THE MITRE CORPORATION

THE MAEC™ LANGUAGE VERSION 4.1 SPECIFICATION

MAEC DEFAULT VOCABULARIES VERSION 1.1

DESIREE BECK, IVAN KIRILLOV, PENNY CHASE, MITRE

JUNE 12, 2014

Malware Attribute Enumeration and Characterization (MAEC™) is a standardized language for sharing structured information about malware based upon attributes such as behaviors, artifacts, and attack patterns.

By eliminating the ambiguity and inaccuracy that currently exists in malware descriptions and by reducing reliance on signatures, MAEC aims to improve human-to-human, human-to-tool, tool-to-tool, and tool-to-human communication about malware; reduce potential duplication of malware analysis efforts by researchers; and allow for the faster development of countermeasures by enabling the ability to leverage responses to previously observed malware instances.

Acknowledgements

The authors would like to thank the MAEC Community for its input and help in reviewing this document.

Trademark Information

Warnings

MITRE PROVIDES MAEC "AS IS" AND MAKES NO WARRANTY, EXPRESS OR IMPLIED, AS TO THE ACCURACY, CAPABILITY, EFFICIENCY, MERCHANTABILITY, OR FUNCTIONING OF MAEC. IN NO EVENT WILL MITRE BE LIABLE FOR ANY GENERAL, CONSEQUENTIAL, INDIRECT, INCIDENTAL, EXEMPLARY, OR SPECIAL DAMAGES, RELATED TO MAEC OR ANY DERIVATIVE THEREOF, WHETHER SUCH CLAIM IS BASED ON WARRANTY, CONTRACT, OR TORT, EVEN IF MITRE HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.[1]

Feedback

The MAEC development team welcomes any feedback regarding the MAEC Language Default Vocabularies Specification. Please send any comments, questions, or suggestions maec@mitre.org.[2]

[1] For detailed information see [TOU].

[2] For more information about the MAEC Language, please visit [MAEC].

Table of Contents

1 Overview

The Malware Attribute Enumeration and Characterization (MAEC) Language is defined by three data models and a set of default controlled vocabularies[3]. As illustrated in **Error! Reference source not found.**, "MAEC Bundle" is the (lowest) Tier 1 data model; "MAEC Package" is the (middle) Tier 2 data model; and "MAEC Container" is the (highest) Tier 3 data model. All three data models offer a stand-alone output format, so a lower level model can be used without the higher tier data model (although each model level encompasses and makes use of all lower tiers).

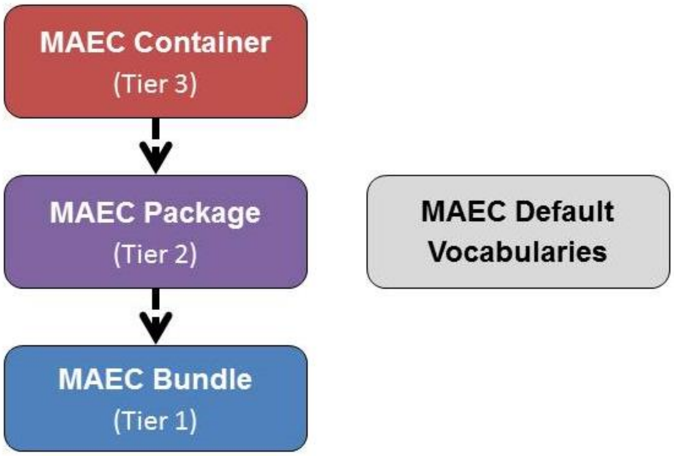

Figure 1-1. MAEC data models

A complete discussion of the structure of the MAEC language can be found in the MAEC Overview [MAEC_O]. In brief:

- MAEC Bundle – provides the ability to capture and share data obtained from the analysis of a single malware instance. Its underlying structure is formed by Actions, Behaviors, and Capabilities.
- MAEC Package – enables a user to capture and share MAEC characterized data for one or more Malware Subjects; in most such cases, the Malware Subjects are related. A Malware Subject is MAEC's representation of a malware instance and all of the known data associated with it, including data derived from analysis and metadata.
- MAEC Container – enables a user to share any collection of MAEC characterized data, including one or more Packages.

[3] Each data model and the default vocabularies are implemented by an XML schema. Other output formats, such as JSON, are being considered for future implementations.

This document serves as the specification for the MAEC default controlled vocabularies. Before we present the vocabularies in Section 2, we provide relevant background information in Subsections 1.1 through 1.6.

1.1 Additional Documents and Information

Numerous overview, specification, and supporting documents are available for the MAEC Language. All documents are shown in **Error! Reference source not found.** Icons are used to indicate whether the material is contained in an actual document ( ) or captured on a Web page ( ). This document is highlighted in yellow.

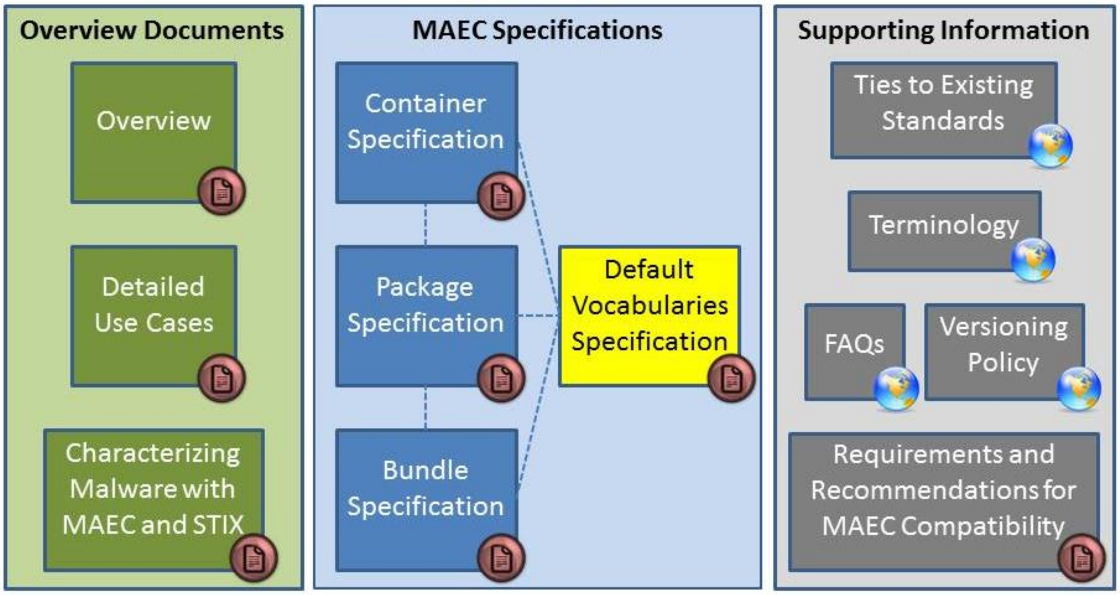

Figure 1-2. MAEC Language v4.1 documents

All documents can be found on the MAEC Website [MAEC], and a summary and link to each is provided below:

- Overview: Introduces and motivates MAEC, provides an overview of the MAEC language, and presents a collection of high level use cases [MAEC$_O$].
- Detailed Use Cases: Provides explicit examples to illustrate how MAEC can be used to capture malware information stemming from various forms of malware analysis [EXAM$_D$].
- Characterizing Malware with MAEC and STIX: Describes the use of MAEC and STIX in the context of malware characterization and malware metadata exchange [MAEC$_S$].
- Container Specification: Specification for the MAEC Container data model [SPEC$_C$].
- Package Specification: Specification for the MAEC Package data model [SPEC$_P$].

- Bundle Specification: Specification for the MAEC Bundle data model [MAEC_B]. (This document.)
- Default Vocabulary Specification: Specification for the MAEC Default Vocabularies [SPEC_V].
- Ties to Existing Standards: Provides an overview of how MAEC is related to MMDEF, CybOX, CPE, CVE, and STIX [TIES].
- Terminology: Contains terms associated with malware and malware analysis, as well as terminology that is specific to MAEC [TERM].
- FAQs: Frequently asked questions about MAEC including questions about the language, use, relationships to other efforts, and the MAEC community [FAQ].
- Versioning Policy: Details the current methodology for determining whether a revision will require a major version change, a minor version change, or an update version change. Note that the MAEC schemas and default vocabularies are versioned independently of the MAEC Language, and their version numbers may or may not coincide with each other or with that of the MAEC Language [VER].
- Requirements and Recommendations for MAEC Compatibility: Specifies requirements for MAEC-compatible tools, services, and repositories [REQ].

1.2 Data Model Conventions

The following information and conventions are used to define the MAEC data models, and may or may not apply to the particular MAEC data model or vocabularies documented in Section **Error! Reference source not found.**.

1.2.1 Data Model Fields and Types

In Section **Error! Reference source not found.**, we define the types associated with the MAEC default controlled vocabulary fields. It is important to understand that “fields” correspond to the malware-related properties captured in a MAEC document and “types” are used to define and express the underlying data model used in the fields.

1.2.2 XML Attributes and Elements

Our methodology for representing a field as either an attribute or an element in the XML implementation[4] is based primarily on the determination of the complexity of the field. Generally, simple fields such as identifiers, data types, and timestamps are represented as attributes. Complex fields, for example, those that have multiplicity greater than one (such as lists), are represented as elements. However, in this specification we have attempted, as much as possible, to abstract away these XML-specific implementation details to provide a more general view of the MAEC controlled vocabularies.

[4] Each data model and the default vocabularies are implemented in MAEC v4.1 via an XML schema.

1.2.3 Non-MAEC Data Models

MAEC draws several components from the CybOX Language (see [MAEC_O]); consequently, the reader is referred to [CYBOX] for the definitions of these entities. In this specification, we do not define any types that are part of a non-MAEC data model. Instead we make note of the referenced data model's specification and explicitly define only the extensions (i.e., new fields and types) that have been made as an extension of the base type.

1.2.4 Primitive Data Types

The following primitive datatypes are used in the MAEC Language.

- binary – Data of this type conforms to the World Wide Web Consortium (W3C) Recommendation for hex-encoded binary data [W3C_1].
- boolean – Data of this type conforms to the W3C Recommendation for boolean data [W3C_2].
- double – Data of this type conforms to the W3C Recommendation for double data [W3C_3].
- float – Data of this type conforms to the W3C Recommendation for float data [W3C_4].
- int – Data of this type conforms to the W3C Recommendation for integer data [W3C_5].
- QName – Data of this type conforms to the W3C Recommendation for an XML namespace-qualified name [W3C_6].
- string – Data of this type conforms to the W3C Recommendation for string data [W3C_7].
- unsigned int – Data of this type conforms to the W3C Recommendation for unsigned int data [W3C_8].
- URI – Data of this type conforms to the W3C Recommendation for anyURI data [W3C_9].
- dateTime – Data of this type represents a time value that conforms to the yyyy-mm-ddThh:mm:ss format.

1.3 Controlled Vocabularies

Some of the fields defined in the MAEC schemas are of type `cyboxCommon:ControlledVocabularyStringType`. A field of this type is implemented through the `xsi:type` XML abstract type extension mechanism. The default vocabulary applicable to the particular type will be provided in the "Description" column of the property table. Default vocabularies are defined in the maec_default_vocabularies.xsd file available at [REL_D]. Please see Section 2 for more information.

1.4 ID Formats

In MAEC v4.1, all MAEC IDs are captured and formatted as XML QNames[5]. Each such ID includes both a namespace portion (optional) and an ID portion (required), separated by a colon (":"). The recommended approach to creating a MAEC ID is to define a producer namespace and namespace prefix and then use the form:

```
[ns prefix]:[construct type]-[GUID]
```

The "ns prefix" SHOULD be a namespace prefix bound to a namespace owned/controlled by the producer of the content. For consistency across MAEC documents, the "construct type" SHOULD correspond to the labels provided in **Error! Reference source not found.** below (datatypes are defined in MAEC v4.1 unless otherwise indicated). Finally, the "GUID" SHOULD correspond to a globally unique ID. For example, a MAEC Bundle could have the following ID:

```
somecompany:bundle-2f44522e-8164-4050-8e13-e01f9a
```

In order to use this approach, the namespace and prefix MUST be defined in the head of the XML document, e.g., `xmlns:somecompany="http://company.example.com"`.

This format provides high assurance that IDs will be both meaningful and unique. Meaning comes from the producer namespace, which denotes who is producing it, as well as the construct type, which denotes to what the ID pertains. Uniqueness is achieved when the meaningful portion is combined with a globally unique ID.

[5] In MAEC v4.1, restrictions on ID syntax have been lifted in all IDs used in MAEC types so that all MAEC IDs are now compatible with the implementations used in CybOX and STIX. Consequently, the additional schematron and XSL files used in earlier MAEC versions primarily for ID syntax validation have been deprecated.

Table 1-1. Recommended construct type labels

Construct Name	Datatype (defining ID)	Construct Type (in ID)
BUNDLE IDs and IDREFs		
action_collection	ActionCollectionType	action_collection
action_implementation	ActionImplementationType	action_implementation
action_equivalence_reference	BehavioralAction EquivalenceReferenceType	action_equivalence
action	cybox:ActionType	action
behavior	BehaviorType	behavior
behavior_collection	BehaviorCollectionType	behavior_collection
maec_bundle	BundleType	bundle
candidate_indicator_collection	CandidateIndicatorCollectionType	candidate_indicator_collection
candidate_indicator	CandidateIndicatorType	candidate_indicator
capability	CapabilityType	capability
malware_instance_object_attributes	cybox:ObjectType	object
strategic_objective	CapabilityObjectiveType	objective
tactical_objective	CapabilityObjectiveType	objective
object_collection	ObjectCollectionType	object_collection
process_tree_node	ProcessTreeNodeType	process_tree
object	cybox:ObjectType	object
PACKAGE IDs and IDREFs		
action_equivalence	ActionEquivalenceType	action_equivalence
analysis	AnalysisType	analysis
malware_subject	MalwareSubjectType	malware_subject
object_equivalence	ObjectEquivalenceType	object_equivalence
maec_package	PackageType	package
malware_instance_object_attributes	cybox:ObjectType	object
CONTAINER IDs		
maec_container	ContainerType	container

1.5 XML Implementation

The XML implementation of the MAEC Language data model is documented in a series of XML Schemas.[6] These schemas describe how the information presented in this Specification is formatted and represented as XML. Please refer to the appropriate Schema for more information about a specific XML implementation.

MAEC Container Model
https://maec.mitre.org/language/version4.1/maec-container-schema.xsd

MAEC Package Model
https://maec.mitre.org/language/version4.1/maec-package-schema.xsd

MAEC Bundle Model
https://maec.mitre.org/language/version4.1/maec-bundle-schema.xsd

MAEC Default Vocabularies
https://maec.mitre.org/language/version4.1/maec-default-vocabularies.xsd

The complete listing of XML representation resources can be found on the MAEC website [REL4].

1.6 Document Conventions

The following conventions are used in this document.

1.6.1 Key Words

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].

1.6.2 Fonts

The following font and font style conventions are used in the document:

- Capitalization is used for MAEC high level concepts, which are defined as basic components in the MAEC Overview document [MAEC_O] (see Section 2 in [MAEC_O]).

  Examples: Bundle, Strategic Objective, Malware Subject

[6] XML Schema Part 0: Primer Second Edition http://www.w3.org/TR/xmlschema-0

- The `Courier New` font is used for writing constructs in the MAEC Language Data Model (and related data models).

  Examples: `CandidateIndicatorType`, `Malware_Subject`

  Note that all high level concepts have a corresponding data model construct (e.g., Malware Subject → `Malware_Subject`).

- The '*italic, with single quotes*' font is used for noting values for MAEC Language properties.

  Examples: *'2.1', 'MAEC Default Device Driver Action Names'*

1.6.3 Namespaces

The Bundle, Package, and Container specifications use the concept of namespaces[7] to logically group MAEC constructs throughout the Data Model sections of the documents, as well as other parts of the specifications. The format of these namespaces is `prefix:namespace`, where the prefix is the namespace component, and the namespace is the actual namespace URI. Table 1-2 on page 10 provides a listing of the default namespaces used in MAEC to help provide context as to the particular source data model or vocabulary used in a field. Table 1-2 also lists the relevant version of each of the data models. These namespaces are compatible with XML Namespaces [W3C$_0$], though the MAEC language is not restricted to XML serialization.

1.6.4 UML Diagrams

The Data Model makes use of Unified Modeling Language (UML) diagrams where appropriate, to visually depict relationships for the MAEC Language constructs. Diagrams are included for any construct that inherits from other constructs or has a compositional relationship.

1.6.5 Property Table Notation

Throughout the data model, tables are used to describe each data type and its properties. Each property table will consist of a column of field names to identify the property, a type column to reflect the datatype of the property, a multiplicity column to reflect the allowed number of occurrences of the property, and a description column that will describe the property. In addition:

- Fields that are part of a "choice" relationship (e.g., Field1 OR Field2 is used but not both) will be denoted by a unique letter subscript (e.g., API_Call$_A$, Code$_B$) and single logic expression in the Multiplicity column. For example, if there is a choice of field

[7] Namespaces (computer science): http://en.wikipedia.org/wiki/Namespace_(computer_science)

API_Call$_A$ and Code$_B$, the expression "A(1)|B(0..1)" will indicate that the API_Call field can be chosen with multiplicity 1 or the Code property can be chosen with multiplicity 0..1.

Values in the type column are either primitive datatypes or other types defined in this document. These values will be cross referenced to the base definition of their types.

