Draft
Obsoletes: 4765 (if approved)
Intended status: Standards Track
Expires: October 14, 2025

G. Lehmann
Telecom SudParis
April 12, 2025

# The Incident Detection Message Exchange Format version 2 (IDMEFv2)
## draft-lehmann-idmefv2-05

## Abstract

The Incident Detection Message Exchange Format version 2 (IDMEFv2) defines a date representation for security incidents detected on cyber and/or physical infrastructures.

The format is agnostic so it can be used in standalone or combined cyber (SIEM), physical (PSIM) and availability (NMS) monitoring systems. IDMEFv2 can also be used to represent man made or natural hazards threats.

IDMEFv2 improves situational awareness by facilitating correlation of multiple types of events using the same base format thus enabling efficient detection of complex and combined cyber and physical attacks and incidents.

If approved this draft will obsolete RFC4765.

## Status of this Memo



## Copyright Notice



[1] https://datatracker.ietf.org/drafts/current/
[2] https://trustee.ietf.org/license-info

# 1. Introduction

The Incident Detection Message Exchange Format (IDMEF) is intended to solve the problem of security monitoring compartmentalization by proposing a single format to represent any type of incident, whether cyber or physical, intentional or accidental, natural or man-made.

Indeed security is often associated to the Confidentiality-Integrity-Availability triad, performance and availability management systems are still run independently from security management systems.

Additionally, with the adoption and integration of Internet of Things (IoT) and Industrial Internet of Things (IIoT) devices, and the exponential emergence of smart systems (transport, cities, buildings, etc), an increasingly interconnected mesh of cyber-physical systems (CPS) has emerged. This expansion of the attack and incident surfaces blurs the once-clear functions of cybersecurity and physical security.

Finally, as IT infrastructure moves out of data centers it becomes more exposed to external threats, including natural and man-made hazards,

Incident detection systems have traditionally focused on detecting cyber incidents or physical incident or availability incidents. There is an increasing need nowadays to have a unified view and management of all those incidents and their interconnection.

To achieve this goal the Incident Detection Message Exchange Format offers a unique data representation for multiple types of events:

- Cyber-security events (e.g. authentication failure/success, virus/malware detection, bruteforce/scan detection, etc.)
- Physical security events (e.g. intrusion detection, object detection, face or activity recognition, fire/smoke/noise/rain detection, etc.)
- Availability/observability/performance events (e.g. system failure, service malfunction, performance decrease, etc.)
- Natural and man made hazards events (e.g. wildfires, avalanches, droughts, earthquakes, pollution, fire, explosion, etc.)

## 1.1. IDMEFv2 deployment architecture

IDMEFv2 can be used to exchange incident detection information between specialized managers (SIEM, PSIM, NMS) and a universal "Cyber & Physical SIEM" (CPSIEM) or directly from specialized analyzers and a CPSIEM.

```
                    +-----------------------------+    +---------+
                    |    "Universal" CPSIEM       |<---| Hazards |
                    +-----------------------------+    +---------+
                                  |         |         |
                             +-------+   +-----+   +------+
             Managers        | PSIM |   | NMS |   | SIEM |
                             +-------+   +-----+   +------+
                                  |         |         |
                           +--------+ +----------+ +-----+
        Analyzers/Sensors  |Physical| |Monitoring| |Cyber|
                           +--------+ +----------+ +-----+
                                 |          |          |
                          +-------------------------------+
                          |    Cyber & Physical System    |
                          +-------------------------------+
```

Figure 1: IDMEF Use Architecture

Thanks to its universality IDMEFv2 improves situational awareness by enabling correlation of multiple types of events using the same base format.

This document defines a model serialization methodes for the purpose of describing and sharing these events.

## 1.2. IDMEFv1 (Intrusion Detection Message Exchange Format) - RFC 4765 - Legacy

IDMEFv2 (Incident Detection Message Exchange Format) is based on IDMEFv1 (Intrusion Detection Message Exchange Format) concepts. But IDMEFv1 was cyber intrusion focused as IDMFv2 perimeter is much larger. Thus retro-compatibility although partly possible has not been a priority.

## 1.3. Relationship between IDMEFv2 and other event/incident formats

IDMEFv2 focuses essentialy on high level event/incident correlation and detection.There are many standard and proprietary formats on the incident detection market and in particular on the cybersecurity market. IDMEFv2 is complementary to most of these formats.

IDMEFv1 (Intrusion Detection Message Exchange Format - RFC 4765) : IDMEFv2 (Incident Detection) replaces and obsoletes IDMEFv1 (Intrusion Detection) by covering a wider spectrum.

IODEFv2 (Incident Object Definition Exchange Format - RFC 5070) : IDMEFv2 helps detect incident. When an incident is detected it will be analysed and eventually fully described and shared with other security teams through IODEFv2. IODEFv2. IDMEF is used upstream IODEFv2. IDMEfv2 ALerts can be “attached” to IODEFv2 object to provide technical details about incidents.

Syslog (System Logging) : Syslog is a loosy format with no formal structuration. Syslog can be used by sensors to send information to analyzers. Out of those multi-format syslogs the analyzer might detect an incident or an event of interest. The analyzer will then use IDMEFv2 to notify the manager which might correlate this information with other datas to confirm the incident.

SNMP (Simple Network Management Protocol) : SNMP polls information from devices which is then compared to thresholds to detect incident. IDMEFv2 can be used when incident is detected downstream of SNMP to communicate the incident to the manager. IDMEFv2 can have a similar role as SNMP Traps.

STIX (Structured Threat Information Expression) : is a language and serialization format used to exchange cyber threat intelligence (CTI). IDMEFv2 can help detect incidents which might lead to the creation and sharing of STIX information. Cyber analyzer can also rely on STIX information to detect incidents that will be notified in IDMEFv2 format.

SIEM proprietory formats (CEF, LEEF, ECS, CIM, ...) : By covering cyber, physical and monitoring incidents type, IDMEFv2 offers a wider spectrum than those formats. Gateways between IDMEFv2 and those formats can be developped to connect legacy cyber detection systems to an IDMEFv2 architecture.

# 2. Terminology

## 2.1. Keywords

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

## 2.2. Normative sections

Implementations of IDMEFv2 are REQUIRED to fully implement:

- The data types defined in Section 3
- The data model defined in Section 5
- The JavaScript Object Notation (JSON) serialization method Section 5.9.

## 2.3. Concepts related to event processing

### 2.3.1. Event

An event is something that triggered a notice. Any incident starts off as an event or a combination of events, but not all events result in an incident. An event need not be an indication of wrongdoing. E.g. someone successfully logging in or entering a building is an event.

### 2.3.2. Incident

An incident is an event that compromises or has a significant probability of compromising at least one of the organization's security criteria such as Confidentiality, Integrity or Availability. An incident may affect a production tool, personnel, etc. It may be logical, physical or organizational in nature. Last but not least, an incident may be caused on purpose or by accident.

### 2.3.3. Alert

An alert is a notification/message that a particular event/incident (or series of events/incidents) has occurred.

### 2.3.4. Manager

The manager is the central console toward which all analyzers send their alerts. The manager collects, correlates, stores and display the alerts to the operators.

Example : - A SIEM (Security Information & Event Management) or a Log Manager) - A PSIM (Physical Security Information Management) - A NMS (Network Management System) - A CPSIEM (Cyber & Physical Security Information Management System)

### 2.3.5. Operator

The level 1 operator is in charge of receiving manager notifications and identify or confirm when an event should be considered as an incident. The operator must also decide if there is a know resolution for this incident or if it needs a deeper analysys.

### 2.3.6. Analyst

The analyst will be contacted by the operator to analyze complex incidents that can't be easily resolved. The investigation starts with the IDMEFv2 information but the analyst might need more information like raw logs for a deeper forensics.

### 2.3.7. Attack

An attack is an attempt to destroy, expose, alter, disable, steal or gain unauthorized access to or make unauthorized use of a cyber or physical asset. An attack is one or many kinds of incidents.

### 2.3.8. Correlation

Correlation is the identification of relationships between two or more events.

### 2.3.9. Aggregation

Aggregation is the consolidation of similar events into a single event.

# 3. The IDMEF Data Types

Each object inside the IDMEF data model has an associated data type. This type may be used to validate the content of incoming IDMEF messages.

## 3.1. Classes

The classes are meant to group related attributes together. Some of the classes may be instanciated multiple times (e.g. Source, Target, etc.) while others may only appear once in an IDMEF message (e.g. Analyzer).

## 3.2. Numbers

### 3.2.1. Integers

Integers inside the IDMEF data model are expressed using the following ABNF [RFC5234] grammar:

```
integer         =  *1minus int
int             =  zero / ( digit1-9 *DIGIT )
minus           =  %x2D                        ; -
zero            =  %0x30                       ; 0
digit1-9        =  %x31-39                     ; 1-9
```

E.g. 123.

Such values are indicated with the "INT" type annotation in the model.

### 3.2.2. Floating-point values

Floating-point values inside the IDMEF data model are expressed using the following ABNF grammar:

```
float           =  integer *1frac
frac            =  decimal-point 1*DIGIT
decimal-point   =  %x2E                         ; .
```

This grammar reuses some of the production rules listed in Section 3.2.1.

E.g. 12.34.

Such values are indicated with the "FLOAT" type annotation in the model.

## 3.3. Strings

Strings are series of characters from the [UNICODE] standard and are used to represent a text.

For readability, this document uses quotes (") to delimit strings, but please note that these quotes are not syntactically part of the actual strings.

E.g. "Hello world".

Some of the strings used in the IDMEFv2 data model follow a stricter syntax. These are included below for completeness.

Such values are indicated with the "STRING" type annotation in the model.

### 3.3.1. Enumerations

Enumerations are special strings used when valid values for an IDMEF attribute are restricted to those present in a predefined list.

Such values are indicated with the "ENUM" type annotation in the model.

### 3.3.2. Timestamps

Timestamps are used to indicate a specific moment in time. The timestamps used in the IDMEF data model follow the syntax defined by the "date-time" production rule of the grammar in [RFC3339] ch 5.6.

E.g. "1985-04-12T23:59:59.52Z" represents a moment just before April 5th, 1985 in Coordinated Universal Time (UTC).

Such values are indicated with the "TIMESTAMP" type annotation in the model.

### 3.3.3. Geographical Locations

Some attributes inside the IDMEF data model may refer to geographical locations using a set of coordinates. The reference system for all geographical coordinates is a geographic coordinate reference system, using the World Geodetic System 1984 [WGS84]. The reference system used is the same as for the Global Positioning System (GPS).

The format for such values can be either "latitude,longitude" or "latitude,longitude,altitude". Each of these coordinates is represented as a floating-point value. The latitude and longitude are expressed in degrees while the altitude is expressed in meters.

E.g. "48.8584,2.2945,276.13" matches the (3-dimensional) geographical location for the top floor or the Eiffel Tower located in Paris, France, while "48.8584,2.2945" matches the same location in two dimensions (with the altitude removed).

Such values are indicated with the "GEOLOC" type annotation in the model.

### 3.3.4. UNECE Location Codes (UN/LOCODE)

Some attributes inside the IDMEF data model may refer to geographical locations using Locations Codes. These codes can be assimilated to an enumeration, where the list of possible values is defined in the United Nations Economic Commission for Europe (UNECE) Codes for Trade [UN-LOCODE].

E.g. "FR PAR" is the Location Code for the city of Paris, France.

Such values are indicated with the "UNLOCODE" type annotation in the model.

### 3.3.5. Uniform Resource Identifiers (URIs)

The IDMEF data model uses Uniform Resource Identifiers (URIs), as defined in [RFC3986], when referring to external resources. Unless otherwise specified, either a Uniform Resource Location (URL) or a Uniform Resource Name (URN) may be used where a URI is expected.

E.g. both "https://example.com/resource" and "urn:myapp:resource" are valid Uniform Resource Identifiers.

Such values are indicated with the "URI" type annotation in the model.

### 3.3.6. IP Addresses

IP addresses inside the IDMEF data model are expressed as strings using the traditionnal dotted-decimal notation for IPv4 addresses (defined by the "dotnum" production rule in the grammar in [RFC5321]), while IPv6 addresses are expressed using the text representation defined in [RFC4291] ch 2.2.

E.g. "192.0.2.1" represents a valid IPv4 address, while "::1/128" represents a valid IPv6 address.

It is RECOMMENDED that implementations follow the recommendations for IPv6 text representation stated in [RFC5952].

Such values are indicated with the "IP" type annotation in the model.

### 3.3.7. E-mail addresses

E-mail addresses inside the IDMEF data model are expressed as strings using the address specification syntax defined in [RFC5322] ch 3.4.1.

E.g. "root@example.com".

Such values are indicated with the "EMAIL" type annotation in the model.

### 3.3.8. Attachment names

Attachments inside the IDMEF data model are identified using a unique name, composed of a string whose character set is limited to the ASCII letters (A-Z a-z) and digits (0-9).

E.g. "state" is a valid name for an attachment.

The constraint on name unicity is enforced per class. That is, but it is not possible for two attachments to share the same name inside the same alert.

Such values are indicated with the "ID" type annotation in the model.

### 3.3.9. Media types

Media types are used in the IDMEF data model to describe an attachment's content. The syntax for such values is defined in [RFC2046].

IANA keeps a list of all currently registered media types in the Media Types registry .

E.g. "application/xml" or "text/plain; charset=utf-8".

Such values are indicated with the "MEDIATYPE" type annotation in the model.

### 3.3.10. Universally Unique IDentifiers (UUIDs)

Universally Unique Identifiers (UUIDs) are used to uniquely identify IDMEF messages. It is also possible for an IDMEF message to reference other IDMEF messages using their UUIDs. The syntax for UUIDs is defined in [RFC4122].

To limit the risk of UUID collisions, implementors SHOULD NOT generate version 4 UUIDs (randomly or pseudo-randomly generated UUIDs).

E.g. "ba2e4ef4-8719-42bb-a712-d6e8871c5c5a".

UUIDs are case-insensitive when used in comparisons.

Such values are indicated with the "UUID" type annotation in the model.

### 3.3.11. Protocol Names

Such values are indicated with the "PROTOCOL" type annotation in the model.

### 3.3.12. IDMEF Paths

This document defines a way to represent the path to every possible attribute inside an IDMEF message. For conciseness, the top-level "Alert" class is omitted from the path.

This representation can be used in contexts where the path to an IDMEF attribute is expected. An example of such usage can be seen in the definition of the "AggrCondition" attribute inside the Alert class (Section 5.2).

The syntax for these IDMEF paths is expressed in the following ABNF grammar:

```
class-name      =  "Analyzer" / "Sensor" / "Source" / "Target" /
                   "Vector" / "Attachment"
attribute-name  =  1*ALPHA
class-reference =  class-name "."
num             =  *1"-" 1*DIGIT
list-index      =  "(" num ")"
path            =  *1class-reference attribute-name *1list-index
```

Valid attribute names are limited to those defined for the specified class-reference (or in the top-level "Alert" class if class-reference is omitted).

For example, the following path refers to the "CeaseTime" attribute of the top-level "Alert" class: "CeaseTime".

Likewise, the following path refers to the "Name" attribute of the "Analyzer" class: "Analyzer.Name".

For attributes defined as lists (see Section 3.4), the path may include the (0-based) index for an entry inside the list. The index defaults to 0 if omitted. This means that several (valid) representations may be used to reference the same IDMEF attribute when list attributes are involved.

For example, both of the following paths refer to the IP address of the first source associated with an IDMEF message:

```
Source.IP
Source(0).IP
```

Compatible implementations MUST reject paths that reference an unknown class, an unknown attribute, or use a list-index for an IDMEF field which is not defined as a list.

A compatible implementation MUST also normalize paths before comparing them (e.g. by stripping the text "(0)" from paths referring to list attributes).

#### 3.3.13. Hashes

Hashes are sometimes used inside the data model to protect the integrity (and optionally, authenticity) of attachments.

The syntax for these values is "function:hash_result", where "function" refers to one of the hashing function names listed in and "hash_result" contains the hexadecimal notation for the hash result obtained by calling the specified hash function on the input value.

In the context of IDMEF, either a keyless or keyed hash function may be used to process the raw input value.

E.g. "sha256:a02735ed8b10ad432d557bd4849c0dac3b23d64706e0618716d6df2def338374"

Hashes are case-insensitive when used in comparisons.

Such values are indicated with the "HASH" type annotation in the model.

### 3.4. Lists

Some attributes of the IDMEF data model accept ordered lists of values.

Such ordered lists are indicated with the "X[]" type annotation in the model. where "X" refers to one of the data types defined in Section 3. For example, "ENUM[]" refers to an ordered list of enumeration values.

# 4. The IDMEF extension

In order to support the dynamic nature of security operations and to adapt to specific needs, the IDMEFv2 data model will need to continue to evolve. This section discusses how new data elements can be incorporated into the IDMEFv2. There is support to add additional enumerated values and new attributes.

These extension mechanisms are designed so that adding new data elements is possible without requiring modifications to this document. Extensions can be implemented publicly or privately. With proven value, well-documented extensions can be incorporated into future versions of the specification.

## 4.1. Extending the Enumerated Values of Attributes

Additional enumerated values can be added to select attributes either through the use of specially marked attributes with the "ext-" prefix or through a set of corresponding IANA registries. The former approach allows for the extension to remain private. The latter approach is public.

### 4.1.1. Private Extension of Enumerated Values

The data model supports adding new enumerated values to an attribute without public registration. For each attribute that supports this extension technique, there is a corresponding attribute in the same element whose name is identical but with a prefix of "ext-". This special attribute is referred to as the extension attribute. The attribute being extended is referred to as an extensible attribute. For example, an extensible attribute named "foo" will have a corresponding extension attribute named "ext-foo". An element may have many extensible attributes.

In addition to a corresponding extension attribute, each extensible attribute has "ext-value" as one its possible enumerated values. Selection of this particular value in an extensible attribute signals that the extension attribute contains data. Otherwise, this "ext-value" value has no meaning.

In order to add a new enumerated value to an extensible attribute, the value of this attribute MUST be set to "ext-value", and the new desired value MUST be set in the corresponding extension attribute. For example, extending the Category attribute of the Analyzer class would look as follows:

```
Analyzer: {
    ...
    "Category":["ext-value"],
    "ext-Category": "my-new-analyzer-category",
    ....
}
```

A given extension attribute MUST NOT be set unless the corresponding extensible attribute has been set to "ext-value".

### 4.1.2. Public Extension of Enumerated Values

The data model also supports publicly extending select enumerated attributes. A new entry can be added by registering a new entry in the appropriate IANA registry. Section (Table 18) provides a mapping between the extensible attributes and their corresponding registry.

## 4.2. Private Extension of Attributes

Use of new attributes is possible through the use of the attachment class. New attributes and their corresponding values should be stored in the Content attribute of an Attachment and the ContentEnconding must be set to JSON. For example creating a new attribute to store the email of the operator (in charge of solving the incident) will look as follows:

```
"Attachment" : [
    {
    "Name": "Operator",
    "ContentEnconding": "JSON",
    "Content": "{\"OperatorMail\":\"John.Does@acme.com\"}",
    }
[
```

# 5. The IDMEF Data Model

In this section, the individual components of the IDMEF data model will be discussed in detail. For each class, the semantics will be described.

## 5.1. Overview

```
                         +--------+     +-----+
                         |Analyzer|---> |Alert|
                         +--------+     +-----+
                                           ||
                                  +------+
                                  |Sensor|
                                  +------+
                   \--------\       /\
                    ) Vector )     /  \
                   /--------/     /    \
            +------+              +------+
            |Source|              |Target|
            +------+              +------+
```

Figure 2: IDMEFv2 Overview Classes

An IDMEF message is composed of an instance of the Alert class (Section 5.2) representing the overall properties of the message. It also contains exactly one instance of the Analyzer class (Section 5.3) and zero or more instances of the Sensor class (Section 5.4).

The message may also describe various aspects of an event using the Source (Section 5.5), Target (Section 5.6) and Vector (Section 5.7) classes.

Last but not least, it may also include zero or more instances of the Attachment class (Section 5.8), e.g. captured files or network packets related to the event for example.