Table 1-2. Namespace prefixes used by MAEC

Data Model / Vocab	Namespace Prefix	Description	Example
MAEC Bundle v4.1	maecBundle	The MAEC Bundle data model captures the constructs used in a MAEC Bundle.	`maecBundle:ActionType`
MAEC Package v2.1	maecPackage	The MAEC Package data model captures the constructs used in a MAEC Package.	`maecPackage:MalwareSubjectType`
MAEC Container v2.1	maecContainer	The MAEC Container data model captures all MAEC characterized data.	`maecContainer:PackageListType`
MAEC Default Vocabularies v1.1	maecVocabs	The MAEC default vocabularies define types for default controlled vocabularies used within MAEC.	`maecVocabs:FileActionNameVocab`
Malware Metadata Exchange Format (MMDEF) v1.2	metadata	The MMDEF data model captures some constructs used in exchanging malware sample data.	`metadata:fieldDataEntry`
CybOX Core v2.1	cybox	The CybOX core data model captures all the core constructs used in CybOX.	`cybox:ObjectType`
CybOX Common v2.1	cyboxCommon	The CybOX common data model captures common constructs used across CybOX objects and other types.	`cyboxCommon:MeasureSourceType`
CybOX Default Vocabularies v2.1	cyboxVocabs	The CybOX default vocabularies define types for default controlled vocabularies used within CybOX.	`cyboxVocabs:HashNameVocab`
Code Object v2.1	CodeObj	The CybOX Code Object data model is intended to characterize a body of computer code.	`CodeObj:CodeObjectType`
System Object v2.1	SystemObj	The CybOX System Object data model is intended to characterize computer	`SystemObj:SystemObjectType`

		systems (as a combination of both software and hardware).	
Process Object v2.1	ProcessObj	The CybOX Process Object data model is intended to characterize system processes.	`ProcessObj:ProcessObjectType`

2 MAEC Default Controlled Vocabularies

The MAEC Vocabularies represent a set of default controlled vocabularies for use in MAEC content and were created to take advantage of the extension mechanisms provided by the CybOX v2.x controlled vocabulary implementation. These vocabularies are broken out from the MAEC Bundle, Package, and Container schemas to support customized extension and replacement. However, it is expected that most MAEC authors will use the provided default vocabularies, and thus most tools that parse MAEC data SHOULD support those vocabularies where appropriate. Details on using default vocabularies are givein in Section 2.1.

The remaining subsections provide a listing of version 1.1 of the Default Vocabularies and the corresponding enumerations for use within MAEC v4.1. The lists have been grouped according to the higher level MAEC entity associated with the vocabularies: Actions (Section 2.2), Candidate Indicators (Section 2.4), Capabilities (Section 2.5), Malware Subjects (Section 2.7), and Packages (Section 2.8). Because the list of default vocabularies associated with Action `Name` fields is considerably lengthy, it has been captured separately in Section 2.3. Similarly, the list of default vocabularies associated with malware Capability `Property` fields and Strategic and Tactical Objective `Name` fields is captured separately in Section 2.6.

Note that if an enumeration has been updated, the previous version is also included in the Default Vocabularies for backward compatibility. However, only the latest vocabulary version is provided in this document.

2.1 Using Default Vocabularies

MAEC default vocabularies are referenced from MAEC elements by using the `xsi:type` extension mechanism to indicate the default vocabulary that is used in a particular element. For example, to specify the ‘download file’ Action name, one would use the ‘NetworkActionVocab’ default vocabulary, which includes this value in its enumeration:

```
<maecBundle:Action id="maec-example-act-1">
   <cybox:Name xsi:type="maecVocabs:NetworkActionNameVocab-1.0">download file</cybox:Name>
</maecBundle:Action>
```

To use a non-default vocabulary, one may similarly use the `xsi:type` extension mechanism to indicate the custom vocabulary that is used in a particular element. For example, to use a custom ‘foo’ vocabulary in the Action name example above, one would simply add the appropriate namespace (xmlns:fooVocabs="http://foo/fooVocabulary-1" for the sake of this example) and schemalocation declarations to their MAEC document, and then reference and use the namespace like any default vocabulary:

```
<maecBundle:Action id="maec-example-act-2">
   <cybox:Name xsi:type="fooVocabs:fooVocabulary">some custom action</cybox:Name>
</maecBundle:Action>
```

Accordingly, any elements that use the controlled vocabulary implementation can also accept arbitrary values without the specification of any vocabulary, which may be useful in certain instances. This is achieved simply by not specifying the `xsi:type` extension mechanism on the element. To continue with the above examples, we could also specify a custom Action name that is not part of a vocabulary:

```
<maecBundle:Action id="maec-example-act-3">
   <cybox:Name>another custom action</cybox:Name>
</maecBundle:Action>
```

An individual vocabulary may be revised at any time. Revisions to vocabularies will result in the creation of new types with the new version number embedded in the name of those types (e.g., `FileActionNameVocab-1.0` might be updated to `FileActionNameVocab-1.1` or `FileActionNameVocab-2.0`).

2.2 Action-Related Default Vocabularies

The default vocabularies in this section are related to Actions in a MAEC Bundle.

2.2.1 ActionObjectAssociationTypeVocab-1.0

The `ActionObjectAssociationTypeVocab` is the default MAEC vocabulary for Action-Object association types in a MAEC Bundle, which are captured in MAEC Actions via the `Association_Type` field of type `AssociatedObjectType` defined in the CybOX Core schema. Thus, the MAEC `ActionObjectAssociationTypeVocab-1.0` SHOULD be used in place of the CybOX `ActionObjectAssociationVocab-1.0`.

The MAEC ActionObjectAssociationTypeVocab-1.0 extends the ControlledVocabularyStringType defined in the CybOX Common schema. Thus, Association_Type fields that make use of this vocabulary are restricted to the enumerated entries contained in maecVocabs:ActionObjectAssociationTypeEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is MAEC *'Default Action-Object Association Names'*
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#ActionObjectAssociationTypeVocab-1.0.'*

2.2.1.1 ActionObjectAssociationTypeEnum-1.0

The ActionObjectAssociationTypeEnum is a non-exhaustive enumeration of types of Action-Object associations.

Enumeration Value	Description
input	Specifies that the Associated_Object field serves as an input to the Action. This includes cases where an Object is used by the Action or an existing Object is modified by the Action.
output	Specifies that the Associated_Object field serves as an output to the Action. This includes cases where the Object is created anew by the Action or otherwise returned by the Action.
side-effect	Specifies that the Associated_Object field serves as a side-effect resulting from the Action. This includes cases where the Object is modified indirectly by the Action.

2.3 Default Vocabularies for Specific Action Names

The default vocabularies in this section are related to MAEC Action names.

2.3.1 DebuggingActionNameVocab-1.0

The DebuggingActionNameVocab is the default MAEC vocabulary for Action names associated with debugging, which are captured in MAEC Actions via the Name element of the ActionType that is defined in CybOX Core and extended by MAEC's

MalwareActionType in the MAEC Bundle. Thus, for debugging Action names, the MAEC DebuggingActionNameVocab-1.0 SHOULD be used in place of the CybOX ActionNameVocab default vocabulary.

The MAEC DebuggingActionNameVocab-1.0 type extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in maecVocabs:DebuggingActionNameEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Debugging Action Names*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#DebuggingActionNameVocab-1.0*.'

2.3.1.1 DebuggingActionNameEnum-1.0

The DebuggingActionNameEnum is a non-exhaustive enumeration of the different Action names associated with debugging.

Enumeration Value	Description
check for remote debugger	Specifies the defined Action of checking for the presence of a remote debugger.
check for kernel debugger	Specifies the defined Action of checking for the presence of a kernel debugger.

2.3.2 DeviceDriverActionNameVocab-1.1

The DeviceDriverActionNameVocab is the default MAEC vocabulary for Action names associated with device drivers, which are captured in MAEC Actions via the Name element of the ActionType that is defined in CybOX Core, and extended by MAEC's MalwareActionType in the MAEC Bundle. Thus, for device driver Action names, the MAEC DeviceDriverActionNameVocab-1.1 type SHOULD be used in place of the CybOX ActionNameVocab default vocabulary.

The MAEC DeviceDriverActionNameVocab-1.1 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in maecVocabs:DeviceDriverActionNameEnum-1.1; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Device Driver Action Names*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#DeviceDriverActionNameVocab-1.1*.'

2.3.2.1 DeviceDriverActionNameEnum-1.1

The DeviceDriverActionNameEnum is a non-exhaustive enumeration of the different Action names associated with device drivers.

Enumeration Value	Description
load and call driver	Specifies the defined Action of loading a driver into a system and then calling the loaded driver.
load driver	Specifies the defined Action of loading a driver into a system.
unload driver	Specifies the defined Action of unloading a driver from a system.
emulate driver	Specifies the defined Action of emulating an existing driver on a system.

2.3.3 DirectoryActionNameVocab-1.1

The DirectoryActionNameVocab is the default MAEC vocabulary for Action names associated with file directories, which are captured in MAEC Actions via the Name element of ActionType that is defined in CybOX Core, and extended by MAEC's MalwareActionType in the MAEC Bundle. Thus, for directory Action names, the MAEC DirectoryActionNameVocab-1.1 SHOULD be used in place of the CybOX ActionNameVocab default vocabulary.

The MAEC DirectoryActionNameVocab-1.1 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:DirectoryActionNameEnum-1.1; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Directory Action Names*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#DirectoryActionNameVocab-1.1*.'

2.3.3.1 DirectoryActionNameEnum-1.1

The DirectoryActionNameEnum is a non-exhaustive enumeration of the different Action names associated with directories.

Enumeration Value	Description
create directory	Specifies the defined Action of creating a new directory on the filesystem.
delete directory	Specifies the defined Action of deleting an existing directory on the filesystem.
monitor directory	Specifies the defined Action of monitoring an existing directory on the filesystem for changes.
hide directory	Specifies the defined Action of hiding an existing directory.

2.3.4 DiskActionNameVocab-1.1

The DiskActionNameVocab is the default MAEC vocabulary for Action names associated with disks, which are captured in MAEC Actions via the Name element of the ActionType that is defined in CybOX Core, and extended by MAEC's MalwareActionType in the MAEC Bundle. Thus, for disk Action names, the MAEC DiskActionNameVocab-1.1 SHOULD be used in place of the CybOX ActionNameVocab default vocabulary.

The MAEC DiskActionNameVocab-1.1 type extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:DiskActionNameEnum-1.1; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Disk Action Names*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#DiskActionNameVocab-1.1*.'

2.3.4.1 DiskActionNameEnum-1.1

The `DiskActionNameEnum` is a non-exhaustive enumeration of the different Action names associated with hard disks.

Enumeration Value	Description
get disk type	Specifies the defined Action of getting the disk type.
get disk attributes	Specifies the defined Action of querying the attributes of a disk, such as the amount of available free space.
mount disk	Specifies the defined Action of mounting an existing file system to a mounting point.
unmount disk	Specifies the defined Action of unmounting an existing file system from a mounting point.
emulate disk	Specifies the defined Action of emulating an existing disk.
list disks	Specifies the defined Action of listing all disks available on a system.
monitor disks	Specifies the defined Action of monitoring an existing disk for changes.

2.3.5 DNSActionNameVocab-1.0

The `DNSActionNameVocab` is the default MAEC vocabulary for Action names associated with the Domain Name System (DNS), which are captured in MAEC Actions via the `Name` element of the `ActionType` that is defined in CybOX Core, and extended by MAEC's `MalwareActionType` in the MAEC Bundle. Thus, for DNS Action names, the MAEC `DNSActionNameVocab-1.0` SHOULD be used in place of the CybOX `ActionNameVocab` default vocabulary.

The MAEC `DNSActionNameVocab-1.0` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:DNSActionNameEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default DNS Action Names*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#DNSActionNameVocab-1.0*.'

2.3.5.1 DNSActionNameEnum-1.0

The `DNSActionNameEnum` is a non-exhaustive enumeration of the different Action names associated with DNS.

Enumeration Value	Description
send dns query	Specifies the defined Action of sending a DNS query
send reverse dns lookup	Specifies the defined Action of sending a reverse DNS lookup

2.3.6 FileActionNameVocab-1.1

The `FileActionNameVocab` is the default MAEC vocabulary for Action names associated with files, which are captured in MAEC Actions via the `Name` element of the `ActionType` that is defined in CybOX Core, and extended by MAEC's `MalwareActionType` in the MAEC Bundle. Thus, for file Action names, the MAEC `FileActionNameVocab-1.1` SHOULD be used in place of the CybOX `ActionNameVocab` default vocabulary.

The MAEC `FileActionNameVocab-1.1` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:FileActionNameEnum-1.1`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default File Action Names*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is

			‘https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#FileActionNameVocab-1.1.’

2.3.6.1 FileActionNameEnum-1.1

The `FileActionNameEnum` is a non-exhaustive enumeration of the different Action names associated with files.

Enumeration Value	Description
create file	Specifies the defined Action of creating a new file.
delete file	Specifies the defined Action of deleting an existing file.
copy file	Specifies the defined Action of copying an existing file from one location to another.
create file symbolic link	Specifies the defined Action of creating a symbolic link to an existing file.
find file	Specifies the defined Action of searching for an existing file.
get file attributes	Specifies the defined Action of getting the attributes of an existing file.
set file attributes	Specifies the defined Action of setting the file attributes for an existing file.
lock file	Specifies the defined Action of locking an existing file.
unlock file	Specifies the defined Action of unlocking an existing file.
modify file	Specifies the defined Action of modifying an existing file in some manner.
move file	Specifies the defined Action of moving an existing file from one location to another.
open file	Specifies the defined Action of opening an existing file for reading or writing.
read from file	Specifies the defined Action of reading from an existing file.
write to file	Specifies the defined Action of writing to an existing file.
rename file	Specifies the defined Action of renaming an existing file.
create file alternate data stream	Specifies the defined Action of creating an alternate data stream in an existing file.
send control code to file	Specifies the defined Action of sending a control code to a file.
create file mapping	Specifies the defined Action of creating a new file mapping object.
open file mapping	Specifies the defined Action of opening an existing file mapping object.
execute file	Specifies the defined Action of executing an existing file.
hide file	Specifies the defined Action of hiding an existing file.
close file	Specifies the defined Action of closing an existing file that previously opened for reading or writing.

2.3.7 FTPActionNameVocab-1.0

The FTPActionNameVocab is the default MAEC vocabulary for Action names associated with the File Transfer Protocol (FTP), which are captured in MAEC Actions via the Name element of the ActionType that is defined in CybOX Core, and extended by MAEC's MalwareActionType in the MAEC Bundle. Thus, for FTP Action names, the MAEC FTPActionNameVocab-1.0 SHOULD be used in place of the CybOX ActionNameVocab default vocabulary.

The MAEC FTPActionNameVocab-1.0 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:FTPActionNameEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default FTP Action Names*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#FTPActionNameVocab-1.0*.'

2.3.7.1 FTPActionNameEnum-1.0

The FTPActionNameEnum is a non-exhaustive enumeration of the different Action names associated with FTP.

Enumeration Value	Description
connect to ftp server	Specifies the defined Action of connecting to an existing FTP server.
disconnect from ftp server	Specifies the defined Action of disconnecting from an existing FTP server.
send ftp command	Specifies the defined Action of sending a command on an FTP server connection.

2.3.8 GUIActionNameVocab-1.0

The GUIActionNameVocab is the default MAEC vocabulary for Action names associated with Graphical User Interfaces (GUIs), which are captured in MAEC Actions via the Name element of the ActionType that is defined in CybOX Core, and extended by

MAEC's MalwareActionType in the MAEC Bundle. Thus, for GUI Action names, the MAEC GUIActionNameVocab-1.0 SHOULD be used in place of the CybOX ActionNameVocab default vocabulary.

The MAEC GUIActionNameVocab-1.0 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:GUIActionNameEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default GUI Action Names*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#GUIActionNameVocab-1.0*.'

2.3.8.1 GUIActionNameEnum-1.0

The GUIActionNameEnum is a non-exhaustiveenumeration of the different Action names associated with GUIs.

Enumeration Value	Description
create window	Specifies the defined Action of creating a new window.
kill window	Specifies the defined Action of killing an existing window.
create dialog box	Specifies the defined Action of creating a new dialog box.
enumerate windows	Specifies the defined Action of enumerating all open windows
find window	Specifies the defined Action of search for a particular window.
hide window	Specifies the defined Action of hiding an existing window.
show window	Specifies the defined Action of showing an existing window

2.3.9 HookingActionNameVocab-1.1

The HookingActionNameVocab is the default MAEC vocabulary for Action names associated with hooking, which are captured in MAEC Actions via the Name element of the ActionType that is defined in CybOX Core, and extended by MAEC's

MalwareActionType in the MAEC Bundle. Thus, for hooking Action names, the MAEC HookingActionNameVocab-1.1 SHOULD be used in place of the CybOX ActionNameVocab default vocabulary.

The MAEC HookingActionNameVocab-1.1 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:HookingActionNameEnum-1.1; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Hooking Action Names*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#HookingActionNameVocab-1.1*.'

2.3.9.1 HookingActionNameEnum-1.1

The HookingActionNameEnum is a non-exhaustive enumeration of the different Action names associated with hooking.

Enumeration Value	Description
add system call hook	Specifies the defined Action of adding a new system call hook.
add windows hook	Specifies the defined Action of adding a new Windows application-defined hook procedure.
hide hook	Specifies the defined action of hiding an existing hook.

2.3.10 HTTPActionNameVocab-1.0

The HTTPActionNameVocab is the default MAEC vocabulary for Action names associated with the Hypertext Transfer Protocol (HTTP), which are captured in MAEC Actions via the Name element of the ActionType that is defined in CybOX Core, and extended by MAEC's MalwareActionType in the MAEC Bundle. Thus, for HTTP Action names, the MAEC HTTPActionNameVocab-1.0 SHOULD be used in place of the CybOX ActionNameVocab default vocabulary.

The MAEC HTTPActionNameVocab-1.0 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:HTTPActionNameEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default HTTP Action Names*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#HTTPActionNameVocab-1.0*.'

2.3.10.1 HTTPActionNameEnum-1.0

The HTTPActionNameEnum is a non-exhaustive enumeration of the different Action names associated with HTTP.

Enumeration Value	Description
send http get request	Specifies the defined Action of sending an HTTP GET client request to an existing server.
send http head request	Specifies the defined Action of sending an HTTP HEAD client request to an existing server.
send http post request	Specifies the defined Action of sending an HTTP HEAD client request to an existing server.
send http put request	Specifies the defined Action of sending an HTTP PUT client request to an existing server.
send http delete request	Specifies the defined Action of sending an HTTP DELETE client request to an existing server.
send http trace request	Specifies the defined Action of sending an HTTP TRACE client request to an existing server.
send http options request	Specifies the defined Action of sending an HTTP OPTIONS client request to an existing server.
send http connect request	Specifies the defined Action of sending an HTTP CONNECT client request to an existing server.
send http patch request	Specifies the defined Action of sending an HTTP PATCH client request to an existing server.
receive http response	Specifies the defined Action of receiving an HTTP server response for a prior HTTP request.