The relationship between the main Alert class and other classes of the data model is shown in Figure 3 (attributes are omitted for clarity).

```
+-------+            +-------------
| Alert |<>----------|  Analyzer  |
+-------+            +------------+
|       |       0..* +------------+
|       |<>----------|   Sensor   |
|       |            +------------+
|       |       0..* +------------+
|       |<>----------|   Source   |
|       |            +------------+
|       |       0..* +------------+
|       |<>----------|   Target   |
|       |            +------------+
|       |       0..* +------------+
|       |<>----------|   Vector   |
|       |            +------------+
|       |       0..* +------------+
|       |<>----------| Attachment |
+-------+            +------------+
```

Figure 3: IDMEFv2 Classes

It is important to note that the data model does not specify how an alert should be categorized or identified. For example, an attacker scanning a network for machines listening on a specific port may be identified by one analyzer as a single attack against multiple targets, while another analyzer may identify it as multiple attacks from a single source. However, once an analyzer has determined the type of alert it plans on sending, the data model dictates how that alert should be formatted.

## 5.2. The Alert Class

The Alert class contains high level information about the event that triggered the alert.

```
+-----------------------------+
|              Alert          |
+-----------------------------+
| STRING      Version         |
| UUID        ID              |
| STRING      OrganisationName |
| STRING      OrganisationId  |
| STRING      EntityName      |
| STRING      EntityId        |
| ENUM[]      Category        |
| STRING      ext-Category    |
| ENUM        Cause           |
| STRING      Description     |
| ENUM        Status          |
| ENUM        Priority        |
| FLOAT       Confidence      |
| STRING      Note            |
| TIMESTAMP   CreateTime      |
| TIMESTAMP   StartTime       |
| TIMESTAMP   EndTime         |
| STRING[]    AltNames        |
| STRING[]    AltCategory     |
| URI[]       Ref             |
| UUID[]      CorrelID        |
| CONDITION[] AggrCondition   |
| UUID[]      PredID          |
| UUID[]      RelID           |
+-----------------------------+
```

Figure 4: The Alert class

The aggregate classes that make up Alert are:

Analyzer
: Exactly one. An instance of the Analyzer class (Section 5.3) that describes the tool/device responsible for the analysis that resulted in the alert being created and sent.

Sensor
: Zero or more. Instances of the Sensor class (Section 5.4) used to describe the sensor(s) that captured the information used during the analysis.
Depending on the tools/devices used to detect incidents, an Analyzer may rely on the output from a single sensor or from multiple sensors to generate alerts. In addition, the Analyzer and Sensor may actually be part of the same physical device and may share some of their attributes (e.g. IP, Hostname, Model, etc.).

Source
: Zero or more. Instances of the Source class (Section 5.5) used to describe the source(s) of the incident (e.g. attackers, faulty device, etc.).

Target
: Zero or more. Instances of the Target class (Section 5.6) used to describe the target(s) of the incident, i.e. the impacted devices/users/services/locations.

Vector
: Zero or more. Instances of the Vector class (Section 5.7) used to describe the means which were employed by the sources to disrupt the targets.
E.g. to describe a drone crashing into a building and resulting in service loss or a malware email delivered opened in a mailbox and resulting in service loss.

Attachment
: Zero or more. Instances of the Attachment class (Section 5.8) used to describe the electronic artifacts captured in relation with the event.
The intent of the Attachment class is to keep track of the electronic files left as a trail during the event. This may include things like on-disk files (e.g. malware samples), network packet captures, videos or still images from a camera feed, voice recording, etc.

The Alert class has the following attributes:

Version
: Mandatory. The version of the IDMEF format in use by this alert.
During the drafts tuning period the version is equal to the draft version.
Therefore it is "2.D.V0X" for Draft V0X.

ID
: Mandatory. Unique identifier for the alert.

OrganisationName
: Optional. Corporate/Main Office Organisation Name
Useful if alerts are sent to a multi-organisation incident detection system.
Example: ACME Corporation

OrganisationId
: Optional. Corporate/Main Office Organisation ID. Where possible official organisation ID manage by national authority.
Useful if alerts are sent to a multi-organisation incident detection system.
This ID has to be chosen depending on the overall detection perimeter and the nature of the monitored organisation (Private/Public, Commercial, International, etc.)
Examples: OrganisationId in France could be SIREN, in England could be CR, Germany could be Handelsregisternummer, Spain could be CIF, Italia could be Partita IVA, USA could be EIN, etc. Commercial OrganisationId in Europe could be V.A.T ID

EntityName
: Optional. Entity Name, monitored by the organisation, where the incident occurred.
Could be a town, region or country name or an internal name. Could also be the name of a client for a MSSP centralizing it's client incidents in a single system.
Do not repeat the organisation name in the EntityName
Example :

- ACME HeadQuaters is located in Paris France and has a local office in India
- If the incident occurred in the local office : "OrganisationName" : " ACME" , "EntityName" : "India"
- If the incident occurred in the headquaters : "OrganisationName" : "ACME", "EntityName" : "Headquaters" (or "Paris")

EntityId
: Optional. Entity ID, monitored by the organisation, where the incident occurred.
Useful if organisation and entity are not directly linked, like a client and a MSSP.

Category
: Optional. The incident's category & subcategory as listed in [ENISA-RIST] using the format "category.subcategory" (e.g. "Attempt.Exploit").

| Rank | Keyword | Description |
|---|---|---|
| 0 | Abusive.Spam | Or 'Unsolicited Bulk Email', this means that the recipient has not granted verifiable permission for the message to be sent and that the message |

| Rank | Keyword | Description |
|---|---|---|
| | | is sent as part of a larger collection of messages, all having a functionally comparable content. This IOC refers to resources, which make up a SPAM infrastructure, be it a harvesters like address verification, URLs in spam e-mails etc. |
| 1 | Abusive.Harassment | Discretization or discrimination of somebody, e.g. cyber stalking, racism or threats against one or more individuals. |
| 2 | Abusive.Illicit | Child Sexual Exploitation (CSE), Sexual content, glorification of violence, etc. |
| 3 | Malicious.System | System infected with malware, e.g. PC, smartphone or server infected with a rootkit. Most often this refers to a connection to a sinkholed C2 server |
| 4 | Malicious.Botnet | Command-and-control server contacted by malware on infected systems. |
| 5 | Malicious.Distribution | URI used for malware distribution, e.g. a download URL included in fake invoice malware spam or exploit-kits (on websites). |
| 6 | Malicious.Configuration | URI hosting a malware configuration file, e.g. web-injects for a banking trojan. |
| 7 | Recon.Scanning | Attacks that send requests to a system to discover weaknesses. This also includes testing processes to gather information on hosts, services and accounts. Examples: fingerd, DNS querying, ICMP, SMTP (EXPN, RCPT, ...), port scanning. |
| 8 | Recon.Sniffing | Observing and recording of network traffic (wiretapping). |
| 9 | Recon.SocialEngineering | Gathering information from a human being in a non-technical way (e.g. lies, tricks, bribes, or threats). |
| 10 | Attempt.Exploit | An attempt to compromise a system or to disrupt any service by exploiting vulnerabilities with a standardised identifier such as CVE name (e.g. buffer |

| Rank | Keyword | Description |
|---|---|---|
| | | overflow, backdoor, cross site scripting, etc.) |
| 11 | Attempt.Login | Multiple login attempts (Guessing / cracking of passwords, brute force). This IOC refers to a resource, which has been observed to perform brute-force attacks over a given application protocol. |
| 12 | Attempt.NewSignature | An attack using an unknown exploit. |
| 13 | Intrusion.AdminCompromise | Compromise of a system where the attacker gained administrative privileges. |
| 14 | Intrusion.UserCompromise | Compromise of a system using an unprivileged (user/service) account. |
| 15 | Intrusion.AppCompromise | Compromise of an application by exploiting (un-)known software vulnerabilities, e.g. SQL injection. |
| 16 | Intrusion.SysCompromise | Compromise of a system, e.g. unauthorised logins or commands. This includes compromising attempts on honeypot systems. |
| 17 | Intrusion.Burglary | Physical intrusion, e.g. into corporate building or data-centre. |
| 18 | Availability.DoS | Denial of Service attack, e.g. sending specially crafted requests to a web application which causes the application to crash or slow down. |
| 19 | Availability.DDoS | Distributed Denial of Service attack, e.g. SYN-Flood or UDP-based reflection/amplification attacks. |
| 20 | Availability.Misconf | Software misconfiguration resulting in service availability issues, e.g. DNS server with outdated DNSSEC Root Zone KSK. |
| 21 | Availability.Theft | Physical theft, e.g. stolen laptop computer, stolen USB key, stolen paper document, etc. |
| 22 | Availability.Sabotage | Physical sabotage, e.g cutting wires or malicious arson. |
| 23 | Availability.Outage | Outage caused e.g. by air condition failure or natural disaster. |
| 24 | Availability.Failure | Failure, malfunction (e.g. : bug, wear, faults, etc.) |

| Rank | Keyword | Description |
|---|---|---|
| 25 | Information. UnauthorizedAccess | Unauthorised access to information, e.g. by abusing stolen login credentials for a system or application, intercepting traffic or gaining access to physical documents. |
| 26 | Information. UnauthorizedModification | Unauthorised modification of information, e.g. by an attacker abusing stolen login credentials for a system or application or a ransomware encrypting data. Also includes defacements. |
| 27 | Information.DataLoss | Loss of data, e.g. caused by harddisk failure or physical theft. |
| 28 | Information.DataLeak | Leaked confidential information like credentials or personal data. |
| 29 | Fraud.UnauthorizedUsage | Using resources for unauthorised purposes including profit-making ventures, e.g. the use of e-mail to participate in illegal profit chain letters or pyramid schemes. |
| 30 | Fraud.Copyright | Offering or Installing copies of unlicensed commercial software or other copyright protected materials (Warez). |
| 31 | Fraud.Masquerade | Type of attack in which one entity illegitimately impersonates the identity of another in order to benefit from it. |
| 32 | Fraud.Phishing | Masquerading as another entity in order to persuade the user to reveal private credentials. This IOC most often refers to a URL, which is used to phish user credentials. |
| 33 | Vulnerable.Crypto | Publicly accessible services offering weak crypto, e.g. web servers susceptible to POODLE/FREAK attacks. |
| 34 | Vulnerable.DDoS | Publicly accessible services that can be abused for conducting DDoS reflection/amplification attacks, e.g. DNS open-resolvers or NTP servers with monlist enabled. |
| 35 | Vulnerable.Surface | Potentially unwanted publicly accessible services, e.g. Telnet, RDP or VNC. |

| Rank | Keyword | Description |
|---|---|---|
| 36 | Vulnerable.Disclosure | Publicly accessible services potentially disclosing sensitive information, e.g. SNMP or Redis. |
| 37 | Vulnerable.System | A system which is vulnerable to certain attacks. Example: misconfigured client proxy settings (example: WPAD), outdated operating system version, XSS vulnerabilities, etc. |
| 38 | Geophysical.Earthquake | A hazard originating from solid earth. This term is used interchangeably with the term geological hazard. |
| 39 | Geophysical.MassMovement | A hazard originating from solid earth. This term is used interchangeably with the term geological hazard. |
| 40 | Geophysical.Volcanic | A hazard originating from solid earth. This term is used interchangeably with the term geological hazard. |
| 41 | Meteorological. Temperature | A hazard caused by short-lived, micro- to meso-scale extreme weather and atmospheric conditions that last from minutes to days. |
| 42 | Meteorological.Fog | A hazard caused by short-lived, micro- to meso-scale extreme weather and atmospheric conditions that last from minutes to days. |
| 43 | Meteorological.Storm | A hazard caused by short-lived, micro- to meso-scale extreme weather and atmospheric conditions that last from minutes to days. |
| 44 | Hydrological.Flood | A hazard caused by the occurrence, movement, and distribution of surface and subsurface freshwater and saltwater. |
| 45 | Hydrological.Landslide | A hazard caused by the occurrence, movement, and distribution of surface and subsurface freshwater and saltwater. |
| 46 | Hydrological.Wave | A hazard caused by the occurrence, movement, and distribution of surface and subsurface freshwater and saltwater. |

| Rank | Keyword | Description |
|---|---|---|
| 47 | Climatological.Drought | A hazard caused by long-lived, meso- to macro-scale atmospheric processes ranging from intra-seasonal to multi-decadal climate variability. |
| 48 | Climatological. LakeOutburst | A hazard caused by long-lived, meso- to macro-scale atmospheric processes ranging from intra-seasonal to multi-decadal climate variability. |
| 49 | Climatological.Wildfire | A hazard caused by long-lived, meso- to macro-scale atmospheric processes ranging from intra-seasonal to multi-decadal climate variability. |
| 50 | Biological.Epidemic | A hazard caused by the exposure to living organisms and their toxic substances (e.g. venom, mold) or vector-borne diseases that they may carry. Examples are venomous wildlife and insects, poisonous plants, and mosquitoes carrying disease-causing agents such as parasites, bacteria, or viruses (e.g. malaria). |
| 51 | Biological.Insect | A hazard caused by the exposure to living organisms and their toxic substances (e.g. venom, mold) or vector-borne diseases that they may carry. Examples are venomous wildlife and insects, poisonous plants, and mosquitoes carrying disease-causing agents such as parasites, bacteria, or viruses (e.g. malaria). |
| 52 | Biological.Animal | A hazard caused by the exposure to living organisms and their toxic substances (e.g. venom, mold) or vector-borne diseases that they may carry. Examples are venomous wildlife and insects, poisonous plants, and mosquitoes carrying disease-causing agents such as parasites, bacteria, or viruses (e.g. malaria). |
| 53 | Extraterrestrial.Impact | A hazard caused by asteroids, meteoroids, and comets as they pass near-earth, enter the Earth’s atmosphere, and/or strike the Earth, and by |

| Rank | Keyword | Description |
|---|---|---|
| | | changes in interplanetary conditions that effect the Earth’s magnetosphere, ionosphere, and thermosphere. |
| 54 | Extraterrestrial. SpaceWeather | A hazard caused by asteroids, meteoroids, and comets as they pass near-earth, enter the Earth’s atmosphere, and/or strike the Earth, and by changes in interplanetary conditions that effect the Earth’s magnetosphere, ionosphere, and thermosphere. |
| 55 | Other.Uncategorised | All incidents which don't fit in one of the given categories should be put into this class or the incident is not categorised. |
| 56 | Other.Undetermined | The categorisation of the incident is unknown/ undetermined. |
| 57 | Test.Test | Meant for testing. |
| 58 | ext-value | A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. (see Section 4.1.1) |

Table 1: Incident taxonomy

ext-Category
: Optional. A means by which to extend the Category attribute. (see Section 4.1.1)

Cause
: Optional. Alert cause. The cause can be modified by any analyser on the way of the alert and later by the operator and/or the analyst if new investigation reveals and confirms a different cause of the event.

| Rank | Keyword | Description |
|---|---|---|
| 0 | Normal | The event is related to an expected phenomenon or to a phenomenon that does not qualify as out of the ordinary. |
| 1 | Error | The event is related to a human error. |
| 2 | Malicious | The event is related to malicious code or malicious actions. |
| 3 | Malfunction | The event is related to a device or service malfunction. |
| 4 | Hazard | The event is related to a hazard phenomenon. |
| 5 | Unknown | The cause of the event is unknown. |

Table 2: Incident causes

| | |
|---|---|
| Description | Optional. Short free text human-readable description of the event. The description can add detail to the alert classification for easiest/faster comprehension by the operator. Example : * Cryptoware WannaCry blocked on pegasus server * Unknown person entering through east doorway |
| Status | Optional. Event state in the overall event lifecycle. |

| **Rank** | **Keyword** | **Description** |
|---|---|---|
| 0 | Event | The event is still considered as an harmless event and should not be treated. |
| 1 | Incident | The event is considered as an incident and should be taken care of. |

Table 3: Incident statuses

| | |
|---|---|
| Priority | Optional. Priority of the alert. Priority is defined by conbining impact and urgency. It indicates how fast the incident should be taken care of. Impact defines the enormity of the situation and mostly deals with “How Many” or “how much” question. It can be in terms of people, finances, systems, etc. How many people and/or systems impacted, how badly are they impacted (is there potential physical impact ?) , how much financial loss, severity of legal liabilities,... Impact could be considered equivalent to "Severity". Urgency is associated with time. The time it takes to have the perceived Impact. For example, a high impact incident may have low urgency if the impact will not affect the business until the end of the financial year. |

| **Rank** | **Keyword** | **Description** |
|---|---|---|
| 0 | Unknown | Priority unknow |
| 1 | Info | No priority, the alert is informational |
| 2 | Low | Low priority |
| 3 | Medium | Medium priority |
| 4 | High | High priority |

Table 4: Incident severities

| | |
|---|---|
| Confidence | Optional. A floating-point value between 0 and 1 indicating the analyzer's confidence in its own reliability of this particular detection, where 0 means that the detection is surely incorrect while 1 means there is no doubt about the detection made. |
| Note | Optional. Free text human-readable additional note, possibly a longer description of the incident if is not already obvious.<br>The Note attribute can be used to store any additional information. It can be additional information about the event and/or about the incident resolution, although the incident resolution information should in principle be stored elsewhere (with a link with the external tool in AltNames) |
| CreateTime | Mandatory. Timestamp indicating when the message was created. |
| StartTime | Optional. Timestamp indicating the deduced start of the event.<br>StartTime can be later than CreateTime in case or Alerts created from forecast information (e.g. Snow Storm in two days staring at 10h00) |
| EndTime | Optional. Timestamp indicating the deduced end of the event. |
| AltNames | Optional. Alternative identifiers; strings which help pair the event to internal systems' information (for example ticket IDs inside a request tracking systems). |

| | |
|---|---|
| AltCategory | Optional. Alternate categories from a reference other than [ENISA-RIST] (e.g. MISP, MITRE ATT@CK or another proprietary/internal reference). |
| Ref | Optional. References to sources of information related to the alert and/or vulnerability, and specific to this alert.<br>This MAY be a URL to additional info, or a URN in a registered or unregistered ad-hoc namespace bearing reasonable information value and uniqueness, such as "urn:cve:CVE-2013-2266". |
| CorrelID | Optional. Identifiers for the messages which were used as information sources to create this message, in case the message has been created based on correlation/analysis/deduction from other messages. |
| AggrCondition | Optional. A list of IDMEF fields used to aggregate events. The values for these fields will be the same in all aggregated events.<br>This attribute should mostly be set by intermediary nodes, which detect duplicates, or aggregate events, spanning multiple detection windows, into a longer one.<br>The "StartTime" and "EndTime" attributes are used in conjunction with this attribute to describe the aggregation window. |
| PredID | Optional. A list containing the identifiers of previous messages which are obsoleted by this message.<br>The obsoleted alerts SHOULD NOT be used anymore. This field can be used to "update" an alert. |
| RelID | Optional. A list containing the identifiers of other messages related to this message. |

## 5.3. The Analyzer Class

The Analyzer class describes the module that has analyzed the data captured by the sensors, identified an event of interest and decided to create an alert.

```
+-----------------------+
|       Analyzer        |
+-----------------------+
| IP        IP          |
| STRING    Name        |
| STRING    Hostname    |
| STRING    Model       |
| ENUM[]    Type        |
| ENUM[]    Category    |
| STRING    ext-Category |
| ENUM[]    Data        |
| STRING    ext-Data    |
| ENUM[]    Method      |
| STRING    ext-Method  |
| GEOLOC    GeoLocation |
| UNLOCODE UnLocation   |
| STRING    Location    |
+-----------------------+
```

Figure 5: The Analyzer class

The Analyzer class has the following attributes:

| | |
|---|---|
| IP | Mandatory. Analyzer IP address. |

| | |
|---|---|
| Name | Mandatory. Name of the analyzer, which must be reasonably unique, however still bear some meaningful sense.<br>This attribute usually denotes the hierarchy of organizational units the detector belongs to and its own name. It MAY also be used to distinguish multiple analyzers running with the same IP address. |
| Hostname | Optional. Hostname of this analyzer.<br>SHOULD be a fully-qualified domain name. |
| Model | Optional. Analyzer model description (usually its generic name, brand and version). |
| Type | Optional. Analyzer type. |

| Rank | Keyword | Description |
|---|---|---|
| 0 | Cyber | The analyzer specializes in the detection of cyber incidents |
| 1 | Physical | The analyzer specializes in the detection of physical incidents |
| 2 | Availability | The analyzer specializes in the detection of availability incidents |
| 3 | Combined | The analyzer specilizes in detections that combine data from multiple domains (e.g. a combination of Cyber and Availability data) |