2.3.11 IPCActionNameVocab-1.0

The IPCActionNameVocab is the default MAEC vocabulary for Action names associated with Inter-process communication (IPC), which are captured in MAEC Actions via the Name element of the ActionType that is defined in CybOX Core, and extended by

MAEC's MalwareActionType in the MAEC Bundle. Thus, for IPC Action names, the MAEC IPCActionNameVocab-1.0 SHOULD be used in place of the CybOX ActionNameVocab default vocabulary.

The MAEC IPCActionNameVocab-1.0 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:IPCActionNameEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default IPC Action Names*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#IPCActionNameVocab-1.0*.'

2.3.11.1 IPCActionNameEnum-1.0

The IPCActionNameEnum is a non-exhaustive enumeration of the different Action names associated with entities related to IPC.

Enumeration Value	Description
create named pipe	Specifies the defined Action of creating a new named pipe.
delete named pipe	Specifies the defined Action of deleting an existing named pipe.
connect to named pipe	Specifies the defined Action of connecting to an existing named pipe.
disconnect from named pipe	Specifies the defined Action of disconnecting from an existing named pipe.
read from named pipe	Specifies the defined Action of reading some data from an existing named pipe.
write to named pipe	Specifies the defined Action of writing some data to an existing named pipe.
create mailslot	Specifies the defined Action of creating a new named mailslot.
read from mailslot	Specifies the defined Action of reading some data from an existing named mailslot.
write to mailslot	Specifies the defined Action of writing some data to an existing named mailslot.

2.3.12 IRCActionNameVocab-1.0

The IRCActionNameVocab is the default MAEC vocabulary for Action names associated with Internet Relay Chat (IRC), which are captured in MAEC Actions via the Name element of the ActionType that is defined in CybOX Core, and extended by MAEC's MalwareActionType in the MAEC Bundle. Thus, for IRC Action names, the MAEC IRCActionNameVocab-1.0 SHOULD be used in place of the CybOX ActionNameVocab default vocabulary.

The MAEC IRCActionNameVocab-1.0 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:IRCActionNameEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default IRC Action Names*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#IRCActionNameVocab-1.0*.'

2.3.12.1 IRCActionNameEnum-1.0

The IRCActionNameEnum is a non-exhaustive enumeration of the different Action names associated with IRC.

Enumeration Value	Description
connect to irc server	Specifies the defined Action of connecting to an existing IRC server.
disconnect from irc server	Specifies the defined Action of disconnecting from an existing IRC server.
set irc nickname	Specifies the defined Action of setting an IRC nickname on an IRC server.
join irc channel	Specifies the defined Action of joining a channel on an IRC server.
leave irc channel	Specifies the defined Action of leaving a channel on an IRC server.
send irc private message	Specifies the defined Action of sending a private message to another user on an IRC server.
receive irc private message	Specifies the defined Action of receiving a private message from another user on an IRC server.

2.3.13 LibraryActionNameVocab-1.1

The `LibraryActionNameVocab` is the default MAEC vocabulary for Action names associated with libraries, which are captured in MAEC Actions via the `Name` element of the `ActionType` that is defined in CybOX Core, and extended by MAEC's `MalwareActionType` in the MAEC Bundle. Thus, for library Action names, the MAEC `LibraryActionNameVocab-1.1` SHOULD be used in place of the CybOX `ActionNameVocab` default vocabulary.

The MAEC `LibraryActionNameVocab-1.1` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:LibraryActionNameEnum-1.1`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Library Action Names*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#LibraryActionNameVocab-1.1*.'

2.3.13.1 LibraryActionNameEnum-1.1

The `LibraryActionNameEnum` is a non-exhaustive enumeration of the different Action names associated with libraries.

Enumeration Value	Description
enumerate libraries	Specifies the defined Action of enumerating the libraries used by a process.
free library	Specifies the defined Action of freeing a library previously loaded into the address space of the calling process.
load library	Specifies the defined Action of loading a library into the address space of the calling process.
get function address	Specifies the defined Action of getting the address of an exported function or variable from a library.
call library function	Specifies the defined action of calling a function exported by a library.

2.3.14 NetworkActionNameVocab-1.1

The `NetworkActionNameVocab` is the default MAEC vocabulary for Action names associated with networking, which are captured in MAEC Actions via the `Name` element of the `ActionType` that is defined in CybOX Core, and extended by MAEC's `MalwareActionType` in the MAEC Bundle. Thus, for network Action names, the MAEC `NetworkActionNameVocab-1.1` SHOULD be used in place of the CybOX `ActionNameVocab` default vocabulary.

The MAEC `NetworkActionNameVocab-1.1` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:NetworkActionNameEnum-1.1`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Network Action Names*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#NetworkActionNameVocab-1.1*.'

2.3.14.1 NetworkActionNameEnum-1.1

The `NetworkActionNameEnum` is a non-exhaustive enumeration of the different Action names associated with networking.

Enumeration Value	Description
open port	Specifies the defined Action of opening a network port.
close port	Specifies the defined Action of closing a network port.
connect to ip	Specifies the defined Action of connecting to an IP address.
disconnect from ip	Specifies the defined Action of disconnecting from a previously established connection with an IP address.
connect to url	Specifies the defined Action of connecting to a URL.
connect to socket address	Specifies the defined Action of connecting to a socket address, consisting of an IP address and port number.
download file	Specifies the defined Action of downloading a file from a remote location.
upload file	Specifies the defined Action of uploading a file to a remote location.
listen on port	Specifies the defined Action of listening on a specific port.

send email message	Specifies the defined Action of sending an email message.
send icmp request	Specifies the defined Action of sending an ICMP request.
send network packet	Specifies the defined action of sending a packet on a network.
receive network packet	Specifies the defined action of receiving a packet on a network.

2.3.15 NetworkShareActionNameVocab-1.0

The NetworkShareActionNameVocab is the default MAEC vocabulary for Action names associated with Windows network shares, which are captured in MAEC Actions via the Name element of the ActionType that is defined in CybOX Core, and extended by MAEC's MalwareActionType in the MAEC Bundle. Thus, for Windows network share Action names, the MAEC NetworkShareActionNameVocab-1.0 SHOULD be used in place of the CybOX ActionNameVocab default vocabulary.

The MAEC NetworkShareActionNameVocab-1.0 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:NetworkShareActionNameEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Network Share Action Names*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#NetworkShareActionNameVocab-1.0*.'

2.3.15.1 NetworkShareActionNameEnum-1.0

The NetworkShareActionNameEnum is a non-exhaustive enumeration of the different Action names associated with Windows network shares.

Enumeration Value	Description
add connection to network share	Specifies the defined Action of adding a connection to an existing network share.
add network share	Specifies the defined Action of adding a new network share on a server.

delete network share	Specifies the defined Action of deleting an existing network share on a server.
connect to network share	Specifies the defined Action of connecting to an existing network share.
disconnect from network share	Specifies the defined Action of disconnecting from an existing network share.
enumerate network shares	Specifies the defined Action of enumerating the available shared resources on a server.

2.3.16 ProcessActionNameVocab-1.0

The `ProcessActionNameVocab` is the default MAEC vocabulary for Action names associated with processes, which are captured in MAEC Actions via the `Name` element of the `ActionType` that is defined in CybOX Core, and extended by MAEC's `MalwareActionType` in the MAEC Bundle. Thus, for process Action names, the MAEC `ProcessActionNameVocab-1.0` SHOULD be used in place of the CybOX `ActionNameVocab` default vocabulary.

The MAEC `ProcessActionNameVocab-1.0` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:ProcessActionNameEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Process Action Names*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#ProcessActionNameVocab-1.0*.'

2.3.16.1 ProcessActionNameEnum-1.0

The `ProcessActionNameEnum` is a non-exhaustive enumeration of the different Action names associated with processes.

Enumeration Value	Description
create process	Specifies the defined Action of creating a new process.
kill process	Specifies the defined Action of killing an existing process.
create process as user	Specifies the defined Action of creating a new process in the security context of a specified user.

enumerate processes	Specifies the defined Action of enumerating all of the running processes on a system.
open process	Specifies the defined Action of opening an existing process.
flush process instruction cache	Specifies the defined Action of flushing the instruction cache of an existing process.
get process current directory	Specifies the defined Action of getting the current directory of an existing process.
set process current directory	Specifies the defined Action of setting the current directory of an existing process.
get process environment variable	Specifies the defined Action of getting an environment variable used by an existing process.
set process environment variable	Specifies the defined Action of setting an environment variable used by an existing process.
sleep process	Specifies the defined Action of sleeping an existing process for some period of time.
get process startupinfo	Specifies the defined Action of getting the STARTUPINFO struct associated with an existing process.

2.3.17 ProcessMemoryActionNameVocab-1.0

The `ProcessMemoryActionNameVocab` is the default MAEC vocabulary for Action names associated with process memory, which are captured in MAEC Actions via the `Name` element of the `ActionType` that is defined in CybOX Core, and extended by MAEC's `MalwareActionType` in the MAEC Bundle. Thus, for process memory Action names, the MAEC `ProcessMemoryActionNameVocab-1.0` SHOULD be used in place of the CybOX `ActionNameVocab` default vocabulary.

The MAEC `ProcessMemoryActionNameVocab-1.0` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:ProcessMemoryActionNameEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Process Memory Action Names*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#ProcessMemoryActionNameVocab-1.0*.'

2.3.17.1 ProcessMemoryActionNameEnum-1.0

The `ProcessMemoryActionNameEnum` is a non-exhaustive enumeration of the different Action names associated with process memory.

Enumeration Value	Description
allocate process virtual memory	Specifies the defined Action of allocating some virtual memory region in an existing process.
free process virtual memory	Specifies the defined Action of freeing some virtual memory region from an existing process.
modify process virtual memory protection	Specifies the defined Action of modifying the protection on a memory region in the virtual address space of an existing process.
read from process memory	Specifies the defined Action of reading from a memory region of an existing process
write to process memory	Specifies the defined Action of writing to a memory region of an existing process.
map file into process	Specifies the defined Action of mapping an existing file into the address space of the calling process.
upmap file from process	Specifies the defined Action of unmapping an existing file from the address space of the calling process.
map library into process	Specifies the defined Action of mapping a library into the address space of the calling process.

2.3.18 ProcessThreadActionNameVocab-1.0

The `ProcessThreadActionNameVocab` is the default MAEC vocabulary for Action names associated with process threads, which are captured in MAEC Actions via the `Name` element of the `ActionType` that is defined in CybOX Core, and extended by MAEC's `MalwareActionType` in the MAEC Bundle. Thus, for process thread Action names, the MAEC `ProcessThreadActionNameVocab-1.0` SHOULD be used in place of the CybOX `ActionNameVocab` default vocabulary.

The MAEC `ProcessThreadActionNameVocab-1.0` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:ProcessThreadActionNameEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Process Thread Action Names*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#ProcessThreadActionNameVocab-1.0*.'

2.3.18.1 ProcessThreadActionNameEnum-1.0

The `ProcessThreadActionNameEnum` is a non-exhaustive enumeration of the different Action names associated with process threads.

Enumeration Value	Description
create thread	Specifies the defined Action of creating a new thread in the virtual address space of the calling process.
kill thread	Specifies the defined Action of killing a thread existing in the virtual address space of the calling process.
create remote thread in process	Specifies the defined Action of creating a thread that runs in the virtual address space of another existing process.
enumerate threads	Specifies the defined Action of enumerating all threads in the calling process.
get thread username	Specifies the defined Action of getting the name or ID of the user associated with an existing thread.
impersonate process	Specifies the defined Action of a thread in the calling process impersonating the security context of another existing process.
revert thread to self	Specifies the defined Action of reverting an existing thread to its own security context.
get thread context	Specifies the defined Action of getting the context structure (containing processor-specific register data) of an existing thread.
set thread context	Specifies the defined Action of setting the context structure (containing processor-specific register data) for an existing thread.
queue apc in thread	Specifies the defined Action of queuing a new Asynchronized Procedure Call (APC) in the context of an existing thread.

2.3.19 RegistryActionNameVocab-1.0

The `RegistryActionNameVocab` is the default MAEC vocabulary for Action names associated with the Windows registry, which are captured in MAEC Actions via the `Name` element of the `ActionType` that is defined in CybOX Core, and extended by MAEC's `MalwareActionType` in the MAEC Bundle. Thus, for Windows registry Action names, the MAEC `RegistryActionNameVocab-1.0` SHOULD be used in place of the CybOX `ActionNameVocab` default vocabulary.

The MAEC `RegistryActionNameVocab-1.0` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:RegistryActionNameEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Registry Action Names*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#RegistryActionNameVocab-1.0*.'

2.3.19.1 RegistryActionNameEnum-1.0

The RegistryActionNameEnum is a non-exhaustive enumeration of the different Action names associated with the Windows registry.

Enumeration Value	Description
create registry key	Specifies the defined Action of creating a new registry key.
delete registry key	Specifies the defined Action of deleting an existing registry key.
open registry key	Specifies the defined Action of opening an existing registry key.
close registry key	Specifies the defined Action of closing a handle to an existing registry key.
create registry key value	Specifies the defined Action of creating a new named value under an existing registry key.
delete registry key value	Specifies the defined Action of deleting an existing named value under an existing registry key.
enumerate registry key subkeys	Specifies the defined Action of enumerating the registry key subkeys under an existing registry key.
enumerate registry key values	Specifies the defined Action of enumerating the named values under an existing registry key.
get registry key attributes	Specifies the defined Action of getting the attributes of an existing registry key.
read registry key value	Specifies the defined Action of reading an existing named value of an existing registry key.
modify registry key value	Specifies the defined Action of modifying an existing named value of an existing registry key.
modify registry key	Specifies the defined Action of modifying an existing registry key.
monitor registry key	Specifies the defined Action of monitoring an existing registry key for changes.

2.3.20 ServiceActionNameVocab-1.1

The ServiceActionNameVocab is the default MAEC vocabulary for Action names associated with services and daemons, which are captured in MAEC Actions via the Name element of the ActionType that is defined in CybOX Core, and extended by MAEC's

MalwareActionType in the MAEC Bundle. Thus, for service Action names, the MAEC ServiceActionNameVocab-1.1 SHOULD be used in place of the CybOX ActionNameVocab default vocabulary.

The MAEC ServiceActionNameVocab-1.1 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:ServiceActionNameEnum-1.1; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Service Action Names*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#ServiceActionNameVocab-1.1*.'

2.3.20.1 ServiceActionNameEnum-1.1

The ServiceActionNameEnum is a non-exhaustive enumeration of the different Action names associated with services or daemons.

Enumeration Value	Description
create service	Specifies the defined Action of creating a new service.
delete service	Specifies the defined Action of deleting an existing service.
start service	Specifies the defined Action of starting an existing service.
stop service	Specifies the defined Action of stopping an existing service.
enumerate services	Specifies the defined Action of enumerating a specific set of services on a system.
modify service configuration	Specifies the defined Action of modifying the configuration parameters of an existing service.
open service	Specifies the defined Action of opening an existing service.
send control code to service	Specifies the defined Action of sending a control code to an existing service.

2.3.21 SocketActionNameVocab-1.0

The SocketActionNameVocab is the default MAEC vocabulary for Action names associated with network sockets, which are captured in MAEC Actions via the Name element of the ActionType that is defined in CybOX Core, and extended by MAEC's MalwareActionType in the MAEC Bundle. Thus, for network socket Action names, the MAEC SocketActionNameVocab-1.0 SHOULD be used in place of the CybOX ActionNameVocab default vocabulary.

The MAEC SocketActionNameVocab-1.0 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:SocketActionNameEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Socket Action Names*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#SocketActionNameVocab-1.0*.'

2.3.21.1 SocketActionNameEnum-1.0

The SocketActionNameEnum is a non-exhaustive enumeration of the different Action names associated with network sockets.

Enumeration Value	Description
accept socket connection	Specifies the defined Action of accepting a socket connection.
bind address to socket	Specifies the defined Action of binding a socket address to a socket.
create socket	Specifies the defined Action of creating a new socket.
close socket	Specifies the defined Action of closing an existing socket.
connect to socket	Specifies the defined Action of connecting to an existing socket.
disconnect from socket	Specifies the defined Action of disconnecting from an existing socket.
listen on socket	Specifies the defined Action of listening on an existing socket.
send data on socket	Specifies the defined Action of sending data on an existing, connected socket.
receive data on socket	Specifies the defined Action of receiving data on an existing socket.

send data to address on socket	Specifies the defined Action of sending data to a specified IP address on an existing, unconnected socket.
get host by address	Specifies the defined Action of getting information on a host from a local or remote host database by its IP address.
get host by name	Specifies the defined Action of getting information on a host from a local or remote host database by its name.

2.3.22 SynchronizationActionNameVocab-1.0

The `SynchronizationActionNameVocab` is the default MAEC vocabulary for Action names associated with process and thread synchronization-related entities, which are captured in MAEC Actions via the `Name` element of the `ActionType` that is defined in CybOX Core, and extended by MAEC's `MalwareActionType` in the MAEC Bundle. Thus, for process and thread synchronization-related Action names, the MAEC `SynchronizationActionNameVocab-1.0` SHOULD be used in place of the CybOX `ActionNameVocab` default vocabulary.

The MAEC SynchronizationActionNameVocab-1.0 extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:SynchronizationActionNameEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Synchronization Action Names*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#SynchronizationActionNameVocab-1.0*.'

2.3.22.1 SynchronizationActionNameEnum-1.0

The `SynchronizationActionNameEnum` is a non-exhaustive enumeration of the different Action names associated with process and thread synchronization-related entities.

Enumeration Value	Description

create mutex	Specifies the defined Action of creating a new named mutex.
delete mutex	Specifies the defined Action of deleting an existing named mutex.
open mutex	Specifies the defined Action of opening an existing named mutex.
release mutex	Specifies the defined Action of releasing ownership of an existing named mutex.
create semaphore	Specifies the defined Action of creating a new named semaphore.
delete semaphore	Specifies the defined Action of deleting an existing named semaphore.
open semaphore	Specifies the defined Action of opening an existing named semaphore.
release semaphore	Specifies the defined Action of releasing ownership of an existing named semaphore.
create event	Specifies the defined Action of creating a new named event object.
delete event	Specifies the defined Action of deleting an existing named event object.
open event	Specifies the defined Action of opening an existing named event object.
reset event	Specifies the defined Action of resetting an existing named event object to the non-signaled state.
create critical section	Specifies the defined Action of creating a new critical section.
delete critical section	Specifies the defined Action of deleting an existing critical section object.
open critical section	Specifies the defined Action of opening an existing critical section object.
release critical section	Specifies the defined Action of releasing an existing critical section object.

2.3.23 SystemActionNameVocab-1.0

The `SystemActionNameVocab` is the default MAEC vocabulary for Action names associated with system-related entities, which are captured in MAEC Actions via the `Name` element of the `ActionType` that is defined in CybOX Core, and extended by MAEC's `MalwareActionType` in the MAEC Bundle. Thus, for system-related field Action names, the MAEC `SystemActionNameVocab-1.0` SHOULD be used in place of the CybOX `ActionNameVocab` default vocabulary.

The MAEC `SystemActionNameVocab-1.0` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:SystemActionNameEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default System Action Names*.'

vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#SystemActionNameVocab-1.0*.'

2.3.23.1 SystemActionNameEnum-1.0

The `SystemActionNameEnum` is a non-exhaustive enumeration of the different Action names associated with system-related entities.

Enumeration Value	Description
add scheduled task	Specifies the defined Action of adding a scheduled task to a system.
shutdown system	Specifies the defined Action of shutting down a system.
sleep system	Specifies the defined Action of sleeping a system for some period of time.
get elapsed system up time	Specifies the defined Action of getting the elapsed up-time for a system.
get netbios name	Specifies the defined Action of getting the NetBIOS name of a system.
set netbios name	Specifies the defined Action of setting the NetBIOS name of a system.
get system host name	Specifies the defined Action of getting the host name of a system.
set system host name	Specifies the defined Action of setting the system host name of a system.
get system time	Specifies the defined Action of getting the system time of a system, represented in Coordinated Universal Time (UTC).
set system time	Specifies the defined Action of setting the system time for a system, represented in Coordinated Universal Time (UTC).
get system local time	Specifies the defined Action of getting the local time of a system.
set system local time	Specifies the defined Action of setting the local time of a system.
get username	Specifies the defined Action of getting the username of the currently logged in user of a system.
enumerate system handles	Specifies the defined Action of enumerating all open handles on a system.
get system global flags	Specifies the defined Action of getting the enabled global flags on a system.
set system global flags	Specifies the defined Action of setting system global flags on a system.
get windows directory	Specifies the defined Action of getting the path to the Windows installation directory on a system.
get windows system directory	Specifies the defined Action of getting the path to the Windows \System directory on a system.
get windows temporary files directory	Specifies the defined Action of getting the path to the Windows Temporary Files Directory on a System.

2.3.24 UserActionNameVocab-1.1

The UserActionNameVocab is the default MAEC vocabulary for Action names associated with users, which are captured in MAEC Actions via the Name element of the ActionType that is defined in CybOX Core, and extended by MAEC's MalwareActionType in the MAEC Bundle. Thus, for user Action names, the MAEC UserActionNameVocab-1.1 SHOULD be used in place of the CybOX ActionNameVocab default vocabulary.

The MAEC UserActionNameVocab-1.1 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:UserActionNameEnum-1.1; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default User Action Names*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#UserActionNameVocab-1.1*.'

2.3.24.1 UserActionNameEnum-1.1

The UserActionNameEnum is a non-exhaustive enumeration of the different Action names associated with users.

Enumeration Value	Description
add user	Specifies the defined Action of adding a new user.
delete user	Specifies the defined Action of deleting an existing user.
enumerate users	Specifies the defined Action of enumerating all users.
get user attributes	Specifies the defined Action of getting the attributes of an existing user.
logon as user	Specifies the defined Action of logging on as a specific user.
change password	Specifies the defined Action of changing an existing user's password.
add user to group	Specifies the defined Action of adding an existing user to an existing group.
remove user from group	Specifies the defined Action of removing an existing user from existing group.
invoke user privilege	Specifies the defined Action of invoking a privilege given to an existing user.

2.4 Candidate-Indicator-Related Default Vocabularies

The default vocabularies in this section are related to MAEC Candidate Indicators.

2.4.1 ImportanceTypeVocab-1.0

ImportanceTypeVocab is the default MAEC vocabulary for relative importance measures in a MAEC Bundle, which are captured via the Importance field in Candidate Indicators of type CandidateIndicatorType, defined in the MAEC Bundle schema.

The MAEC ImportanceTypeVocab-1.0 type extends ControlledVocabularyStringType defined in CybOX Common. Thus, Importance fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:ImportanceTypeEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Importance Types*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#ImportanceTypeVocab-1.0*.'

2.4.1.1 ImportanceTypeEnum-1.0

The ImportanceTypeEnum is a non-exhaustive enumeration of relative importance measures.

Enumeration Value	Description
high	Specifies that the field is of relative high importance.
medium	Specifies that the field is of relative medium importance.
low	Specifies that the field is of relative low importance.
informational	Specifies that the field is only informational in its importance.
numeric	Specifies that the field has a numeric importance value, which is defined in another attribute or element.
unknown	Specifies that the relative importance for the field is unknown.

2.4.2 MalwareEntityTypeVocab-1.0

The MalwareEntityTypeVocab is the default MAEC vocabulary for malware entity types in a MAEC Bundle, which are captured in Candidate Indicators via the Type field, a child of the Malware_Entity field of the CandidateIndicatorType, defined in the MAEC_Bundle schema.

The MAEC MalwareEntityTypeVocab-1.0 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Type fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:MalwareEntityTypeEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Malware Entity Types*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#MalwareEntityTypeVocab-1.0*.'

2.4.2.1 MalwareEntityTypeEnum-1.0

The MalwareEntityTypeEnum is a non-exhaustive enumeration of the different types of entities that a malware indicator or signature may be written against.

Enumeration Value	Description
instance	Specifies that the particular malware entity being referred to is a single malware instance.
family	Specifies that the particular malware entity being referred to is a single malware family.
class	Specifies that the particular malware entity being referred to is a single class of malware.

2.5 Capability-Related Default Vocabularies

The default vocabularies in this section are related to malware Capabilities in a MAEC Bundle.

2.5.1 CapabilityObjectiveRelationshipTypeVocab-1.0

The `CapabilityObjectiveRelationshipTypeVocab` is the default MAEC vocabulary for relationships between Strategic and Tactical Objectives associated with a malware Capability, which are captured in `Relationship` fields via the child `Relationship_Type` field of type `CapabilityObjectiveRelationshipType`, defined in the MAEC Bundle schema.

The MAEC `CapabilityObjectiveRelationshipTypeVocab-1.0` type extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Relationship_Type` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:CapabilityObjectiveRelationshipTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Malware Capability Objective Relationship Types*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#CapabilityObjectiveRelationshipTypeVocab-1.0*.'

2.5.1.1 CapabilityObjectiveRelationshipTypeEnum-1.0

The `CapabilityObjectiveRelationshipEnum` is a non-exhaustive enumeration of relationships between malware Capability Strategic and Tactical Objectives.

Enumeration Value	Description
child of	Indicates that the Objective is a child of the Objective being referenced.
parent of	Indicates that the Objective is a parent of the Objective being referenced.
incorporates	Indicates that the Objective incorporates the Objective being referenced in a supporting or enabling role.
incorporated by	Indicates that the Objective is incorporated in a supporting or enabling role by the Objective being referenced.

2.5.2 CommonCapabilityPropertiesVocab-1.0

The CommonCapabilityPropertiesVocab is the default MAEC vocabulary for properties common to many Capabilities and their child Objectives. The names of these properties are captured in the Name field of the Property field that uses the CapabilityPropertyType as its base type. The Property field is found on the CapabilityType (which is used to define a Capability) and on the CapabilityObjectiveType (which is used to define a Strategic Objective or a Tactical Objective). All aforementioned types are defined in the MAEC Bundle schema.

The MAEC CommonCapabilityPropertiesVocab-1.0 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:CommonCapabilityPropertiesTypeEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Common Capability and Objective Properties*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#CommonCapabilityPropertiesVocab-1.0*.'

2.5.2.1 CommonCapabilityPropertiesEnum-1.0

The CommonCapabilityPropertiesEnum is a non-exhaustive enumeration of properties common to many Capability, Strategic Objective, and Tactical Objective properties.

Enumeration Value	Description
encryption algorithm	Refers to the name of the encryption algorithm used in the Capability or Objective.
protocol used	Refers to the name of the network protocol used in the Capability or Strategic or Tactical Objective. It is recommended that protocols be specified by their acronym or abbreviated name, e.g. "IRC", "HTTP".

2.5.3 MalwareCapabilityVocab-1.0

The `MalwareCapabilityVocab` is the default MAEC vocabulary for names of malware Capabilities, which are captured via the `name` field of the `CapabilityType`, defined in the MAEC Bundle schema.

The MAEC `MalwareCapabilityVocab-1.0` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:MalwareCapabilityTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Malware Capabilities*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#MalwareCapabilityVocab-1.0*.'

2.5.3.1 MalwareCapabilityEnum-1.0

The `MalwareCapabilityEnum` is a non-exhaustive enumeration of Capability names.

Enumeration Value	Description
command and control	Indicates that the malware instance is able to receive and execute remotely submitted commands.
remote machine manipulation	Indicates that the malware instance is able to manipulate or access other remote machines.
privilege escalation	Indicates that the malware instance is able to elevate the privileges under which it executes.
data theft	Indicates that the malware instance is able to steal data from the system on which it executes. This includes data stored in some form, e.g. in a file, as well as data that may be entered into some application such as a web-browser.
spying	Indicates that the malware instance is able to capture information from a system related to user or system activity (e.g., from a system's peripheral devices).
secondary operation	Indicates that the malware instance is able to achieve secondary objectives in conjunction with or after achieving its primary objectives.

anti-detection	Indicates that the malware instance is able to prevent itself and its components from being detected on a system.
anti-code analysis	Indicates that the malware instance is able to prevent code analysis or make it more difficult.
infection/propagation	Indicates that the malware instance is able to propagate through the infection of a machine or is able to infect a file after executing on a system. The malware instance may infect actively (e.g., gain access to a machine directly) or passively (e.g., send malicious email). This Capability does not encompass any aspects of the initial infection that is done independently of the malware instance itself.
anti-behavioral analysis	Indicates that the malware instance is able to prevent behavioral analysis or make it more difficult.
integrity violation	Indicates that the malware instance is able to compromise the integrity of a system.
data exfiltration	Indicates that the malware instance is able to exfiltrate stolen data or perform tasks related to the exfiltration of stolen data.
probing	Indicates that the malware instance is able to probe its host system or network environment; most often this is done to support other Capabilities and their Objectives.
anti-removal	Indicates that the malware instance is able to prevent itself and its components from being removed from a system.
security degradation	Indicates that the malware instance is able to bypass or disable security features and/or controls.
availability violation	Indicates that the malware instance is able to compromise the availability of a system or some aspect of the system.
destruction	Indicates that the malware instance is able to destroy some aspect of a system.
fraud	Indicates that the malware instance is able to defraud a user or a system.
persistence	Indicates that the malware instance is able to persist and remain on a system regardless of system events.
machine access/control	Indicates that the malware instance is able to provide the means to access or control the machine on which it is resident.

2.5.4 MalwareLabelVocab-1.0

The `MalwareLabelVocab` is the default MAEC vocabulary for common labels associated with Malware Subjects, which are captured in a Malware Subject via the `Label` field of the `MalwareSubjectType`, defined in the MAEC Package schema.

The MAEC `MalwareLabelVocab-1.0` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Label` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:MalwareLabelEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Malware Labels*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#MalwareLabelsVocab-1.0*.'

2.5.4.1 MalwareLabelEnum-1.0

The `MalwareLabelEnum` is a non-exhaustive enumeration of common malware labels.

Enumeration Value	Description
adware	Specifies any software that is funded by advertising. Some adware may install itself in such a manner as to become difficult to remove, hiding components and disabling removal techniques. Adware may also gather sensitive user information from a system.
appender	Specifies a file-infecting virus that places its code at the end of the files it infects, adjusting the file's entry point to cause its code to be executed before that of the original file.
backdoor	Specifies a piece of software which, once running on a system, opens a communication vector to the outside so that the computer can be accessed remotely by an attacker.
boot sector virus	Specifies a virus that infects the master boot record of a storage device.
bot	Specifies a program which resides on an infected system, communicating with and forming part of a botnet. The bot may be implanted by a worm or trojan, which opens a backdoor. The bot then monitors the backdoor for further instructions.
clicker	Specifies a trojan that makes a system visit a specific Web page, often very frequently and usually with the aim of increasing the traffic recorded by the site and thus increasing revenue from advertising. Clickers may also be used to carry out DDoS attacks.
companion virus	Specifies a virus that takes the place of a particular file on a system instead of injecting code into it.
cavity filler	Specifies a type of file-infecting virus which seeks out unused space within the files it infects, inserting its code into these gaps to avoid changing the size of the file and thus not alerting integrity-checking software to its presence.
data diddler	Specifies a type of malware that makes small, random changes to data, such as data in a spreadsheet, to render the data contained in a document inaccurate and in some cases worthless.

downloader	Specifies a small trojan file programmed to download and execute other files, usually more complex malware.
dropper file	Specifies a type of Trojan that deposits an enclosed payload onto a destination host computer by loading itself into memory, extracting the malicious payload, and then writing it to the file system.
file infector virus	Specifies a virus that infects a system by inserting itself somewhere in existing files; this is the "classic" form of virus.
fork bomb	Specifies a very simple form of malware, a type of rabbit which simply launches more copies of itself. Once a fork bomb is executed, it will attempt to run several identical processes, which will do the same, the number growing exponentially until the system resources are overwhelmed by the number of identical processes running, which may in some cases bring the system down and cause a denial of service.
greyware	Specifies software that, while not definitely malicious, has a suspicious or potentially unwanted aspect.
implant	Specifies code inserted into an existing program using a code patcher or other tool.
infector	Specifies a function of malware that alters target files for the purpose of persisting and hiding the injected malware.
keylogger	Specifies a type of program implanted on a system to monitor the keys pressed and thus record any sensitive data, such as passwords, entered by the user.
kleptographic worm	Specifies a worm that encrypts information assets on compromised systems so they can only be decrypted by the worm's author, also known as information-stealing worm.
macro virus	Specifies a virus that uses a macro language, for example in Microsoft Office documents.
malcode	Short for malicious code, also known as malware.
mass-mailer	Specifies fies a worm that uses email to propagate across the internet.
metamorphic virus	Specifies a virus that changes its own code with each infection.
mid-infector	Specifies a type of file-infecting virus which places its code in the middle of files it infects. It may move a section of the original code to the end of the file, or simply push the code aside to make space for its own code.
mobile code	Specifies (1) Code received from remote, possibly untrusted systems, but executed on a local system. (2) Software transferred between systems (e.g across a network) and executed on a local system without explicit installation or execution by the recipient.
multipartite virus	Specifies malware that infects boot records, boot sectors, and files.
password stealer	Specifies a type of trojan designed to steal passwords, personal data and details, or other sensitive information from the infected system.
polymorphic virus	Specifies a type of virus that encrypts its code differently with each infection, or generation of infections.

premium dialer/smser	Specifies a piece of malware whose primary aim is to dial or send SMS messages to premium rate numbers.
prepender	Specifies a file-infecting virus which inserts code at the beginning of the files it infects.
ransomware	Specifies a type of malware that encrypts files on a victim's system, demanding payment of ransom in return for the access codes required to unlock files.
rat	Specifies a remote access trojan or RAT, which is a trojan horse capable of controlling a machine through commands issue by a remote attacker.
rogue anti-malware	Specifies a fake security product that demands money to clean phony infections.
rootkit	Generally refers to a method of hiding files or processes from normal methods of monitoring, and is often used by malware to conceal its presence and activities. Originally, the term applied to UNIX-based operating systems - a root kit was a collection of tools to enable a user to obtain root (administrator-level) access to a system and conceal any changes they might make. Such tools often included trojanized versions of standard monitoring software which would hide the root kit operators' activities. More recently the term has generally been applied to malware using stealth techniques. Rootkits can operate at a number of levels, from the application level - simply replacing or adjusting the settings of system software to prevent the display of certain information - through hooking certain functions or inserting modules or drivers into the operating system kernel, to the deeper level of firmware or virtualization rook kits, which are activated before the operating system and thus even harder to detect while the system is running.
shellcode	Specifies (1) A small piece of code that activates a command-line interface to a system that can be used to disable security measures, open a backdoor, or download further malicious code. (2) A small piece of code that opens a system up for exploitation, sometimes by not necessarily involving a command-line shell.
spaghetti packer	A packer that obfuscates programs by emitting "spaghetti" code with a complex and tangled control structure.
spyware	Specifies software that gathers information and passes it to a third-party without adequate permission from the owner of the data. It may also be used in a wider sense, to include software that makes changes to a system or any of its component software, or which makes use of system resources without the full understanding and consent of the system owner.
trojan horse	Specifies a piece of malicious code disguised as something inert or benign.
variant	Refers to the fact that types of malware can be subdivided into a number of families, or groups sharing many similarities, generally based on the same blocks of code and sharing similar behaviours. Within a family, a variant signifies a single individual item that is uniquely different from other members of the same family.
virus	Specifies (1) A self-replicating malicious program that requires human interaction to replicate. (2) A self-

	replicating program that runs and spreads by modifying other programs or files.
wabbit	Specifies a form of self-replicating malware that makes copies of itself on the local system. Unlike worms, wabbits do not attempt to spread across networks.
web bug	Specifies a piece of code, generally a small file such as a tiny, transparent GIF image, which is used to track data on those viewing the page or mail in which it is hidden.
wiper	Specifies a piece of malware whose primary aim is to delete files or entire disks on a machine.
worm	Specifies (1) A self-replicating malicious program that replicates using a network and does not require human interaction. (2) A self-replicating, self-propagating, self-contained program that uses networking mechanisms to spread itself.
zip bomb	Specifies a file compressed into some archive format and that expands to an enormous size when uncompressed, often by looping over the extraction code until the system's resources are exhausted.