Table 5: Analyzer types

| | |
|---|---|
| Category | Optional. Analyzer categories. |

| Rank | Keyword | Description |
|---|---|---|
| 0 | 1DLiS | 1D LIDAR Sensor |
| 1 | 2DLiS | 2D LIDAR Sensor |
| 2 | 3DLiS | 3D LIDAR Sensor |
| 3 | 1DLaS | 1D Laser Sensor |
| 4 | 2DLaS | 2D Laser Sensor |
| 5 | 3DLaS | 3D Laser Sensor |
| 6 | VAD | Voice Activity Detection |
| 7 | HAR | Human Activity Detection |
| 8 | FRC | Face Recognition Camera |
| 9 | VNIR | Visible and Near-InfraRed |
| 10 | SWIR | Short Wavelength InfraRed |
| 11 | MWIR | Middle Wavelength InfraRed |
| 12 | LWIR | Long Wavelength InfraRed |
| 13 | ADS | Anti-Drone System |
| 14 | ODC | Object Detection Camera |
| 15 | DDOS | Anti-DDoS (Distributed Denial of Service) protection |
| 16 | SPAM | Spam detection, phishing detection, etc. |
| 17 | AV | Signature-based virus/malware detection |
| 18 | EDR | Endpoint Detection and Response |
| 19 | FW | Firewall |

| Rank | Keyword | Description |
|---|---|---|
| 20 | NIDS | Network Intrusion Detection System |
| 21 | HIDS | Host Intrusion Detection System |
| 22 | WIDS | Wi-Fi Intrusion Detection System |
| 23 | PROX | Proxy, e.g. detection of violations to the company's security policy |
| 24 | WAF | Web Application Firewall |
| 25 | HPT | Honeypot |
| 26 | LOG | Log analyzer |
| 27 | IAM | Identity and Access Management tool |
| 28 | VPN | Devices/tools related to Virtual Private Network |
| 29 | ETL | Extract-Transform-Load tools |
| 30 | RASP | Runtime Application Self-Protection |
| 31 | BAST | Clientless Remote Desktop Gateway / administration bastions |
| 32 | NAC | Devices/tools related to Network Access Control |
| 33 | SIEM | Security Information and Event Management systems |
| 34 | NMS | Network Management Systems |
| 35 | ext-value | A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. (see Section 4.1.1) |

Table 6: Analyzer categories

ext-Category
Optional. A means by which to extend the Category attribute. (see Section 4.1.1)

Data
Optional. Type of data analyzed during the detection.

| Rank | Keyword | Description |
|---|---|---|
| 0 | Light | |
| 1 | Noise | |
| 2 | Touch | |
| 3 | Images | |
| 4 | Vibrations | |
| 5 | Lidar | |
| 6 | Thermic | |
| 7 | Seismic | |
| 8 | Temperature | |
| 9 | Rain | |
| 10 | Water | |
| 11 | Humidity | |
| 12 | Particles | |
| 13 | Contact | |

| Rank | Keyword | Description |
|---|---|---|
| 14 | MagneticField | |
| 15 | Acoustics | |
| 16 | Fog | |
| 17 | External | |
| 18 | Reporting | |
| 19 | Connection | |
| 20 | Datagram | |
| 21 | Content | |
| 22 | Data | |
| 23 | File | |
| 24 | Flow | |
| 25 | Log | |
| 26 | Protocol | |
| 27 | Host | |
| 28 | Network | |
| 29 | Alert | |
| 30 | Relay | |
| 31 | Auth | |
| 32 | SNMP | |
| 33 | ext-value | A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. (see Section 4.1.1) |

Table 7: Analyzer data

ext-Data Optional. A means by which to extend the Data attribute. (see Section 4.1.1)

Method Optional. Detection method.

| Rank | Keyword | Description |
|---|---|---|
| 0 | Biometric | |
| 1 | Policy | |
| 2 | Heat | |
| 3 | Movement | |
| 4 | Blackhole | |
| 5 | Signature | |
| 6 | Statistical | |
| 7 | Heuristic | |
| 8 | Integrity | |
| 9 | Honeypot | |
| 10 | Tarpit | |
| 11 | Recon | |
| 12 | Correlation | |
| 13 | Monitor | |
| 14 | AI | |
| 15 | Threshold | |
| 16 | ext-value | A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. (see Section 4.1.1) |

Table 8: Analyzer methods

| | |
|---|---|
| ext-Method | Optional. A means by which to extend the Method attribute. (see Section 4.1.1) |
| GeoLocation | Optional. GPS coordinates for the analyzer. |
| UnLocation | Optional. Standard UN/Locode for the analyzer. |
| Location | Optional. Internal name for the location of the analyzer. |

## 5.4. The Sensor Class

The Sensor class describes the module that captured the data before sending it to an analyzer. The Sensor may be a subpart of the Analyzer.

```
+---------------------+
|        Sensor       |
+---------------------+
| IP       IP         |
| STRING   Name       |
| STRING   Hostname   |
| STRING   Model      |
| GEOLOC   GeoLocation |
| UNLOCODE UnLocation  |
| STRING   Location   |
| STRING   CaptureZone |
+---------------------+
```

Figure 6: The Sensor class

The Sensor class has the following attributes:

IP
: Mandatory. The sensor's IP address.

Name
: Mandatory. Name of the sensor, which must be reasonably unique, however still bear some meaningful sense.
This attribute usually denotes the hierarchy of organizational units the sensor belongs to and its own name. It MAY also be used to distinguish multiple sensors running with the same IP address.

Hostname
: Optional. The sensor's hostname.
This SHOULD be a fully qualified domain name, but may not conform exactly because values extracted from logs, messages, DNS, etc. may themselves be malformed.
An empty string MAY be used to explicitly state that this value was inquired but not found (missing DNS entry).

Model
: Optional. The sensor model's description (usually its generic name, brand and version).

GeoLocation
: Optional. GPS coordinates for the analyzerr.

UnLocation
: Optional. Standard UN/Locode for the sensor.

Location
: Optional. Internal name for the location of the sensor.

CaptureZone
: Optional. A string that describes the "capture zone" of the sensor, as a JSON-serialized string.
Depending on the type of sensor, the capture zone may for instance refer to:

- A JSON object describing a camera's settings (elevation, horizontal and vertical field of view, azimuth, etc.)
- A description of the IP network where packet capture is taking place.

## 5.5. The Source Class

The Source class describes the origin(s) of the event(s) leading up to the alert.

```
+-----------------------+
|         Source        |
+-----------------------+
| IP          IP        |
| STRING      Hostname  |
| STRING      Note      |
| STRING[]    TI        |
| STRING      User      |
| EMAIL       Email     |
| PROTOCOL[] Protocol   |
| INT[]       Port      |
| GEOLOC      GeoLocation |
| UNLOCODE    UnLocation |
| STRING      Location  |
| ID[]        Attachment |
+-----------------------+
```

Figure 7: The Source class

The Source class has the following attributes:

IP
: Optional. Source IP address.

Hostname
: Optional. Hostname of this source.
This SHOULD be a fully qualified domain name, but may not conform exactly because values extracted from logs, messages, DNS, etc. may themselves be malformed.
An empty string MAY be used to explicitly state that this value was inquired but not found (missing DNS entry).

Note
: Optional. Free text human-readable additional note for this source.

TI
: Optional. Threat Intelligence data about the source.
Values in this list MUST use the format "attribute:origin", where "attribute" refers to the attribute inside this source found inside a Threat Intelligence database, and "origin" contains a short identifier for the Threat Intelligence database. E.g. "IP:Dshield".
Please note that the same attribute may appear multiple times inside the list (because a match was found in multiple Threat Intelligence databases).

User
: Optional. User ID or login responsible for the alert.

Email
: Optional. Email address responsible for the alert.
E.g. the value of the "Reply-To" or "From" header inside a phishing e-mail.

Protocol
: Optional. Protocols related to connections from/to this source.
If several protocols are stacked, they MUST be ordered from the lowest (the closest to the medium) to the highest (the closest to the application) according to the ISO/OSI model.

Port
: Optional. Source ports involved in the alert.
Values in this list MUST be integers and MUST be in the range 1-65535.

GeoLocation
: Optional. GPS coordinates for the source.

UnLocation
: Optional. Standard UN/Locode for the source.

Location
: Optional. Internal name for the location of the source.

Attachment
: Optional. Identifiers for attachments related to this source.
Each identifier listed here MUST match the "Name" attribute for one of the attachments described using the Attachment class (Section 5.8).

## 5.6. The Target Class

The Target class describes the target(s) impacted by the event(s) leading up to the alert.

```
+------------------------+
|         Target         |
+------------------------+
| IP         IP          |
| STRING     Hostname    |
| STRING     Note        |
| STRING     Service     |
| STRING     User        |
| EMAIL      Email       |
| INT[]      Port        |
| GEOLOC     GeoLocation |
| UNLOCODE   UnLocation  |
| STRING     Location    |
| ID[]       Attachment  |
+------------------------+
```

Figure 8: The Target class

The Target class has the following attributes:

IP
: Optional. Target IP address.

Hostname
: Optional. Hostname of this target.
This SHOULD be a fully qualified domain name, but may not conform exactly because values extracted from logs, messages, DNS, etc. may themselves be malformed.
An empty string MAY be used to explicitly state that this value was inquired but not found (missing DNS entry).

Note
: Optional. Free text human-readable additional note for this target.

Service
: Optional. Service or process impacted by the alert.

User
: Optional. User ID or login targeted by the alert.

Email
: Optional. Email address targeted by the alert.
E.g. the value of the "To" header inside a phishing e-mail.

Port
: Optional. Target ports involved in the alert.
Values in this list MUST be integers and MUST be in the range 1-65535.

GeoLocation
: Optional. GPS coordinates for the target.

UnLocation
: Optional. Standard UN/Locode for the target.

Location
: Optional. Internal name for the location of the target.

Attachment
: Optional. Identifiers for attachments related to this target.
Each identifier listed here MUST match the "Name" attribute for one of the attachments described using the Attachment class (Section 5.8).