2.6 Default Vocabularies for Specific Capabilities

The default vocabularies in this section are related to the properties, Strategic Objectives, and Tactical Objectives associated with specific MAEC Capabilities.

2.6.1 AntiBehavioralAnalysisPropertiesVocab-1.0

The `AntiBehavioralAnalysisPropertiesVocab` is the default MAEC vocabulary for properties of the anti-behavioral analysis Capability and its child Objectives. The names of these properties are captured in the `Name` field, a child of the `Property` field of type `CapabilityPropertyType`. The `Property` field is found on the `CapabilityType` (which is used to define a Capability) and on the `CapabilityObjectiveType` (which is used to define a Strategic Objective or a Tactical Objective). All aforementioned types are defined in the MAEC Bundle schema.

The MAEC `AntiBehavioralAnalysisPropertiesVocab-1.0` type extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:AntiBehavioralAnalysisPropertiesTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description

vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Anti-Behavioral Analysis Capability and Objective Properties*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#AntiBehavioralAnalysisPropertiesVocab-1.0*.'

2.6.1.1 AntiBehavioralAnalysisPropertiesEnum-1.0

The `AntiBehavioralAnalysisPropertiesEnum` is a non-exhaustive enumeration of anti-behavioral analysis Capability, Strategic Objective, and Tactical Objective properties.

Enumeration Value	Description
targeted vm	Refers to the name of a virtual machine (VM) targeted by the anti-behavioral analysis Capability or one of its child Strategic or Tactical Objectives.
targeted sandbox	Refers to the name of a sandbox targeted by the anti-behavioral analysis Capability or one of its child Strategic or Tactical Objectives.

2.6.2 AntiBehavioralAnalysisStrategicObjectivesVocab-1.0

The `AntiBehavioralAnalysisStrategicObjectivesVocab` is the default MAEC vocabulary for Strategic Objectives of the anti-behavioral analysis Capability. The names of these Strategic Objectives are captured in the `Name` field, a child of the `Strategic_Objective` field of type `CapabilityObjectiveType` (defined in the MAEC Bundle schema).

The MAEC `AntiBehavioralAnalysisStrategicObjectivesVocab-1.0` type extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:AntiBehavioralAnalysisStrategicObjectivesTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Anti-Behavioral Analysis Capability Strategic Objectives*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is

			‘*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#AntiBehavioralAnalysisStrategicObjectivesVocab-1.0*.’

2.6.2.1 AntiBehavioralAnalysisStrategicObjectivesEnum-1.0

The `AntiBehavioralAnalysisStrategicObjectivesEnum` is a non-exhaustive enumeration of Strategic Objectives of the anti-behavioral analysis Capability.

Enumeration Value	Description
anti-vm	Indicates that the malware instance is able to prevent virtual machine (VM) based behavioral analysis or make it more difficult.
anti-sandbox	Indicates that the malware instance is able to prevent sandbox-based behavioral analysis or make it more difficult.

2.6.3 AntiBehavioralAnalysisTacticalObjectivesVocab-1.0

The `AntiBehavioralAnalysisTacticalObjectivesVocab` is the default MAEC vocabulary for Tactical Objectives of the anti-behavioral analysis Capability. The names of these Tactical Objectives are captured in the `Name` field, a child of the `Tactical_Objective` field of type `CapabilityObjectiveType` (defined in the MAEC Bundle schema).

The MAEC `AntiBehavioralAnalysisTacticalObjectivesVocab-1.0` type extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:AntiBehavioralAnalysisTacticalObjectivesTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is ‘*MAEC Default Anti-Behavioral Analysis Capability Tactical Objectives*.’
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is ‘*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#AntiBehavioralAnalysisTacticalObjectivesVocab-1.0*.’

2.6.3.1 AntiBehavioralAnalysisTacticalObjectivesEnum-1.0

The AntiBehavioralAnalysisTacticalObjectivesEnum is a non-exhaustive enumeration of Tactical Objectives of the anti-behavioral analysis Capability.

Enumeration Value	Description
detect vm environment	Indicates that the malware instance is able to detect whether it is being executed in a virtual machine (VM).
overload sandbox	Indicates that the malware instance is able to overload a sandbox (e.g., by generating a flood of meaningless behavioral data).
prevent execution in sandbox	Indicates that the malware instance is able to prevent its execution in a sandbox.
detect sandbox environment	Indicates that the malware instance is able to detect whether it is being executed in a sandbox environment.
prevent execution in vm	Indicates that the malware instance is able to prevent its execution in a virtual machine (VM).

2.6.4 AntiCodeAnalysisStrategicObjectiveVocab-1.0

The AntiCodeAnalysisStrategicObjectivesVocab is the default MAEC vocabulary for Strategic Objectives of the anti-code analysis Capability. The names of these Strategic Objectives are captured in the Name field, a child of the Strategic_Objective field of type CapabilityObjectiveType (defined in the MAEC Bundle schema).

The MAEC AntiCodeAnalysisStrategicObjectivesVocab-1.0 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:AntiCodeAnalysisStrategicObjectivesTypeEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Anti-Code Analysis Capability Strategic Objectives*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#AntiCodeAnalysisStrategicObjectivesVocab-1.0*.'

2.6.4.1 AntiCodeAnalysisStrategicObjectivesEnum-1.0

The AntiCodeAnalysisStrategicObjectivesEnum is a non-exhaustive enumeration of Strategic Objectives of the anti-code analysis Capability.

Enumeration Value	Description
anti-debugging	Indicates that the malware instance is able to prevent itself from being debugged and/or from being run in a debugger or is able to make debugging more difficult.
code obfuscation	Indicates that the malware instance is able to obfuscate its code.
anti-disassembly	Indicates that the malware instance is able to prevent itself from being disassembled or make disassembly more difficult.

2.6.5 AntiCodeAnalysisTacticalObjectiveVocab-1.0

The AntiCodeAnalysisTacticalObjectivesVocab is the default MAEC vocabulary for Tactical Objectives of the anti-code analysis Capability. The names of these Tactical Objectives are captured in the Name field, a child of the Tactical_Objective field of type CapabilityObjectiveType (defined in the MAEC Bundle schema).

The MAEC AntiCodeAnalysisTacticalObjectivesVocab-1.0 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:AntiCodeAnalysisTacticalObjectivesTypeEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Anti-Code Analysis Capability Tactical Objectives*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#AntiCodeAnalysisTacticalObjectivesVocab-1.0*.'

2.6.5.1 AntiCodeAnalysisTacticalObjectivesEnum-1.0

The AntiCodeAnalysisTacticalObjectivesEnum is a non-exhaustive enumeration of Tactical Objectives of the anti-code analysis Capability.

Enumeration Value	Description
transform control flow	Indicates that the malware instance is able to transform its control flow.
restructure arrays	Indicates that the malware instance is able to restructure its arrays, making disassembly more difficult.
detect debugging	Indicates that the malware instance is able to detect its execution in a debugger.
prevent debugging	Indicates that the malware instance is able to prevent its execution in a debugger.
defeat flow-oriented (recursive traversal) disassemblers	Indicates that the malware instance is able to defeat its disassembly in a flow-oriented (recursive traversal) disassembler.
defeat linear disassemblers	Indicates that the malware instance is able to prevent its disassembly in a linear disassembler.
obfuscate instructions	Indicates that the malware instance obfuscates its instructions.
obfuscate imports	Indicates that the malware instance is able to obfuscate its import table, making disassembly more difficult.
defeat call graph generation	Indicates that the malware instance is able to defeat accurate call graph generation during disassembly.
obfuscate runtime code	Indicates that the malware instance is able to obfuscate its runtime code.

2.6.6 AntiDetectionStrategicObjectivesVocab-1.0

The AntiDetectionStrategicObjectivesVocab is the default MAEC vocabulary for Strategic Objectives of the anti-detection Capability. The names of these Strategic Objectives are captured in the Name field, a child of the Strategic_Objective field of type CapabilityObjectiveType (defined in the MAEC Bundle schema).

The MAEC AntiDetectionStrategicObjectivesVocab-1.0 extends ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:AntiDetectionStrategicObjectivesTypeEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Anti-Detection Capability Strategic Objectives*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is

			‘*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#AntiDetectionStrategicObjectivesVocab-1.0*.’

2.6.6.1 AntiDetectionStrategicObjectivesEnum-1.0

The AntiDetectionStrategicObjectivesEnum is a non-exhaustive enumeration of Strategic Objectives of the anti-detection Capability.

Enumeration Value	Description
security software evasion	Indicates that the malware instance is able to evade security software (e.g., anti-virus tools).
hide executing code	Indicates that the malware instance is able to hide its executing code.
self-modification	Indicates that the malware instance is able to modify itself.
anti-memory forensics	Indicates that the malware instance is able to prevent or make memory forensics more difficult
hide non-executing code	Indicates that the malware instance is able to hide its non-executing code.
hide malware artifacts	Indicates that the malware instance is able to hide its artifacts.

2.6.7 AntiDetectionTacticalObjectivesVocab-1.0

The AntiDetectionTacticalObjectivesVocab is the default MAEC vocabulary for Tactical Objectives of the anti-detection analysis Capability. The names of these Tactical Objectives are captured in the Name field, a child of the Tactical_Objective field of type CapabilityObjectiveType (defined in the MAEC Bundle schema).

The MAEC AntiDetectionTacticalObjectivesVocab-1.0 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:AntiDetectionTacticalObjectivesTypeEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is ‘*MAEC Default Anti-Detection Capability Tactical Objectives*.’
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is ‘*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#AntiDetectionTacticalObjectivesVocab-1.0*.’

2.6.7.1 AntiDetectionTacticalObjectivesEnum-1.0

The `AntiDetectionTacticalObjectivesEnum` is a non-exhaustive enumeration of Tactical Objectives of the anti-detection Capability.

Enumeration Value	Description
hide open network ports	Indicates that the malware instance is able to hide its open network ports.
execute before/external to kernel/hypervisor	Indicates that the malware instance is able to execute some or all of its code before or external to the system's kernel or hypervisor (e.g., through the BIOS).
encrypt self	Indicates that the malware is able to encrypt itself.
hide processes	Indicates that the malware instance is able to hide its processes.
hide network traffic	Indicates that the malware instance is able to hide its network traffic.
change/add content	Indicates that the malware instance is able to change or add to its content.
execute stealthy code	Indicates that the malware instance is able to execute some or all of its code in a hidden manner (e.g., by injecting it into a benign process).
hide registry artifacts	Indicates that the malware instance is able to hide its Windows registry artifacts.
hide userspace libraries	Indicates that the malware instance is able to hide its usage of userspace libraries.
hide arbitrary virtual memory	Indicates that the malware instance is able to hide arbitrary virtual memory to prevent retrieval.
execute non-main cpu code	Indicates that the malware instance is able to execute some or all of its code on a secondary, non CPU processor (e.g., a GPU).
feed misinformation during physical memory acquisition	Indicates that the malware instance is able to report inaccurate data when the content of physical memory is retrieved.
prevent physical memory acquisition	Indicates that the malware instance is able to prevent the contents of a system's physical memory from being retrieved.
prevent native api hooking	Indicates that the malware instance is able to prevent other software from hooking native APIs.
obfuscate artifact properties	Indicates that the malware instance is able to hide the properties of its artifacts (e.g., by altering timestamps).
hide kernel modules	Indicates that the malware instance is able to hide its usage of kernel modules.
hide code in file	Indicates that the malware instance is able to hide its code in a file.
hide services	Indicates that the malware instance is able to hide any system services it creates or injects itself into.
hide file system artifacts	Indicates that the malware instance is able to hide its file system artifacts.
hide threads	Indicates that the malware instance is able to hide its threads.

2.6.8 AntiRemovalStrategicObjectivesVocab-1.0

The AntiRemovalStrategicObjectivesVocab is the default MAEC vocabulary for Strategic Objectives of the anti-removal Capability. The names of these Strategic Objectives are captured in the Name field, a child of the Strategic_Objective field of type CapabilityObjectiveType (defined in the MAEC Bundle schema).

The MAEC AntiRemovalStrategicObjectivesVocab-1.0 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:AntiRemovalStrategicObjectivesTypeEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Anti-Removal Capability Strategic Objectives*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#AntiRemovalStrategicObjectivesVocab-1.0*.'

2.6.8.1 AntiRemovalStrategicObjectivesEnum-1.0

The AntiRemovalStrategicObjectivesEnum is a non-exhaustive enumeration of Strategic Objectives of the anti-removal Capability.

Enumeration Value	Description
prevent malware artifact access	Indicates that the malware instance is able to prevent its artifacts from being accessed.
prevent malware artifact deletion	Indicates that the malware instance is able to prevent its artifacts from being deleted from a system.

2.6.9 AntiRemovalTacticalObjectivesVocab-1.0

The AntiRemovalTacticalObjectivesVocab is the default MAEC vocabulary for Tactical Objectives of the anti-removal Capability. The names of these Tactical Objectives are captured in the Name field, a child of the Tactical_Objective field of type CapabilityObjectiveType (defined in the MAEC Bundle schema).

The MAEC AntiRemovalTacticalObjectivesVocab-1.0 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:AntiRemovalTacticalObjectivesTypeEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Anti-Removal Capability Tactical Objectives*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#AntiRemovalTacticalObjectivesVocab-1.0*.'

2.6.9.1 AntiRemovalTacticalObjectivesEnum-1.0

The AntiRemovalTacticalObjectivesEnum is a non-exhaustive enumeration of Tactical Objectives of the anti-removal Capability.

Enumeration Value	Description
prevent registry deletion	Indicates that the malware instance is able to prevent its Windows registry entries from being deleted from a system.
prevent api unhooking	Indicates that the malware instance is able to prevent its API hooks from being removed.
prevent file access	Indicates that the malware instance is able to prevent access to the file system.
prevent memory access	Indicates that the malware instance is able to prevent access to system memory where it may be storing code or data.
prevent registry access	Indicates that the malware instance is able to prevent access to the Windows registry.
prevent file deletion	Indicates that the malware instance is able to prevent its files from being deleted from a system.

2.6.10 AvailabilityViolationPropertiesVocab-1.0

The AvailabilityViolationPropertiesVocab is the default MAEC vocabulary for properties of the availability violation Capability and its child Objectives. The names of these properties are captured in the Name field, a child of the Property field of type CapabilityPropertyType. The Property field is found on the CapabilityType (which is used to define a

Capability) and on the CapabilityObjectiveType (which is used to define a Strategic Objective or a Tactical Objective). All aforementioned types are defined in the MAEC Bundle schema.

The MAEC AvailabilityViolationPropertiesVocab-1.0 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:AvailabilityViolationPropertiesTypeEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Availability Violation Capability and Objective Properties*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#AvailabilityViolationPropertiesVocab-1.0*.'

2.6.10.1 AvailabilityViolationPropertiesEnum-1.0

The AvailabilityViolationPropertiesEnum is a non-exhaustive enumeration of availability violation Capability, Strategic Objective, and Tactical Objective properties.

Enumeration Value	Description
cryptocurrency type	Refers to the type of cryptocurrency targeted by the 'mine for cryptocurrency' Tactical Objective.

2.6.11 AvailabilityViolationStrategicObjectivesVocab-1.0

The AvailabilityViolationStrategicObjectivesVocab is the default MAEC vocabulary for Strategic Objectives of the availability violation Capability. The names of these Strategic Objectives are captured in the Name field, a child of the Strategic_Objective field of type CapabilityObjectiveType (defined in the MAEC Bundle schema).

The MAEC AvailabilityViolationStrategicObjectivesVocab-1.0 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs: AvailabilityViolationStrategicObjectivesTypeEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Availability Violation Capability Strategic Objectives*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#AvailabilityViolationStrategicObjectivesVocab-1.0*.'

2.6.11.1 AvailabilityViolationStrategicObjectivesEnum-1.0

The AvailabilityViolationStrategicObjectivesEnum is a non-exhaustive enumeration of Strategic Objectives of the availability violation Capability.

Enumeration Value	Description
compromise data availability	Indicates that the malware instance is able to compromise the availability of data on a system.
compromise system availability	Indicates that the malware instance compromises the availability of the system.
cosume system resources	Indicates that the malware instance is able to consume system resources for its own purposes.

2.6.12 AvailabilityViolationTacticalObjectivesVocab-1.0

AvailabilityViolationTacticalObjectivesVocab is the default MAEC vocabulary for Tactical Objectives of the availability violation Capability. The names of these Tactical Objectives are captured in the Name field, a child of the Tactical_Objective of type CapabilityObjectiveType (defined in the MAEC Bundle schema).

The MAEC AvailabilityViolationTacticalObjectivesVocab-1.0 type extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs: AvailabilityViolationTacticalObjectivesTypeEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Availability Violation Capability Tactical Objectives*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#AvailabilityViolationTacticalObjectivesVocab-1.0*.'

2.6.12.1 AvailabilityViolationTacticalObjectivesEnum-1.0

The `AvailabilityViolationTacticalObjectivesEnum` is a non-exhaustive enumeration of Tactical Objectives of the availability violation Capability.

Enumeration Value	Description
denial of service	Indicates that the malware instance is able to cause a server to be unavailable, otherwise known as a denial of service (DOS).
compromise local system availability	Indicates that the malware instance is able to cause the local system to be unavailable.
crack passwords	Indicates that the malware instance is able to consume system resources for password cracking.
mine for cryptocurrency	Indicates that the malware instance is able to consume system resources for cryptocurrency mining.
compromise access to information assets	Indicates that the malware instance is able to prevent data from being accessed (e.g., by encrypting user data on a compromised system).

2.6.13 CommandandControlPropertiesVocab-1.0

`CommandandControlPropertiesVocab` is the default MAEC vocabulary for properties of the command and control Capability and its child Objectives. The names of these properties are captured in the `Name` field, a child of the `Property` field of type `CapabilityPropertyType`. The `Property` field is found on the `CapabilityType` (which is used to define a Capability) and on the `CapabilityObjectiveType` (which is used to define a Strategic Objective or a Tactical Objective). All aforementioned types are defined in the MAEC Bundle schema.

The MAEC `CommandandControlPropertiesVocab-1.0` type extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:CommandandControlPropertiesTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Command and Control Capability and Objective Properties*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#CommandandControlPropertiesVocab-1.0*.'

2.6.13.1 CommandandControlPropertiesEnum-1.0

The `CommandandControlPropertiesEnum` is a non-exhaustive enumeration of command and control Capability, Strategic Objective, and Tactical Objective properties.

Enumeration Value	Description
frequency	Refers to a description of the frequency that the 'receive data from c2 server' and 'send data to c2 server' Strategic Objectives, as well as their child Tactical Objectives, are employed. It is recommended that the description follow the format of "every x [units]", e.g., "every 5 minutes".

2.6.14 CommandandControlStrategicObjectivesVocab-1.0

The `CommandandControlStrategicObjectivesVocab` is the default MAEC vocabulary for Strategic Objectives of the command and control Capability. The names of these Strategic Objectives are captured in the `Name` field, a child of the `Strategic_Objective` field of type `CapabilityObjectiveType` (defined in the MAEC Bundle schema).

The MAEC `CommandandControlStrategicObjectivesVocab-1.0` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:CommandandControlStrategicObjectivesTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC*

			Default Command and Control Capability Strategic Objectives.’
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is ‘*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#CommandandControlStrategicObjectivesVocab-1.0*.’

2.6.14.1 CommandandControlStrategicObjectivesEnum-1.0

The `CommandandControlStrategicObjectivesEnum` is a non-exhaustive enumeration of Strategic Objectives of the command and control Capability.

Enumeration Value	Description
determine c2 server	Indicates that the malware instance is able to identify one or more command and control (C2) servers with which to communicate.
receive data from c2 server	Indicates that the malware instance is able to control its behavior through some external stimulus (e.g., a remotely submitted command).
send data to c2 server	Indicates that the malware instance is able to send some data to a command and control server.

2.6.15 CommandandControlTacticalObjectivesVocab-1.0

The `CommandandControlTacticalObjectivesVocab` is the default MAEC vocabulary for Tactical Objectives of the command and control Capability. The names of these Tactical Objectives are captured in the `Name` field, a child of the `Tactical_Objective` field of type `CapabilityObjectiveType` (defined in the MAEC Bundle schema).

The MAEC `CommandandControlTacticalObjectivesVocab-1.0` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:CommandandControlTacticalObjectivesTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is ‘*MAEC Default Command and Control Capability Tactical Objectives*.’

vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#CommandandControlTacticalObjectivesVocab-1.0*.'

2.6.15.1 CommandandControlTacticalObjectivesEnum-1.0

The `CommandandControlTacticalObjectivesEnum` is a non-exhaustive enumeration of Tactical Objectives of the command and control Capability.

Enumeration Value	Description
check for payload	Indicates that the malware instance is able to query a command and control server to check whether a new malicious payload is available for download
validate data	Indicates that the malware instance is able to validate the integrity of the data it receives from a command and control server.
control malware via remote command	Indicates that the malware instance is able to execute commands issued to it from a remote source such as a command and control server for the purpose of controlling its behavior.
send system information	Indicates that the malware instance is able to send data regarding the system on which it is executing to a command and control server.
send heartbeat data	Indicates that the malware instance is able to send heartbeat data to a command and control server, indicating that it is still active on the host system and able to communicate.
generate c2 domain name(s)	Indicates that the malware instance is able to generate the domain name of the command and control server to which it connects.
update configuration	Indicates that the malware instance is able to update its configuration using data received from a command and control server.

2.6.16 DataExfiltrationPropertiesVocab-1.0

The `DataExfiltrationPropertiesVocab` is the default MAEC vocabulary for properties of the data exfiltration Capability and its child Objectives. The names of these properties are captured in the `Name` field, a child of the `Property` field of type `CapabilityPropertyType`. The `Property` field is found on the `CapabilityType` (which is used to define a Capability) and on the `CapabilityObjectiveType` (which is used to define a Strategic Objective or a Tactical Objective). All aforementioned types are defined in the MAEC Bundle schema.

The MAEC DataExfiltrationPropertiesVocab-1.0 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:DataExfiltrationPropertiesTypeEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Data Exfiltration Capability and Objective Properties*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#DataExfiltrationPropertiesVocab-1.0*.'

2.6.16.1 DataExfiltrationPropertiesEnum-1.0

The DataExfiltrationPropertiesEnum is a non-exhaustive enumeration of data exfiltration Capability, Strategic Objective, and Tactical Objective properties.

Enumeration Value	Description
archive type	Refers to the name of the file archive format used in the 'stage data for exfiltration' Strategic Objective and/or its 'package data' Tactical Objective.
file type	Refers to the name of the file format used for storing data to be exfiltrated as part of the data exfiltration Capability or its child Objectives.

2.6.17 DataExfiltrationStrategicObjectivesVocab-1.0

The DataExfiltrationStrategicObjectivesVocab is the default MAEC vocabulary for Strategic Objectives of the data exfiltration Capability. The names of these Strategic Objectives are captured in the Name field, a child of the Strategic_Objective field of type CapabilityObjectiveType (defined in the MAEC Bundle schema).

The MAEC DataExfiltrationStrategicObjectivesVocab-1.0 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:DataExfiltrationStrategicObjectivesTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Data Exfiltration Capability Strategic Objectives*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#DataExfiltrationStrategicObjectivesVocab-1.0*.'

2.6.17.1 DataExfiltrationStrategicObjectivesEnum-1.0

The `DataExfiltrationStrategicObjectivesEnum` is a non-exhaustive enumeration of Strategic Objectives of the data exfiltration Capability.

Enumeration Value	Description
perform data exfiltration	Indicates that the malware instance is able to perform data exfiltration via some physical or virtual means.
obfuscate data for exfiltration	Indicates that the malware is able to obfuscate data that will be exfiltrated.
stage data for exfiltration	Indicates that the malware instance is able to gather and prepare data for exfiltration.

2.6.18 DataExfiltrationTacticalObjectivesVocab-1.0

The `DataExfiltrationTacticalObjectivesVocab` is the default MAEC vocabulary for Tactical Objectives of the data exfiltration Capability. The names of these Tactical Objectives are captured in the `Name` field, a child of the `Tactical_Objective` field of type `CapabilityObjectiveType` (defined in the MAEC Bundle schema).

The MAEC `DataExfiltrationTacticalObjectivesVocab-1.0` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:DatExfiltrationTacticalObjectivesTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Data Exfiltration Capability Tactical Objectives*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#DataExfiltrationTacticalObjectivesVocab-1.0*.'

2.6.18.1 DataExfiltrationTacticalObjectivesEnum-1.0

The `DataExfiltrationTacticalObjectivesEnum` is a non-exhaustive enumeration of Tactical Objectives of the data exfiltration Capability.

Enumeration Value	Description
exfiltrate via covert channel	Indicates that that the malware instance is able to exfiltrate data using a covert channel.
exfiltrate via fax	Indicates that the malware instance is able to exfiltrate data using a fax system.
exfiltrate via physical media	Indicates that the malware instance is able to exfiltrate data using physical media (e.g., a USB drive).
encrypt data	Indicates that the malware instance is able to encrypt data that will be exfiltrated.
exfiltrate via network	Indicates that the malware instance is able to exfiltrate data across the network.
hide data	Indicates that the malware instance is able to hide data that will be exfiltrated in other formats (also known as steganography).
package data	Indicates that the malware instance is able to package data for exfiltration.
exfiltrate via dumpster dive	Indicates that the malware instance is able to exfiltrate data via dumpster dive (i.e., encoded data printed by malware is viewed as garbage and thrown away to then be physically picked up).
move data to staging server	Indicates that the malware instance is able to move data to be exfiltrated to a particular server to prepare for exfiltration.
exfiltrate via voip/phone	Indicates that the malware instance is able to exfiltrate data (encoded as audio) using a phone system.

2.6.19 DataTheftPropertiesVocab-1.0

The `DataTheftPropertiesVocab` is the default MAEC vocabulary for properties of the data theft Capability and its child Objectives. The names of these properties are captured in the `Name` field, a child of the `Property` field of type `CapabilityPropertyType`. The `Property` field is found on the `CapabilityType` (which is used to define a Capability) and on the CapabilityObjectiveType (which is used to define a Strategic Objective or a Tactical Objective). All aforementioned types are defined in the MAEC Bundle schema.

The MAEC DataTheftPropertiesVocab-1.0 type extends ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:DataTheftPropertiesTypeEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Data Theft Capability and Objective Properties*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#DataTheftPropertiesVocab-1.0*.'

2.6.19.1 DataTheftPropertiesEnum-1.0

The DataTheftPropertiesEnum is a non-exhaustive enumeration of data theft Capability, Strategic Objective, and Tactical Objective properties.

Enumeration Value	Description
targeted application	Refers to the name of an application targeted by the 'steal authentication credentials' Strategic Objective.
targeted website	Refers to the domain name of a website targeted by the 'steal web/network credential' Tactical Objective.

2.6.20 DataTheftStrategicObjectivesVocab-1.0

The DataTheftStrategicObjectivesVocab is the default MAEC vocabulary for Strategic Objectives of the data theft Capability. The names of these Strategic Objectives are captured in the Name field, a child of the Strategic_Objective of type CapabilityObjectiveType (defined in the MAEC Bundle schema).

The MAEC DataTheftStrategicObjectivesVocab-1.0 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:DataTheftStrategicObjectivesTypeEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Data Theft Capability Strategic Objectives*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#DataTheftStrategicObjectivesVocab-1.0*.'

2.6.20.1 DataTheftStrategicObjectivesEnum-1.0

The DataTheftStrategicObjectivesEnum is a non-exhaustive enumeration of Strategic Objectives of the data theft Capability.

Enumeration Value	Description
steal stored information	Indicates that the malware instance is able to steal information stored on a system (e.g., files).
steal user data	Indicates that the malware instance is able to steal user data (e.g., email).
steal system information	Indicates that the malware instance is able to steal information about a system (e.g., network address data).
steal authentication credentials	Indicates that the malware instance is able to steal authentication credentials.

2.6.21 DataTheftTacticalObjectivesVocab-1.0

The DataTheftTacticalObjectivesVocab is the default MAEC vocabulary for Tactical Objectives of the data theft Capability. The names of these Tactical Objectives are captured in the Name field, a child of the Tactical_Objective field of type CapabilityObjectiveType (defined in the MAEC Bundle schema).

The MAEC `DataTheftTacticalObjectivesVocab-1.0` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:DataTheftTacticalObjectivesTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Data Theft Capability Tactical Objectives*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#DataTheftTacticalObjectivesVocab-1.0*.'

2.6.21.1 DataTheftTacticalObjectivesEnum-1.0

The `DataTheftTacticalObjectivesEnum` is a non-exhaustive enumeration of Tactical Objectives of the data theft Capability.

Enumeration Value	Description
steal dialed phone numbers	Indicates that the malware instance is able to steal the list of phone numbers that a user has dialed.
steal email data	Indicates that the malware instance is able to steal a user's email data
steal referrer urls	Indicates that the malware instance is able to steal HTTP referrer information (URL of the Web page that linked to the resource being requested).
steal cryptocurrency data	Indicates that the malware instance is able to steal cryptocurrency data (e.g., Bitcoin wallets).
steal pki software certificate	Indicates that the malware instance is able to steal one or more public key infrastructure (PKI) software certificates.
steal browser cache	Indicates that the malware instance is able to steal a user's browser cache
steal serial numbers	Indicates that the malware instance is able to steal serial numbers stored on a system.

2.6.22 DestructionPropertiesVocab-1.0

The `DestructionPropertiesVocab` is the default MAEC vocabulary for properties of the destruction Capability and its child Objectives. The names of these properties are captured in the `Name` field, a child of the `Property` field of type `CapabilityPropertyType`. The `Property` field is found on the `CapabilityType` (which is used to define a Capability) and on the CapabilityObjectiveType (which is used to define a Strategic Objective or a Tactical Objective). All aforementioned types are defined in the MAEC Bundle schema.

The MAEC DestructionPropertiesVocab-1.0 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:DestructionPropertiesTypeEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Destruction Capability and Objective Properties*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#DestructionPropertiesVocab-1.0*.'

2.6.22.1 DestructionPropertiesEnum-1.0

The DestructionPropertiesEnum is a non-exhaustive enumeration of destruction Capability, Strategic Objective, and Tactical Objective properties.

Enumeration Value	Description
erasure scope	Refers to the scope of the erasure performed by the 'erase data' Tactical Objective. Recommended values are: 'whole disk', or 'targeted files'.

2.6.23 DestructionStrategicObjectivesVocab-1.0

The DestructionStrategicObjectivesVocab is the default MAEC vocabulary for Strategic Objectives of the destruction Capability. The names of these Strategic Objectives are captured in the Name field, a child of the Strategic_Objective field of type CapabilityObjectiveType (defined in the MAEC Bundle schema).

The MAEC DestructionStrategicObjectivesVocab-1.0 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:DestructionStrategicObjectivesTypeEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Destruction Capability Strategic Objectives*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#DestructionStrategicObjectivesVocab-1.0*.'

2.6.23.1 DestructionStrategicObjectivesEnum-1.0

The DestructionStrategicObjectivesEnum is a non-exhaustive enumeration of Strategic Objectives of the destruction Capability.

Enumeration Value	Description
destroy physical entity	Indicates that the malware instance is able to destroy a physical entity.
destroy virtual entity	Indicates that the malware instance is able to destroy a virtual entity.

2.6.24 DestructionTacticalObjectivesVocab-1.0

The DestructionTacticalObjectivesVocab is the default MAEC vocabulary for Tactical Objectives of the destruction Capability. The names of these Tactical Objectives are captured in the Name field, a child of the Tactical_Objective field of type CapabilityObjectiveType (defined in the MAEC Bundle schema).

The MAEC DestructionTacticalObjectivesVocab-1.0 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:DestructionTacticalObjectivesTypeEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description

vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Destruction Capability Tactical Objectives*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#DestructionTacticalObjectivesVocab-1.0*.'

2.6.24.1 DestructionTacticalObjectivesEnum-1.0

The `DestructionTacticalObjectivesEnum` is a non-exhaustive enumeration of Tactical Objectives of the destruction Capability.

Enumeration Value	Description
erase data	Indicates that the malware instance is able to destroy data by erasure.
destroy firmware	Indicates that the malware instance is able to destroy a system's firmware.
destroy hardware	Indicates that the malware instance is able to destroy a system's hardware.

2.6.25 FraudStrategicObjectivesVocab-1.0

The `FraudStrategicObjectivesVocab` is the default MAEC vocabulary for Strategic Objectives of the fraud Capability. The names of these Strategic Objectives are captured in the `Name` field, a child of the `Strategic_Objective` field of type `CapabilityObjectiveType` (defined in the MAEC Bundle schema).

The MAEC `FraudStrategicObjectivesVocab-1.0` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:FraudStrategicObjectivesTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Fraud Capability Strategic Objectives*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#FraudStrategicObjectivesVocab-1.0*.'

2.6.25.1 FraudStrategicObjectivesEnum-1.0

The `FraudStrategicObjectivesEnum` is a non-exhaustive enumeration of Strategic Objectives of the fraud Capability.

Enumeration Value	Description
perform premium rate fraud	Indicates that the malware instance is able to send text messages or dial phone numbers that are charged at premium rates.
perform click fraud	Indicates that the malware instance is able to simulate clicks on website advertisements for the purpose of revenue generation.

2.6.26 FraudTacticalObjectivesVocab-1.0

The `FraudTacticalObjectivesVocab` is the default MAEC vocabulary for Tactical Objectives of the fraud Capability. The names of these Tactical Objectives are captured in the `Name` field, a child of the `Tactical_Objective` field of type `CapabilityObjectiveType` (defined in the MAEC Bundle schema).

The MAEC `FraudTacticalObjectivesVocab-1.0` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:FraudTacticalObjectivesTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Fraud Capability Tactical Objectives*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#FraudTacticalObjectivesVocab-1.0*.'

2.6.26.1 FraudTacticalObjectivesEnum-1.0

The `FraudTacticalObjectivesEnum` is a non-exhaustive enumeration of Tactical Objectives of the fraud Capability.

Enumeration Value	Description

access premium service	Indicates that the malware instance is able to access a premium service.

2.6.27 InfectionPropagationPropertiesVocab-1.0

The `InfectionPropagationPropertiesVocab` is the default MAEC vocabulary for properties of the infection propagation Capability and its child Objectives. The names of these properties are captured in the `Name` field, a child of the `Property` field of type `CapabilityPropertyType`. The `Property` field is found on the `CapabilityType` (which is used to define a Capability) and on the `CapabilityObjectiveType` (which is used to define a Strategic Objective or a Tactical Objective). All aforementioned types are defined in the MAEC Bundle schema.

The MAEC `InfectionPropagationPropertiesVocab-1.0` extends `the ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:InfectionPropagationPropertiesTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default InfectionPropagation Capability and Objective Properties*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#InfectionPropagationPropertiesVocab-1.0*.'

2.6.27.1 InfectionPropagationPropertiesEnum-1.0

The `InfectionPropagationPropertiesEnum` is a non-exhaustive enumeration of infection propagation Capability, Strategic Objective, and Tactical Objective properties.

Enumeration Value	Description
scope	Refers to the scope of the infection or propagation performed by the malware instance via the Infection/Propagation Capability, i.e., whether it infects just the local machine or actively propagates to other machines as well. Recommended values are: 'local' or 'remote'.
infection targeting	Refers to the type of targeting employed by the 'infect remote machine' Strategic Objective, i.e.,

	whether the targeted machines are randomly selected, or chosen from some particular set. Recommended values are: 'targeted', 'semi-targeted', or 'untargeted'.
autonomy	Refers to the type of autonomy employed by the 'infect remote machine' Strategic Objective, i.e., whether the remote infection is performed autonomously. Recommended values are: 'semi-autonomous', 'autonomous'.
targeted file type	Refers to the types of files targeted by the 'infect file' Strategic Objective. It is recommended that files be specified via their extension, e.g., "exe", "pdf", etc.
targeted file architecture type	Refers to type of file architecture targeted by the 'infect file' Strategic Objective. Recommended values are: '32 bit' or '64 bit'.
file infection type	Refers to the type of file infection employed by the 'infect file' Strategic Objective. Recommended values are: 'appending', 'prepending', 'overwriting', 'companion', 'variable key', 'polymorphic', or 'metamorphic'.