## 5.7. The Vector Class

The Vector class describes the vector(s) of the event(s) leading up to the alert. • Name, location, description, …

```
+-------------------------+
|         Vector          |
+-------------------------+
| ENUM[]     Category     |
| STRING     ext-Category |
| STRING     Name         |
| STRING     Note         |
| STRING[]   TI           |
| GEOLOC     GeoLocation  |
| FLOAT      GeoRadius    |
| UNLOCODE   UnLocation   |
| STRING     Location     |
| ID[]       Attachment   |
+-------------------------+
```

Figure 9: The Vector class

The Vector class has the following attributes:

Category Mandatory. Category for the detected "vector".

| Rank | Keyword | Description |
|---|---|---|
| 0 | Unknown | |
| 1 | Face | |
| 2 | RunningMan | |
| 3 | Human | |
| 4 | Man | |
| 5 | Woman | |
| 6 | Children | |
| 7 | Animal | |
| 8 | Object | |
| 9 | Blast | |
| 10 | Fire | |
| 11 | Wind | |
| 12 | Snow | |
| 13 | Rain | |
| 14 | Chemical | |
| 15 | Smoke | |
| 16 | Vapors | |
| 17 | Drug | |
| 18 | Device | |
| 19 | Drone | |
| 20 | Car | |
| 21 | Truck | |
| 22 | Vehicle | |
| 23 | Bird | |
| 24 | Storm | |
| 25 | HighTemperature | |
| 26 | Artifact | |
| 27 | Autonomous System | |
| 28 | Directory | |
| 29 | Domain Name | |
| 30 | Email Address | |
| 31 | Email Message | |
| 32 | File | |
| 33 | IPv4 Address | |

| Rank | Keyword | Description |
| --- | --- | --- |
| 34 | IPv6 Address | |
| 35 | Mutex | |
| 36 | Network Traffic | |
| 37 | Process | |
| 38 | URL | |
| 39 | User Account | |
| 40 | Windows Registry Key | |
| 41 | X509 Certificate | |
| 42 | ext-value | A value used to indicate that this attribute is extended and the actual value is provided using the corresponding ext-* attribute. (see Section 4.1.1) |

Table 9: Vector categories

ext-Category
: Optional. A means by which to extend the Category attribute. (see Section 4.1.1)

Name
: Optional. Name of the detected vector or "Unknown".
Please note that this name does not need to be unique across vectors.

Note
: Optional. Free text human-readable additional note for this vector.

TI
: Optional. Threat Intelligence data about the vector.
Values in this list MUST use the format "attribute:origin", where "attribute" refers to the attribute inside this vector found inside a Threat Intelligence database, and "origin" contains a short identifier for the Threat Intelligence database. E.g. "Name:FBI-Wanted".
Please note that the same attribute may appear multiple times inside the list (because a match was found in multiple Threat Intelligence databases).

GeoLocation
: Optional. GPS coordinates for the vector.

GeoRadius
: Optional. Estimated radius around the provided geolocation in meters.
This attribute can be interpreted as an error margin related to the detection of this vector.

UnLocation
: Optional. Standard UN/Locode for the vector.

Location
: Optional. Internal name for the location of the vector.

Attachment
: Optional. Identifiers for attachments related to this vector.
Each identifier listed here MUST match the "Name" attribute for one of the attachments described using the Attachment class (Section 5.8).

## 5.8. The Attachment Class

The Attachment class contains additional data which was captured in relation with the event.

```
+---------------------------+
|         Attachment        |
+---------------------------+
| ID         Name           |
| STRING     FileName       |
| HASH[]     Hash           |
| INT        Size           |
| URI[]      Ref            |
| URI[]      ExternalURI    |
| STRING     Note           |
| MEDIATYPE  ContentType    |
| STRING     ContentEncoding |
| STRING     Content        |
+---------------------------+
```

Figure 10: The Attachment class

The Attachment class has the following attributes:

Name
: Mandatory. A unique identifier among attachments that can be used to reference this attachment from other classes using the "Attachment" attribute.

FileName
: Optional. Attachment filename.
This will usually be the original name of the captured file or the name of the file containing the captured content (e.g. a packet capture file).

Hash
: Optional. A list of hash results for the attachment's Content.
The values in this list are computed by taking the raw value of the attachment's "Content" attribute. The hash result is computed before any other transformation (e.g. Base64 encoding) is applied to the content, so that a receiving IDMEF system may reverse the transformation, apply the same hashing function and obtain the same hash result. See also the definition for the "ContentEncoding" attribute below.
It is RECOMMENDED that compatible implementations use one of the hashing functions from the SHA-2 [RFC6234] or SHA-3 [NIST.FIPS.202] families to compute the hash results in this list.

Size
: Optional. Length of the content (in bytes).
This value MUST be a non-negative integer.

Ref
: Optional. References to sources of information related to the alert and/or vulnerability, and specific to this attachment.

ExternalURI
: Optional. If the attachment's content is available and/or recognizable from an external resource, this is the URI (usually a URL) to that resource.
This MAY also be a URN in a registered or unregistered ad-hoc namespace bearing reasonable information value and uniqueness, such as "urn:mhr:55eaf7effadc07f866d1eaed9c64e7ee49fe081a" or "magnet:?xt=urn:sha1:YNCKHTQCWBTRNJIV4WNAE52SJUQCZO5C".

Note
: Optional. Free text human-readable additional note for this attachment.

ContentType
: Optional. Internet Media Type of the attachment.
For compatibility reasons, implementations SHOULD prefer one of the well-known media types registered in IANA .

ContentEncoding — Optional. Content encoding.
The following encodings are defined in this version of the specification:

- "json": The content refers to a JSON object which has been serialized to a string using the serialization procedure defined in [RFC8259].
- "base64": The content has been serialized using the Base64 encoding defined in [RFC4648].

The "base64" encoding SHOULD be used when the content contains binary data.
If omitted, the "json" encoding MUST be assumed.

Content — Optional. The attachment's content, in case it is directly embedded inside the message.
For large attachments, it is RECOMMENDED that implementations make use of the "ExternalURI" attribute to refererence a copy of the content saved in an external storage mechanism.

## 5.9. The JavaScript Object Notation Serialization Method

This serialization method aims to convert IDMEFv2 messages to a format that is easy to parse and process, both by software/hardware processors, as well as humans. It relies on the the JavaScript Object Notation (JSON) Data Interchange Format defined in [RFC8259].

Conforming implementations MUST implement all the requirements specified in [RFC8259].

In addition, the following rules MUST be observed when serializing an IDMEFv2 message:

- The top-level Alert class (Section 4.2) is represented as a JSON object ([RFC8259]). This JSON object is returned to the calling process at the end of the serialization process.
- Aggregate classes are represented as JSON objects and stored as members of the top-level JSON object, using the same name as in the IDMEF data model. E.g. the appears under the name "Analyzer" inside the top-level JSON object.
- Attributes are stored as members of the JSON object representing the class they belong to, using the same name as in the IDMEF data model. E.g. the "Version" attribute from the is stored under the name "Version" inside the top-level JSON object.
- Lists from the IDMEF data model are represented as JSON arrays ([RFC8259]). This also applies to aggregate classes where a list is expected. E.g. the "Sensor" member inside the top-level JSON object contains a list of objects, where each object represents an instance of the .
- The various string-based data types listed in Section 3 are represented as JSON strings ([RFC8259]). Please note that the issues outlined in [RFC8259] regarding strings processing also apply here.
- IDMEF attributes with the "NUMBER" data type are represented as JSON numbers ([RFC8259]).

## 5.10. Attributes completeness

The next table shows when each attributes is required depending on it's Type: physical, cyber or availability.

Legend:

- R: REQUIRED
- r: Recommanded
- o: Optional
- NA: Not Applicable

| **Attributes** | **Type** | **Phy** | **Cyb** | **Avail** |
|---|---|---|---|---|
| **Alert** | | | | |
| Version | String | R | R | R |
| ID | UUID | R | R | R |
| OrganisationName | String | o | o | o |
| OrganisationID | String | o | o | o |

| **Attributes** | **Type** | **Phy** | **Cyb** | **Avail** |
|---|---|---|---|---|
| **Alert** | | | | |
| EntityName | String | o | o | o |
| EntityID | String | o | o | o |
| Category | Array of ENUM | r | r | r |
| Cause | ENUM | r | r | r |
| Description | String | r | r | r |
| Status | ENUM | r | r | r |
| Priority | ENUM | r | r | r |
| Confidence | Number | o | o | o |
| Note | String | o | o | o |
| CreateTime | Timestamp | R | R | R |
| StartTime | Timestamp | r | r | r |
| CeaseTime | Timestamp | o | o | o |
| DeleteTime | Timestamp | o | o | o |
| AltNames | Array of String | o | o | o |
| AltCategory | Array of String | o | o | o |
| Ref | Array of URI | o | o | o |
| CorrelID | Array of UUID | o | o | o |
| AggrCondition | Array of String | o | o | o |
| PredID | Array of UUID | o | o | o |
| RelID | Array of UUID | o | o | o |

Table 10: Attributes completness - Alert

| **Attributes** | **Type** | **Phy** | **Cyb** | **Avail** |
|---|---|---|---|---|
| **Analyzer** | **Class** | **R** | **R** | **R** |
| IP | IPAddress | R | R | R |
| Name | String | R | R | R |
| Hostname | String | r | r | r |
| Type | ENUM | r | r | r |
| Model | String | R | R | R |
| Category | Array of ENUM | R | R | R |
| Data | Array of ENUM | R | R | R |
| Method | Array of ENUM | R | R | R |
| GeoLocation | GeoLocation | r | o | o |
| UnLocation | UN/LOCODE | o | o | o |
| Location | String | o | o | o |

Table 11: Attributes completness - Analyzer

| **Attributes** | **Type** | **Phy** | **Cyb** | **Avail** |
|---|---|---|---|---|
| **Sensor** | **Array of Class** | **o** | **o** | **o** |
| IP | IPAddress | R | R | R |
| Name | String | R | R | R |
| Hostname | String | r | r | r |
| Model | String | R | R | R |
| UnLocation | UN/LOCODE | o | o | o |
| Location | String | o | o | o |
| CaptureZone | String | o | o | o |