2.6.28 InfectionPropagationStrategicObjectivesVocab-1.0

The `InfectionPropagationStrategicObjectivesVocab` is the default MAEC vocabulary for Strategic Objectives of the infection/propagation Capability. The names of these Strategic Objectives are captured in the `Name` field, a child of the `Strategic_Objective` field of type `CapabilityObjectiveType` (defined in the MAEC Bundle schema).

The MAEC `InfectionPropagationStrategicObjectivesVocab-1.0` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:InfectionPropagationStrategicObjectivesTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Infection/Propagation Capability Strategic Objectives*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#InfectionPropagationStrategicObjectivesVocab-1.0*.'

2.6.28.1 InfectionPropagationStrategicObjectivesEnum-1.0

The InfectionPropagationStrategicObjectivesEnum is a non-exhaustive enumeration of Strategic Objectives of the infection/propagation Capability.

Enumeration Value	Description
prevent duplicate infection	Indicates that the malware instance is able to prevent itself from infecting a machine multiple times.
infect file	Indicates that the malware instance is able to infect a file.
infect remote machine	Indicates that the malware instance is able to self-propagate or infect a machine with malware that is different than itself.

2.6.29 InfectionPropagationTacticalObjectivesVocab-1.0

The InfectionPropagationTacticalObjectivesVocab is the default MAEC vocabulary for Tactical Objectives of the infection/propagation Capability. The names of these Tactical Objectives are captured in the Name field, a child of the Tactical_Objective field of type CapabilityObjectiveType (defined in the MAEC Bundle schema).

The MAEC InfectionPropagationTacticalObjectivesVocab-1.0 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:InfectionPropagationTacticalObjectivesTypeEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Infection/Propagation Capability Tactical Objectives*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#InfectionPropagationTacticalObjectivesVocab-1.0*.'

2.6.29.1 InfectionPropagationTacticalObjectivesEnum-1.0

The `InfectionPropagationTacticalObjectivesEnum` is a non-exhaustive enumeration of Tactical Objectives of the infection/propagation Capability.

Enumeration Value	Description
identify file	Indicates that the malware instance is able to identify a file or files on a local, removable, and/or network drive for infection.
perform autonomous remote infection	Indicates that the malware instance is able to infect a remote machine autonomously, without the involvement of any end user (e.g., through the exploitation of a remote procedure call vulnerability).
identify target machine(s)	Indicates that the malware instance is able to identify one or more machines to be targeted for infection via some remote means (e.g., via email or the network).
perform social-engineering based remote infection	Indicates that the malware instance is able to infect remote machines via some method that involves social engineering (e.g., sending an email with a malicious attachment).
inventory victims	indicates that the malware instance is able to keep an inventory of the victims that it remotely infects.
write code into file	indicates that the malware instance is able to write code into a file.
modify file	indicates that the malware instance is able to modify a file in some other manner than writing code to it, such as packing it (in terms of binary executable packing).

2.6.30 IntegrityViolationStrategicObjectivesVocab-1.0

The `IntegrityViolationStrategicObjectivesVocab` is the default MAEC vocabulary for Strategic Objectives of the integrity violation Capability. The names of these Strategic Objectives are captured in the `Name` field, a child of the `Strategic_Objective` field of type `CapabilityObjectiveType` (defined in the MAEC Bundle schema).

The MAEC `IntegrityViolationStrategicObjectivesVocab-1.0` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:IntegrityViolationStrategicObjectivesTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC*

			Default Integrity Violation Capability Strategic Objectives.’
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is ‘*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#IntegrityViolationStrategicObjectivesVocab-1.0*.’

2.6.30.1 IntegrityViolationStrategicObjectivesEnum-1.0

The `IntegrityViolationStrategicObjectivesEnum` is a non-exhaustive enumeration of Strategic Objectives of the integrity violation Capability.

Enumeration Value	Description
compromise system operational integrity	Indicates that the malware instance is able to compromise the operational integrity of a system.
compromise user data integrity	Indicates that the malware instance is able to compromise a system's user data.
annoy user	Indicates that the malware instance is able to annoy the users of a system.
compromise network operational integrity	Indicate that the malware instance is able to compromise the operational integrity of a network.
compromise system data integrity	Indicates that the malware instance is able to compromise the integrity of a system's data.

2.6.31 IntegrityViolationTacticalObjectivesVocab-1.0

The `IntegrityViolationTacticalObjectivesVocab` is the default MAEC vocabulary for Tactical Objectives of the integrity violation Capability. The names of these Tactical Objectives are captured in the `Name` field, a child of the `Tactical_Objective` field of type `CapabilityObjectiveType` (defined in the MAEC Bundle schema).

The MAEC `IntegrityViolationTacticalObjectivesVocab-1.0` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:IntegrityViolationTacticalObjectivesTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is ‘*MAEC Default Integrity Violation Capability Tactical Objectives*.’

vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#IntegrityViolationTacticalObjectivesVocab-1.0.*'

2.6.31.1 IntegrityViolationTacticalObjectivesEnum-1.0

The `IntegrityViolationTacticalObjectivesEnum` is a non-exhaustive enumeration of Tactical Objectives of the integrity violation Capability.

Enumeration Value	Description
subvert system	Indicates that the malware instance is able to subvert a system to perform beyond its operational boundaries or to perform tasks for which it was not originally intended.
corrupt system data	Indicates that the malware instance is able to corrupt a system's data.
annoy local system user	Indicates that the malware instance is able to annoy local system users.
intercept/manipulate network traffic	Indicates that the malware is able to intercept and/or manipulate traffic on a network.
annoy remote user	Indicates that the malware instance is able to annoy a remote user.
corrupt user data	Indicates that the malware instance is able to corrupt a system's user data.

2.6.32 MachineAccessControlPropertiesVocab-1.0

The `MachineAccessControlPropertiesVocab` is the default MAEC vocabulary for properties of the machine access control Capability and its child Objectives. The names of these properties are captured in the `Name` field, a child of the `Property` field of type `CapabilityPropertyType`. The `Property` field is found on the `CapabilityType` (which is used to define a Capability) and on the `CapabilityObjectiveType` (which is used to define a Strategic Objective or a Tactical Objective). All aforementioned types are defined in the MAEC Bundle schema.

The MAEC `MachineAccessControlPropertiesVocab-1.0` extends `the ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:MachineAccessControlPropertiesTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description

vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is ‘*MAEC Default Machine Access Control Capability and Objective Properties*.’
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is ‘*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#MachineAccessControlPropertiesVocab-1.0*.’

2.6.32.1 MachineAccessControlPropertiesEnum-1.0

The `MachineAccessControlPropertiesEnum` is a non-exhaustive enumeration of machine access control Capability, Strategic Objective, and Tactical Objective properties.

Enumeration Value	Description
backdoor type	Refers to the type of backdoor, e.g., reverse shell, employed by the ‘install backdoor’ Strategic Objective.

2.6.33 MachineAccessControlStrategicObjectivesVocab-1.0

The `MachineAccessControlStrategicObjectivesVocab` is the default MAEC vocabulary for Strategic Objectives of the machine access control Capability. The names of these Strategic Objectives are captured in the `Name` field, a child of the `Strategic_Objective` field of type `CapabilityObjectiveType` (defined in the MAEC Bundle schema).

The MAEC `MachineAccessControlStrategicObjectivesVocab-1.0` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:MachineAccessControlStrategicObjectivesTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is ‘*MAEC Default Machine Access Control Capability Strategic Objectives*.’
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is ‘*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#MachineAccessControlStrategicObjectivesVocab-1.0*.’

2.6.33.1 MachineAccessControlStrategicObjectivesEnum-1.0

The `MachineAccessControlStrategicObjectivesEnum` is a non-exhaustive enumeration of Strategic Objectives of the machine access control Capability.

Enumeration Value	Description
control local machine	Indicates that the malware instance is able to control the machine on which it is resident. Examples of malware with this capability include bots, backdoors, and RATs.
install backdoor	Indicates that the malware instance is able to install a backdoor, capable of providing covert remote access to the machine on which it is resident.

2.6.34 MachineAccessControlTacticalObjectivesVocab-1.0

The `MachineAccessControlTacticalObjectivesVocab` is the default MAEC vocabulary for Tactical Objectives of the machine access control Capability. The names of these Tactical Objectives are captured in the `Name` field, a child of the `Tactical_Objective` field of type `CapabilityObjectiveType` (defined in the MAEC Bundle schema).

The MAEC `MachineAccessControlTacticalObjectivesVocab-1.0` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:MachineAccessControlTacticalObjectivesTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Machine Access Control Capability Tactical Objectives*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#MachineAccessControlTacticalObjectivesVocab-1.0*.'

2.6.34.1 MachineAccessControlTacticalObjectivesEnum-1.0

The MachineAccessControlTacticalObjectivesEnum is a non-exhaustive enumeration of Tactical Objectives of the machine access control Capability.

Enumeration Value	Description
control machine via remote command	Indicates that the malware instance is able to execute commands issued to it from a remote source, for the purpose of controlling the machine on which it is resident.

2.6.35 PersistencePropertiesVocab-1.0

The PersistencePropertiesVocab is the default MAEC vocabulary for properties of the persistence Capability and its child Objectives. The names of these properties are captured in the Name field, a child of the Property field of type CapabilityPropertyType. The Property field is found on the CapabilityType (which is used to define a Capability) and on the CapabilityObjectiveType (which is used to define a Strategic Objective or a Tactical Objective). All aforementioned types are defined in the MAEC Bundle schema.

The MAEC PersistencePropertiesVocab-1.0 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:PersistencePropertiesTypeEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Persistence Capability and Objective Properties*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#PersistencePropertiesVocab-1.0*.'

2.6.35.1 PersistencePropertiesEnum-1.0

The PersistencePropertiesEnum is a non-exhaustive enumeration of persistence Capability, Strategic Objective, and Tactical Objective properties.

Enumeration Value	Description
scope	Refers to the scope of persistence employed by the persistence Capability, i.e., whether the malware instance make itself persist, or whether it makes other malware components persist. Recommended values are: 'self', or 'other malware/components'.

2.6.36 PersistenceStrategicObjectivesVocab-1.0

The PersistenceStrategicObjectivesVocab is the default MAEC vocabulary for Strategic Objectives of the persistence Capability. The names of these Strategic Objectives are captured in the Name field, a child of the Strategic_Objective of type CapabilityObjectiveType (defined in the MAEC Bundle schema).

The MAEC PersistenceStrategicObjectivesVocab-1.0 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:PersistenceStrategicObjectivesTypeEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Persistence Capability Strategic Objectives*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#PersistenceStrategicObjectivesVocab-1.0*.'

2.6.36.1 PersistenceStrategicObjectivesEnum-1.0

The PersistenceStrategicObjectivesEnum is a non-exhaustive enumeration of Strategic Objectives of the persistence Capability.

Enumeration Value	Description
persist to re-infect system	Indicates that the malware instance is able to re-infect a system after some of its components have been removed.

gather information for improvement	Indicates that the malware instance is able to gather information from its environment to make itself less likely to be detected.
ensure compatibility	Indicates that the malware instance is able to manipulate or modify the system on which it executes to ensure that it is able to continue executing.
persist to continuously execute on system	Indicates that the malware instance is able to continue to execute on a system after significant system events (e.g., after a reboot).

2.6.37 PersistenceTacticalObjectivesVocab-1.0

The `PersistenceTacticalObjectivesVocab` is the default MAEC vocabulary for Tactical Objectives of the persistence Capability. The names of these Tactical Objectives are captured in the `Name` field, a child of the `Tactical_Objective` field of type `CapabilityObjectiveType` (defined in the MAEC Bundle schema) as its base type.

The MAEC `PersistenceTacticalObjectivesVocab-1.0` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:PersistenceTacticalObjectivesTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Persistence Capability Tactical Objectives*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#PersistenceTacticalObjectivesVocab-1.0*.'

2.6.37.1 PersistenceTacticalObjectivesEnum-1.0

The `PersistenceTacticalObjectivesEnum` is a non-exhaustive enumeration of Tactical Objectives of the persistence Capability.

Enumeration Value	Description
reinstantiate self after initial detection	Indicates that the malware instance s able to re-establish itself on the system after it is initially detected.

limit application type/version	Indicates that the malware instance is able to limit the type or version of an application that runs on a system in order to ensure that it is able to continue executing.
persist after os install/reinstall	Indicates that the malware instance is able to continue to execute after the operating system is installed or reinstalled.
drop/retrieve debug log file	Indicates that the malware instance is able to generate and retrieve a log file of errors associated with the malware.
persist independent of hard disk/os changes	Indicates that the malware instance is able to continue to execute after changes to the hard disk or the operating system have been made.
persist after system reboot	Indicates that the malware instance is able to continue to execute after a system reboot.

2.6.38 PrivilegeEscalationPropertiesVocab-1.0

The `PrivilegeEscalationPropertiesVocab` is the default MAEC vocabulary for properties of the privilege escalation Capability and its child Objectives. The names of these properties are captured in the `Name` field, a child of the `Property` field of type `CapabilityPropertyType`. The `Property` field is found on the `CapabilityType` (which is used to define a Capability) and on the `CapabilityObjectiveType` (which is used to define a Strategic Objective or a Tactical Objective). All aforementioned types are defined in the MAEC Bundle schema.

The MAEC `PrivilegeEscalationPropertiesVocab-1.0` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:PrivilegeEscalationPropertiesTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Privilege Escalation Capability and Objective Properties*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#PrivilegeEscalationPropertiesVocab-1.0*.'

2.6.38.1 PrivilegeEscalationPropertiesEnum-1.0

The `PrivilegeEscalationPropertiesEnum` is a non-exhaustive enumeration of privilege escalation Capability, Strategic Objective, and Tactical Objective properties.

Enumeration Value	Description
user privilege escalation type	Refers to the type of user privilege escalation employed by the 'escalate user privilege' Strategic Objective. Recommended values are: 'horizontal', or 'vertical'.

2.6.39 PrivilegeEscalationStrategicObjectivesVocab-1.0

The `PrivilegeEscalationStrategicObjectivesVocab` is the default MAEC vocabulary for Strategic Objectives of the privilege escalation Capability. The names of these Strategic Objectives are captured in the `Name` field, a child of the `Strategic_Objective` field of type `CapabilityObjectiveType` (defined in the MAEC Bundle schema).

The MAEC `PrivilegeEscalationStrategicObjectivesVocab-1.0` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:PrivilegeEscalationStrategicObjectivesTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Privilege Escalation Capability Strategic Objectives*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#PrivilegeEscalationStrategicObjectivesVocab-1.0*.'

2.6.39.1 PrivilegeEscalationStrategicObjectivesEnum-1.0

The `PrivilegeEscalationStrategicObjectivesEnum` is a non-exhaustive enumeration of Strategic Objectives of the privilege escalation Capability.

Enumeration Value	Description
impersonate user	Indicates that the malware instance is able to impersonate another user to operate within a different security context (also known as horizontal privilege escalation).
escalate user privilege	Indicates that the malware instance is able to obtain a higher level of access than intended by the system (also known as vertical privilege escalation).

2.6.40 PrivilegeEscalationTacticalObjectivesVocab-1.0

The `PrivilegeEscalationTacticalObjectivesVocab` is the default MAEC vocabulary for Tactical Objectives of the privilege escalation Capability. The names of these Tactical Objectives are captured in the `Name` field, a child of the `Tactical_Objective` field of type `CapabilityObjectiveType` (defined in the MAEC Bundle schema).

The MAEC `PrivilegeEscalationTacticalObjectivesVocab-1.0` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:PrivilegeEscalationTacticalObjectivesTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Privilege Escalation Capability Tactical Objectives*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#PrivilegeEscalationTacticalObjectivesVocab-1.0*.'

2.6.40.1 PrivilegeEscalationTacticalObjectivesEnum-1.0

The `PrivilegeEscalationTacticalObjectivesEnum` is a non-exhaustive enumeration of Tactical Objectives of the privilege escalation Capability.

Enumeration Value	Description
elevate cpu mode	Indicates that the malware instance is able to elevate the CPU (processor) mode under which it executes.

2.6.41 ProbingStrategicObjectivesVocab-1.0

The `ProbingStrategicObjectivesVocab` is the default MAEC vocabulary for Strategic Objectives of the probing Capability. The names of these Strategic Objectives are captured in the `Name` field, a child of the `Strategic_Objective` field of type `CapabilityObjectiveType` (defined in the MAEC Bundle schema).

The MAEC `ProbingStrategicObjectivesVocab-1.0` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:ProbingStrategicObjectivesTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Probing Capability Strategic Objectives*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#ProbingStrategicObjectivesVocab-1.0*.'

2.6.41.1 ProbingStrategicObjectivesEnum-1.0

The `ProbingStrategicObjectivesEnum` is a non-exhaustive enumeration of Strategic Objectives of the probing Capability.

Enumeration Value	Description
probe host configuration	Indicates that the malware instance is able to probe the configuration of the host system on which it executes.
probe network configuration	Indicates that the malware instance is able to probe the properties of its network environment, e.g., to determine whether it funnels traffic through a proxy.

2.6.42 ProbingTacticalObjectivesVocab-1.0

The `ProbingTacticalObjectivesVocab` is the default MAEC vocabulary for Tactical Objectives of the probing Capability. The names of these Tactical Objectives are captured in the `Name` field, a child of the `Tactical_Objective` field of type `CapabilityObjectiveType` (defined in the MAEC Bundle schema).

The MAEC `ProbingTacticalObjectivesVocab-1.0` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:ProbingTacticalObjectivesTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Probing Capability Tactical Objectives*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#ProbingTacticalObjectivesVocab-1.0*.'

2.6.42.1 ProbingTacticalObjectivesEnum-1.0

The `ProbingTacticalObjectivesEnum` is a non-exhaustive enumeration of Tactical Objectives of the probing Capability.