Table 12: Attributes completness - Sensor

| Attributes | Type | Phy | Cyb | Avail |
|---|---|---|---|---|
| **Source** | **Array of Class** | **o** | **o** | **o** |
| UnLocation | UN/LOCODE | o | o | NA |
| Location | String | o | o | NA |
| GeoLocation | GeoLocation | NA | o | NA |
| Note | String | o | o | o |
| TI | Array of String | o | o | o |
| IP | IPAddress | NA | r | NA |
| Hostname | String | NA | r | NA |
| User | String | NA | o | NA |
| Email | String | NA | o | NA |
| Protocol | Array of ProtocolName | NA | o | NA |
| Port | Array of Port | NA | o | NA |
| Attachment | Array of AttachmentName | NA | o | NA |

Table 13: Attributes completness - Source

| Attributes | Type | Phy | Cyb | Avail |
|---|---|---|---|---|
| **Target** | **Array of Class** | **o** | **R** | **R** |
| UnLocation | UN/LOCODE | o | o | o |
| Location | String | r | o | o |
| GeoLocation | GeoLocation | o | o | o |
| Note | String | o | o | o |
| IP | IPAddress | o | r | R |
| Hostname | String | o | r | r |
| Service | String | NA | o | r |
| User | String | NA | o | NA |
| Email | String | NA | o | NA |
| Port | Array of Port | NA | o | o |
| Attachment | Array of AttachmentName | NA | o | o |

Table 14: Attributes completness - Target

| Attributes | Type | Phy | Cyb | Avail |
|---|---|---|---|---|
| **Vector** | **Array of Class** | **o** | **o** | **o** |
| Category | Array of ENUM | R | R | NA |
| TI | Array of String | o | o | NA |
| Name | String | o | NA | NA |
| Size | ENUM | o | NA | NA |
| UnLocation | UN/LOCODE | o | NA | NA |
| GeoLocation | GeoLocation | o | NA | NA |
| GeoRadius | Number | o | NA | NA |
| Location | String | r | NA | NA |
| Note | String | o | NA | NA |
| Attachment | Array of AttachmentName | o | o | o |

Table 15: Attributes completness - Vector

| Attributes | Type | Phy | Cyb | Avail |
|---|---|---|---|---|
| **Attachment** | **Array of Class** | **o** | **o** | **o** |
| Name | String | R | R | R |

| **Attributes** | **Type** | **Phy** | **Cyb** | **Avail** |
|---|---|---|---|---|
| **Attachment** | **Array of Class** | **o** | **o** | **o** |
| FileName | String | o | o | o |
| Hash | Array of Hashes | r | r | r |
| Size | Number | r | r | r |
| Ref | Array of URI | o | o | o |
| ExternalURI | Array of URI | o | o | o |
| Note | String | o | o | o |
| ContentType | MediaType | o | o | o |
| ContentEncoding | String | r | r | r |
| Content | String | o | o | o |

Table 16: Attributes completness - Attachment

| **Attributes** | **Type** | **Phy** | **Cyb** | **Avail** |
|---|---|---|---|---|
| **Name** | **String** | **R** | **R** | **R** |
| Reference | String | r | r | r |
| Content | String | R | R | R |

Table 17: Attributes completness

# 6. Security Considerations

This document describes a data representation for exchanging security-related information between incident detection system implementations. Although there are no security concerns directly applicable to the format of this data, the data itself may contain security-sensitive information whose confidentiality, integrity, and/or availability may need to be protected.

This suggests that the systems used to collect, transmit, process, and store this data should be protected against unauthorized use and that the data itself should be protected against unauthorized access.

The underlying messaging format and protocol used to exchange instances of the IDMEF MUST provide appropriate guarantees of confidentiality, integrity, and authenticity. The use of a standardized security protocol is encouraged.

The draft-lehmann-idmefv2-https-transport-01.txt document defines the transportation of IDMEF over HTTPs that provides such security.

# 7. IANA Considerations

This document creates 10 identically structured registries to be managed by IANA:

- Name of the registry group: "Incident Detection Message Exchange Format v2 (IDMEF)"
- URL of the registry: <http://www.iana.org/assignments/idmefv2>
- Namespace format: A registry entry consists of:
  - Rank. A uniq integer for this namespace. Range starts at 0 and ends at the length of this list. The maximum length of this list is 255.
  - Keyword. A keyword for a given IDMEF attribute. It MUST conform to the formatting specified by the IDMEF "ENUM" data type (Section 3.3.1).
  - Description. A short description of the enumerated keyword.
  - Reference. An optional list of URIs to further describe the value.
- Allocation policy: Expert Review per [RFC8126]. This reviewer will ensure that the requested registry entry conforms to the prescribed formatting. The reviewer will also ensure that the entry is an appropriate value for the attribute per the information model (Section 5).

The registries to be created are named in the "Registry Name" column of Table 18. Each registry is initially populated with ranks, keywords and descriptions that come from an attribute specified in the IDMEF model (Section 5). The initial Ranks, Keywords and Description fields of a given registry are listed in "Initial Values". The "Initial Values" column points to a table in this document that lists and describes each enumerated keyword. Each enumerated keyword in the table gets a corresponding entry in a given registry. The initial value of the Reference field of every registry entry described below should be this document.

| Registry Name | Initial Values |
|---|---|
| Alert-Category | Table 1 (Alert class (Section 5.2)) |
| Alert-Cause | Table 2 (Alert class (Section 5.2)) |
| Alert-Priority | Table 4 (Alert class (Section 5.2)) |
| Alert-Status | Table 3 (Alert class (Section 5.2)) |
| Analyzer-Category | Table 6 (Alert class (Section 5.2)) |
| Analyzer-Data | Table 7 (Analyzer class (Section 5.3)) |
| Analayzer-Method | Table 8 (Analyzer class (Section 5.3)) |
| Analyzer-Type | Table 5 (Analyzer class (Section 5.3)) |
| Vector-Category | Table 9 (Vector (Section 5.7)) |

Table 18: IANA Enumerated Value Registries

# 8. Acknowledgement

The following groups and individuals contributed to the creation of this document and should be recognized for their efforts.

- The former Prelude SIEM team : Thomas Andrejak & François Poirotte (Co-authors of the first version of this document), Antoine Luong, Song Tran, Selim Menouar and Camille Gardet
- The core members of the SECEF (SECurity Exchange Format) consortium : Herve Debar (Author of RFC 4765 - IDMEFv1), Guillaume Hiet and François Dechelle
- The H2020 7SHIELD project (Safety and Security Standards of Space Systems, ground Segments and Satellite data assets , via prevention, detection, response and mitigation of physical and cyber threats) who implemented in real scale first versions of IDMEFv2 on five pilots around Europe helping greatly to improve it.
- The CESNET team for their work on the [IDEA0] format (based on IDMEFv1) which inspired multiples concepts to IDMEFv2.
- The [ENISA-RIST] Reference Security Incident Taxonomy Working Group

# 9. References

## 9.1. Normative References

[RFC5321] Klensin, J., "Simple Mail Transfer Protocol", RFC 5321, DOI 10.17487/RFC5321, October 2008, <https://www.rfc-editor.org/info/rfc5321>.

[RFC2046] Freed, N. and N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types", RFC 2046, DOI 10.17487/RFC2046, November 1996, <https://www.rfc-editor.org/info/rfc2046>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC5322] Resnick, P., Ed., "Internet Message Format", RFC 5322, DOI 10.17487/RFC5322, October 2008, <https://www.rfc-editor.org/info/rfc5322>.

[RFC3339] Klyne, G. and C. Newman, "Date and Time on the Internet: Timestamps", RFC 3339, DOI 10.17487/RFC3339, July 2002, <https://www.rfc-editor.org/info/rfc3339>.

[RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, DOI 10.17487/RFC3986, January 2005, <https://www.rfc-editor.org/info/rfc3986>.

[RFC4122] Leach, P., Mealling, M., and R. Salz, "A Universally Unique IDentifier (UUID) URN Namespace", RFC 4122, DOI 10.17487/RFC4122, July 2005, <https://www.rfc-editor.org/info/rfc4122>.

[RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 4291, DOI 10.17487/RFC4291, February 2006, <https://www.rfc-editor.org/info/rfc4291>.

[RFC4648] Josefsson, S., "The Base16, Base32, and Base64 Data Encodings", RFC 4648, DOI 10.17487/RFC4648, October 2006, <https://www.rfc-editor.org/info/rfc4648>.

[RFC5234] Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, DOI 10.17487/RFC5234, January 2008, <https://www.rfc-editor.org/info/rfc5234>.

[RFC5952] Kawamura, S. and M. Kawashima, "A Recommendation for IPv6 Address Text Representation", RFC 5952, DOI 10.17487/RFC5952, August 2010, <https://www.rfc-editor.org/info/rfc5952>.

[RFC8259] Bray, T., Ed., "The JavaScript Object Notation (JSON) Data Interchange Format", STD 90, RFC 8259, DOI 10.17487/RFC8259, December 2017, <https://www.rfc-editor.org/info/rfc8259>.

[UNICODE] Unicode Consortium, "Unicode Standard", version 14.0.0, September 2021, <https://www.unicode.org/versions/Unicode14.0.0/>.

[ENISA-RIST] European Union Agency for Cybersecurity, "Reference Incident Classification Taxonomy", January 2018, <https://github.com/enisaeu/Reference-Security-Incident-Taxonomy-Task-Force/blob/master/working_copy/humanv1.md>.

[IANA_media_types] IANA, "Media Types", <http://www.iana.org/assignments/media-types>.

[IANA_hash_function_text_names] IANA, "Hash Function Textual Names", <http://www.iana.org/assignments/hash-function-text-names>.

[UN-LOCODE] United Nations Economic Commission for Europe, "UN/LOCODE Code List by Country and Territory", July 2021, <https://unece.org/trade/cefact/unlocode-code-list-country-and-territory>.

### 9.2. Informative References

[RFC4765] Debar, H., Curry, D., and B. Feinstein, "The Intrusion Detection Message Exchange Format (IDMEF)", RFC 4765, DOI 10.17487/RFC4765, March 2007, <https://www.rfc-editor.org/info/rfc4765>.

[RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <https://www.rfc-editor.org/info/rfc8126>.

[RFC6234] Eastlake 3rd, D. and T. Hansen, "US Secure Hash Algorithms (SHA and SHA-based HMAC and HKDF)", RFC 6234, DOI 10.17487/RFC6234, May 2011, <https://www.rfc-editor.org/info/rfc6234>.

[NIST.FIPS.202] Dworkin, Morris J., "SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions", NIST NIST FIPS 202, DOI 10.6028/NIST.FIPS.202, July 2015, <https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.202.pdf>.

[WGS84] National Imagery and Mapping Agency, "Department of Defense World Geodetic System 1984: Its Definition and Relationships with Local Geodetic Systems", Third Edition, 1984, <https://apps.dtic.mil/sti/pdfs/ADA280358.pdf>.

[IDEA0] CESNET, "Intrusion Detection Extensible Alert version 0", September 2015, <https://idea.cesnet.cz/en/definition>.

# Appendix A. Examples

This section contains several examples of events/incidents which may be described using the IDMEF Data Model defined in.

For each example, the serialization method listed in Section 5 was used on the original IDMEF message to produce a JSON representation.

## A.1. Physical intrusion

Listing 1 describes an incident where an unidentified man was detected on company premises near the building where server room A is located.

```
{
  "Version": "2.D.V0X",
  "ID": "819df7bc-35ef-40d8-bbee-1901117370b1",
  "Description": "Potential intruder detected",
  "Priority": "Low",
  "Status": "Incident",
  "Cause": "Malicious",
  "CreateTime": "2021-05-10T16:52:13.075994+00:00",
  "StartTime": "2021-05-10T16:52:13+00:00",
  "Category": [
    "Intrusion.Burglary"
  ],
  "Analyzer": {
    "Name": "BigBrother",
    "Hostname": "bb.acme.com",
    "Type": "Physical",
    "Model": "Big Brother v42",
    "Category": [
      "HAR",
      "FRC"
    ],
    "Data": [
      "Images"
    ],
    "Method": [
      "Movement",
      "Biometric",
      "AI"
    ],
    "IP": "192.0.2.1"
  },
  "Sensor": [
    {
      "IP": "192.0.2.2",
      "Name": "Camera #23",
      "Model": "SuperDuper Camera v1",
      "Location": "Hallway to server room A1"
    }
  ],
  "Source": [
    {
      "Note": "Black Organization, aka. APT 4869"
    }
  ],
  "Vector": [
    {
      "Category": ["Man"],
      "TI": ["Name:FBI-Wanted"],
      "Name": "John Doe",
      "Note": "Codename Vodka, known henchman for APT 4869",
      "Location": "Hallway to server room A1",
      "Attachment": ["pic01", "wanted"]
    }
  ],
  "Attachment": [
    {
      "Name": "wanted",
      "FileName": "fbi-wanted-poster.jpg",
      "Size": 1234567,
      "Ref": ["https://www.fbi.gov/wanted/topten"],
```

## A.2. Cyberattack

Listing 2 describes an incident related to a potential bruteforce attack against the "root" user account of the server at 192.0.2.2 and 2001:db8::/32.

```
{
  "Version": "2.D.V0X",
  "ID": "819df7bc-35ef-40d8-bbee-1901117370b2",
  "Description": "Potential bruteforce attack on root user account",
  "Priority": "Medium",
  "CreateTime": "2021-05-10T16:55:29.196408+00:00",
  "StartTime": "2021-05-10T16:55:29+00:00",
  "Category": [
    "Attempt.Login"
  ],
  "Analyzer": {
    "Name": "SIEM",
    "Hostname": "siem.acme.com",
    "Type": "Cyber",
    "Model": "Concerto SIEM 5.2",
    "Category": [
      "SIEM",
      "LOG"
    ],
    "Data": [
      "Log"
    ],
    "Method": [
      "Monitor",
      "Signature"
    ],
    "IP": "192.0.2.1"
  },
  "Sensor": [
    {
      "IP": "192.0.2.5",
      "Name": "syslog",
      "Hostname": "www.acme.com",
      "Model": "rsyslog 8.2110",
      "Location": "Server room A1, rack 10"
    }
  ],
  "Target": [
    {
      "IP": "192.0.2.2",
      "Hostname": "www.acme.com",
      "Location": "Server room A1, rack 10",
      "User": "root"
    },
    {
      "IP": "2001:db8::/32",
      "Hostname": "www.acme.com",
      "Location": "Server room A1, rack 10",
      "User": "root"
    }
  ]
}
```

## A.3. Server outage

Listing 3 describes an incident where the webserver at "www.example.com" encountered some kind of failure condition resulting in an outage.


```
{
  "Version": "2.D.V0X",
  "ID": "819df7bc-35ef-40d8-bbee-1901117370b3",
  "Description": "A server did not reply to an ICMP ping request",
  "Priority": "Medium",
  "Status": "Incident",
  "Cause": "Unknown",
  "CreateTime": "2021-05-10T16:59:11.875209+00:00",
  "StartTime": "2021-05-10T16:59:11.875209+00:00",
  "Category": [
    "Availability.Outage"
  ],
  "Analyzer": {
    "Name": "NMS",
    "Hostname": "nms.example.com",
    "Type": "Availability",
    "Model": "Concerto NMS 5.2",
    "Category": [
      "NMS"
    ],
    "Data": [
      "Network"
    ],
    "Method": [
      "Monitor"
    ],
    "IP": "192.0.2.1"
  },
  "Target": [
    {
      "IP": "192.168.1.2",
      "Hostname": "www.acme.com",
      "Service": "website",
      "Location": "Server room A1, rack 10"
    }
  ]
}
```


## A.4. Combined incident

Listing 4 describes a combined incident resulting from the correlation of the previous physical, cyber and availability incidents.

```
{
  "Version": "2.D.V0X",
  "ID": "819df7bc-35ef-40d8-bbee-1901117370b4",
  "Description": "Intrusion and Sabotage detected",
  "Prioriy": "High",
  "Status": "Incident",
  "Cause": "Malicious",
  "CreateTime": "2021-05-10T16:59:15.075994+00:00",
  "StartTime": "2021-05-10T16:52:11+00:00",
  "Category": [
    "Intrusion.Burglary",
    "Attempt.Login",
    "Intrusion.SysCompromise",
    "Availability.Outage",
    "Availability.Sabotage",
    "Availability.Failure"
  ],
  "CorrelID": [
    "819df7bc-35ef-40d8-bbee-1901117370b1",
    "819df7bc-35ef-40d8-bbee-1901117370b2",
    "819df7bc-35ef-40d8-bbee-1901117370b3"
  ],
  "Analyzer": {
    "Name": "Correlator",
    "Hostname": "correlator.acme.com",
    "Type": "Combined",
    "Model": "Concerto Hybrid Correlator v5.2",
    "Category": [
    ],
    "Data": [
      "Alert"
    ],
    "Method": [
      "Correlation"
    ],
    "IP": "192.0.2.1"
  },
  "Source": [
    {
      "Note": "Black Organization, aka. APT 4869"
    }
  ],
  "Vector": [
    {
      "Category": ["Man"],
      "TI": ["Name:FBI-Wanted"],
      "Name": "John Doe",
      "Note": "Codename Vodka, known henchman for APT 4869",
      "Size": "Medium"
    }
  ],
  "Target": [
    {
      "Location": "Server room A1"
    },
    {
      "IP": "192.0.2.2",
      "Hostname": "www.acme.com",
      "User": "root"
    }
```

# Appendix B. JSON Validation Schema (Non-normative)

Listing 5 contains a JSON Schema that can be used to validate incoming IDMEF messages prior to processing. Please note that extraneous linebreaks have been included due to formatting constraints.

```
{
    "description": "JSON schema for the Intrusion Detection Message
 Exchange Format (IDMEF) version 2 (revision 2.D.V04)",
    "properties": {
        "Version": {
            "description": "The version of the IDMEF format in use by this
 alert. During the drafts tuning period the version is equal to the draft
 version. Therefore it is \"2.D.V0X\" for Draft V0X.",
            "enum": [
                "2.D.V04"
            ]
        },
        "ID": {
            "description": "Unique identifier for the alert.",
            "$ref": "#/definitions/uuidType"
        },
        "OrganisationName": {
            "description": "Corporate/Main Office Organisation Name Useful
 if alerts are sent to a multi-organisation incident detection system.
 Example: ACME Corporation",
            "type": "string"
        },
        "OrganisationId": {
            "description": "Corporate/Main Office Organisation ID. Where
 possible official organisation ID manage by national authority. Useful
 if alerts are sent to a multi-organisation incident detection system.
 This ID has to be chosen depending on the overall detection perimeter and
 the nature of the monitored organisation (Private/Public, Commercial,
 International, etc.) Examples: OrganisationId in France could be SIREN,
 in England could be CR, Germany could be Handelsregisternummer, Spain
 could be CIF, Italia could be Partita IVA, USA could be EIN, etc.
 Commercial OrganisationId in Europe could be V.A.T ID",
            "type": "string"
        },
        "EntityName": {
            "description": "Entity Name, monitored by the organisation,
 where the incident occurred. Could be a town, region or country name
 or an internal name. Could also be the name of a client for a MSSP
 centralizing it's client incidents in a single system. Do not repeat
 the organisation name in the EntityName Example : \n- ACME HeadQuaters
 is located in Paris France and has a local office in India\n- If the
 incident occurred in the local office : \"OrganisationName\" : \"
 ACME\" , \"EntityName\" : \"India\"\n- If the incident occurred in
 the headquaters : \"OrganisationName\" : \"ACME\", \"EntityName\" :
 \"Headquaters\" (or \"Paris\")",
            "type": "string"
        },
        "EntityId": {
            "description": "Entity ID, monitored by the organisation,
 where the incident occurred. Useful if organisation and entity are not
 directly linked, like a client and a MSSP.",
            "type": "string"
        },
        "Category": {
            "description": "The incident's category & subcategory
 as listed in using the format \"category.subcategory\" (e.g.
 \"Attempt.Exploit\").",
            "type": "array"
            "items": {
                "$ref": "#/definitions/categoryEnum"
```

## Author's Address

**Gilles Lehmann**
Telecom SudParis
FR
EMail: gilles.lehmann@telecom-sudparis.eu