Enumeration Value	Description
identify os	Indicates that the malware instance is able to identify the operating system under which it executes.
check for proxy	Indicates that the malware instance is able to check whether the network environment in which it executes contains a hardware or software proxy.
check for firewall	Indicates that the malware instance is able to check whether the network environment in which it executes contains a hardware or software firewall.
check for network drives	Indicates that the malware instance is able to check for network drives that may be present in the network environment.
map local network	Indicates that the malware instance is able to map the layout of the local network environment in which it executes.
inventory system applications	Indicates that the malware instance is able to inventory the applications installed on the system on which it executes.
check language	Indicates that the malware instance is able to check the language of the host system on which it executes.
check for internet connectivity	Indicates that the malware instance is able to check whether the network environment in which it executes is connected to the internet.

2.6.43 RemoteMachineManipulationStrategicObjectivesVocab-1.0

The `RemoteMachineManipulationStrategicObjectivesVocab` is the default MAEC vocabulary for Strategic Objectives of the remote machine manipulation Capability. The names of these Strategic Objectives are captured in the `Name` field, a child of the `Strategic_Objective` field of type `CapabilityObjectiveType` (defined in the MAEC Bundle schema).

The MAEC `RemoteMachineManipulationStrategicObjectivesVocab-1.0` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:RemoteMachineManipulationStrategicObjectivesTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Remove Machine Manipulation Capability Strategic Objectives*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#RemoteMachineManipulationStrategicObjectivesVocab-1.0*.'

2.6.43.1 RemoteMachineManipulationStrategicObjectivesEnum-1.0

The `RemoteMachineManipulationStrategicObjectivesEnum` is a non-exhaustive enumeration of Strategic Objectives of the remote machine manipulation Capability.

Enumeration Value	Description
access remote machine	Indicates that the malware instance is able to access a remote machine.
search for remote machine	Indicates that the malware instance is able to search for remote machines to target.

2.6.44 RemoteMachineManipulationTacticalObjectivesVocab-1.0

The `RemoteMachineManipulationTacticalObjectivesVocab` is the default MAEC vocabulary for Tactical Objectives of the remote machine manipulation Capability. The names of these Tactical Objectives are captured in the `Name` field, a child of the `Tactical_Objective` field of type `CapabilityObjectiveType` (defined in the MAEC Bundle schema).

The MAEC RemoteMachineManipulationTacticalObjectivesVocab-1.0 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs: RemoteMachineManipulationTacticalObjectivesTypeEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Remote Machine Manipulation Capability Tactical Objectives*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#RemoteMachineManipulationTacticalObjectivesVocab-1.0*.'

2.6.44.1 RemoteMachineManipulationTacticalObjectivesEnum-1.0

The RemoteMachineManipulationTacticalObjectivesEnum is a non-exhaustive enumeration of Tactical Objectives of the remote machine manipulation Capability.

Enumeration Value	Description
compromise remote machine	Indicates that the malware instance is able to gain control of a remote machine through compromise.

2.6.45 SecondaryOperationPropertiesVocab-1.0

The SecondaryOperationPropertiesVocab is the default MAEC vocabulary for properties of the secondary operation Capability and its child Objectives. The names of these properties are captured in the Name field, a child of the Property field of type CapabilityPropertyType. The Property field is found on the CapabilityType (which is used to define a Capability) and on the CapabilityObjectiveType (which is used to define a Strategic Objective or a Tactical Objective). All aforementioned types are defined in the MAEC Bundle schema.

The MAEC SecondaryOperationPropertiesVocab-1.0 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:SecondaryOperationPropertiesTypeEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Secondary Operation Capability and Objective Properties*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#SecondaryOperationPropertiesVocab-1.0*.'

2.6.45.1 SecondaryOperationPropertiesEnum-1.0

The SecondaryOperationPropertiesEnum is a non-exhaustive enumeration of secondary operation Capability, Strategic Objective, and Tactical Objective properties.

Enumeration Value	Description
trigger type	Refers to a description of the trigger used to wake or terminate the malware instance in the 'lie dormant' or 'suicide exit' Strategic Objectives, respectively.

2.6.46 SecondaryOperationStrategicObjectivesVocab-1.0

The SecondaryOperationStrategicObjectivesVocab is the default MAEC vocabulary for Strategic Objectives of the secondary operation Capability. The names of these Strategic Objectives are captured in the Name field, a child of the Strategic_Objective field of type CapabilityObjectiveType (defined in the MAEC Bundle schema).

The MAEC SecondaryOperationStrategicObjectivesVocab-1.0 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:SecondaryOperationStrategicObjectivesTypeEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Secondary Operation Capability Strategic Objectives*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#SecondaryOperationStrategicObjectivesVocab-1.0*.'

2.6.46.1 SecondaryOperationStrategicObjectivesEnum-1.0

The `SecondaryOperationStrategicObjectivesEnum` is a non-exhaustive enumeration of Strategic Objectives of the secondary operation Capability.

Enumeration Value	Description
patch operating system file(s)	Indicates that the malware instance is able to patch or modify the critical system files of the operating system under which it executes.
remove traces of infection	Indicates that the malware instance is able to remove traces of its infection of a system.
log activity	Indicates that the malware instance is able to log its own activity.
lay dormant	Indicates that the malware instance is able to lay dormant on a system for some period of time.
install other components	Indicates that the malware instance is able to install additional components. This encompasses the dropping/downloading of other malicious components such as libraries, other malware, and tools.
suicide exit	Indicates that the malware instance is able to terminate itself based on some condition or value.

2.6.47 SecondaryOperationTacticalObjectivesVocab-1.0

The `SecondaryOperationTacticalObjectivesVocab` is the default MAEC vocabulary for Tactical Objectives of the secondary operation Capability. The names of these Tactical Objectives are captured in the `Name` field, a child of the `Tactical_Objective` field of type `CapabilityObjectiveType` (defined in the MAEC Bundle schema).

The MAEC `SecondaryOperationTacticalObjectivesVocab-1.0` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:SecondaryOperationTacticalObjectivesTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Secondary Operation Capability Tactical Objectives*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#SecondaryOperationTacticalObjectivesVocab-1.0*.'

2.6.47.1 SecondaryOperationTacticalObjectivesEnum-1.0

The `SecondaryOperationTacticalObjectivesEnum` is a non-exhaustive enumeration of Tactical Objectives of the secondary operation Capability.

Enumeration Value	Description
install secondary module	Indicates that the malware instance is able to install a secondary module (typically related to itself).
install secondary malware	Indicates that the malware instance is able to install another malware instance.
install legitimate software	Indicates that the malware instance is able to install legitimate software.
remove self	Indicates that the malware instance is able to remove itself from the system.
remove system artifacts	Indicates that the malware instance is able to remove its artifacts from a system.

2.6.48 SecurityDegradationPropertiesVocab-1.0

The `SecurityDegradationPropertiesVocab` is the default MAEC vocabulary for properties of the security degradation Capability and its child Objectives. The names of these properties are captured in the `Name` field, a child of the `Property` field of type `CapabilityPropertyType`. The `Property` field is found on the `CapabilityType` (which is used to define a Capability) and on the `CapabilityObjectiveType` (which is used to define a Strategic Objective or a Tactical Objective). All aforementioned types are defined in the MAEC Bundle schema.

The MAEC `SecurityDegradationPropertiesVocab-1.0` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:SecurityDegradationPropertiesTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Security Degradation Capability and Objective Properties*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#SecurityDegradationPropertiesVocab-1.0*.'

2.6.48.1 SecurityDegradationPropertiesEnum-1.0

The `SecurityDegradationPropertiesEnum` is a non-exhaustive enumeration of security degradation Capability, Strategic Objective, and Tactical Objective properties.

Enumeration Value	Description
targeted program	Refers to the name of a program targeted by the 'degrade security programs' Strategic Objective or one of its child Tactical Objectives.

2.6.49 SecurityDegradationStrategicObjectivesVocab-1.0

The `SecurityDegradationStrategicObjectivesVocab` is the default MAEC vocabulary for Strategic Objectives of the security degradation Capability. The names of these Strategic Objectives are captured in the `Name` field of the `Strategic_Objective` field of type `CapabilityObjectiveType` (defined in the MAEC Bundle schema).

The MAEC `SecurityDegradationStrategicObjectivesVocab-1.0` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:SecurityDegradationStrategicObjectivesTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Security Degradation Capability Strategic Objectives*.'

vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#SecurityDegradationStrategicObjectivesVocab-1.0.*'

2.6.49.1 SecurityDegradationStrategicObjectivesEnum-1.0

The SecurityDegradationStrategicObjectivesEnum is a non-exhaustive enumeration of Strategic Objectives of the security degradation Capability.

Enumeration Value	Description
disable server provider security features	Indicates that the malware instance is able to bypass or disable third-party security features that would otherwise identify or notify users of its presence.
degrade security programs	Indicates that the malware instance is able to degrade security programs running on a system, either by stopping them from executing or by making changes to their code or configuration parameters.
disable system updates	Indicates that the malware instance is able to disable the downloading and installation of system updates.
disable os security features	Indicates that the malware instance is able to bypass inherent operating system security mechanisms that typically involve elevated privileges.
disable [host-based or os] access controls	Indicates that the malware instance is able to bypass access control mechanisms designed to prevent unauthorized or unprivileged use or execution of applications or files.

2.6.50 SecurityDegradationTacticalObjectivesVocab-1.0

The SecurityDegradationTacticalObjectivesVocab is the default MAEC vocabulary for Tactical Objectives of the security degradation Capability. The names of these Tactical Objectives are captured in the Name field, a child of the Tactical_Objective field of type CapabilityObjectiveType (defined in the MAEC Bundle schema).

The MAEC SecurityDegradationTacticalObjectivesVocab-1.0 extends the ControlledVocabularyStringType defined in CybOX Common. Thus, Name fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs: SecurityDegradationTacticalObjectivesTypeEnum-1.0; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Security Degradation Capability Tactical Objectives*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#SecurityDegradationTacticalObjectivesVocab-1.0*.'

2.6.50.1 SecurityDegradationTacticalObjectivesEnum-1.0

The `SecurityDegradationTacticalObjectivesEnum` is a non-exhaustive enumeration of Tactical Objectives of the security degradation Capability.

Enumeration Value	Description
stop execution of security program	Indicates that the malware instance is able to stop one or more security programs that may already be executing on a system.
disable firewall	Indicates that the malware instance is able to evade or disable the host-based firewall or otherwise prevent the blocking of network communications.
disable access right checking	Indicates that the malware instance is able to bbypass, disable, or modify the access tokens or access control lists, thereby enabling the malware to read, write, or execute a file with one or more of these controls set.
disable kernel patching protection	Indicates that the malware instance is able to bypass or disable PatchGuard; thus it is capable of operating at the same level as the kernel and kernel mode drivers (KMD).
prevent access to security websites	Indicates that the malware instance is able to prevent access from a system to one or more security vendor or security-related websites.
remove sms warning messages	Indicates that the malware instance is able to capture the message body of incoming SMS messages and abort the broadcasting of a message that meets a certain criteria.
modify security program configuration	Indicates that the malware instance is able to modify the configuration of one or more security programs running on a system in order to hamper their usefulness and ability to detect the malware instance.
prevent security program from running	Indicates that the malware instance is able to prevent one or more security programs from running on a system.

disable system update services/daemons	Indicates that the malware instance is able to disable system update services or daemons that may be running on a system.
disable system service pack/patch installation	Indicates that the malware instance is able to disable the system's ability to install service packs or patches.
disable system file overwrite protection	Indicates that the malware instance is able to bypass or disable the Windows file protection feature; thus, enabling system files to be modified or replaced.
disable privilege limiting	Indicates that the malware instance is able to bypass controls that limit the privileges that can be granted to a user or entity.
gather security product info	Indicates that the malware instance is able to gather information about the security products installed or running on a system.
disable os security alerts	Indicates that the malware instance is able to evade or disable identification and/or notification of its presence by inherent features of the operating system.
disable user account control	Indicates that the malware instance is able to bypass or disable user account control (UAC); thus, enabling a user to run an application with elevated privileges.

2.6.51 SpyingStrategicObjectivesVocab-1.0

The `SpyingStrategicObjectivesVocab` is the default MAEC vocabulary for Strategic Objectives of the spying Capability. The names of these Strategic Objectives are captured in the `Name` field, a child of the `Strategic_Objective` field of type `CapabilityObjectiveType` (defined in the MAEC Bundle schema).

The MAEC `SpyingStrategicObjectivesVocab-1.0` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:SpyingStrategicObjectivesTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Spying Capability Strategic Objectives*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#SpyingStrategicObjectivesVocab-1.0*.'

2.6.51.1 SpyingStrategicObjectivesEnum-1.0

The `SpyingStrategicObjectivesEnum` is a non-exhaustive enumeration of Strategic Objectives of the spying Capability.

Enumeration Value	Description
capture system input peripheral data	Indicates that the malware instance is able to capture data from a system's input peripheral devices.
capture system state data	Indicates that the malware instance is able to capture information about a system's state (e.g., from its RAM).
capture system interface data	Indicates that the malware instance is able to capture data from a system's interfaces.
capture system output peripheral data	indicates that the malware instance is able to capture data sent to a system's output peripheral devices.

2.6.52 SpyingTacticalObjectivesVocab-1.0

The `SpyingTacticalObjectivesVocab` is the default MAEC vocabulary for Tactical Objectives of the spying Capability. The namesof these Tactical Objectives are captured in the `Name` field, a child of the `Tactical_Objective` field of type `CapabilityObjectiveType` (defined in the MAEC Bundle schema).

The MAEC `SpyingTacticalObjectivesVocab-1.0` extends the `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:SpyingTacticalObjectivesTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Spying Capability Tactical Objectives*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#SpyingTacticalObjectivesVocab-1.0*.'

2.6.52.1 SpyingTacticalObjectivesEnum-1.0

The `SpyingTacticalObjectivesEnum` is a non-exhaustive enumeration of Tactical Objectives of the spying Capability.

Enumeration Value	Description
capture system screenshot	Indicates that the malware instance is able to capture images of what is currently being displayed on a system's screen, either locally or remotely via a remote desktop protocol.
capture camera input	Indicates that the malware instance is able to capture data from a system's camera.
capture file system	Indicates that the malware instance is able to capture data from a system's file system.
capture printer output	Indicates that the malware instance is able to capture data sent to a system's printer.
capture gps data	Indicates that the malware instance is able to capture system GPS data.
capture keyboard input	Indicates that the malware instance is able to capture data from a system's keyboard.
capture mouse input	Indicates that the malware instance is able to capture data from a system's mouse.
capture microphone input	Indicates that the malware instance is able to capture data from a system's microphone.
capture system network traffic	Indicates that the malware instance is able to capture system network traffic.
capture touchscreen input	Indicates that the malware instance is able to capture data from a system's touchscreen.
capture system memory	Indicates that the malware instance is able to capture data from a system's RAM.

2.7 Malware Subject-Related Default Vocabularies

The default vocabularies in this section are related to the Malware Subjects in a MAEC Package.

2.7.1 MalwareConfigurationParameterVocab-1.0

`MalwareConfigurationParameterVocab` is the default MAEC vocabulary for malware configuration parameter names, which are captured in the `Configuration_Details` field (of type `MalwareConfigurationDetailsType`) of a Malware Subject. More specifically, the name of a configuration parameter is captured via the `Name` field, a child of the `Configuration_Parameter` field (of type `MalwareConfiguationParameterType`), itself a child of the `Configuration_Details` field.

The MAEC `MalwareConfigurationParameterVocab-1.0` type extends `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Name` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:MalwareConfigurationParameterEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Malware Configuration Parameter Names*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#MalwareConfigurationParameterVocab-1.0*.'

2.7.1.1 MalwareConfigurationParameterEnum-1.0

The `MalwareConfigurationParameterEnum` is a non-exhaustive enumeration of malware configuration parameter names associated with a Malware Subject.

Enumeration Value	Description
magic number	Refers to a configuration parameter that captures a file signature that may be used to identify or validate the content the malware instance.
id	Refers to a configuration parameter that captures an identifier for the malware instance.
group id	Refers to a configuration parameter that captures an identifier for a collection of malware instances.
mutex	Refers to a configuration parameter that captures a unique mutex value associated the malware instance.
filename	Refers to a configuration parameter that captures the name of a malicious binary such as one that is downloaded or embedded within the malware instance.
installation path	Refers to a configuration parameter that captures a location on disk to which the malware instance is installed, copied, or moved.

2.7.2 MalwareDevelopmentToolVocab-1.0

The `MalwareDevelopmentToolVocab` is the default MAEC vocabulary for the tool types used in the development of the malware instance characterized by the Malware Subject. The type of a tool is captured in the `Type` field, a child of the `Tool` field (of type `cyboxCommon:ToolInformationType`), itself a child of the `Tools` field (of type `cyboxCommon:ToolsInformationType`). The `Tools` field is a child of the `Development_Environment` field (of type `MalwareDevelopmentEnvironmentType`) in a Malware Subject.

The MAEC `MalwareDevelopmentToolVocab-1.0` type extends `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Type` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:MalwareDevelopmentToolEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Malware Development Tool Types*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#MalwareDevelopmentToolVocab-1.0*.'

2.7.2.1 MalwareDevelopmentToolEnum-1.0

The `MalwareDevelopmentToolEnum` is a non-exhaustive enumeration of tool types associated with the development of malware instances characterized by Malware Subjects.

Enumeration Value	Description
builder	Specifies a malware builder tool (commonly used to mass-produce malware) that was used to generate the malware instance.
compiler	Specifies a compiler tool that was used to compile the code composing the malware instance.
linker	Specifies a linker tool that was used to link the object files associated with the malware instance.
packer	Specifies a packer tool that was used to shrink the size of the executable binary associated with the malware instance. Packers are also sometimes referred to as 'compressors'.
crypter	Specifies a crypter tool that was used to encrypt the executable binary associated with the malware instance.
protector	Specifies a protector tool that was used to obfuscate the executable binary associated with the malware instance to make it more difficult to reverse engineer.

2.7.3 MalwareSubjectRelationshipTypeVocab-1.1

The MalwareSubjectRelationshipTypeVocab is the default MAEC vocabulary for the Malware Subject relationships in a MAEC Package, which are captured in Malware Subjects via the Type field of MalwareSubjectRelationshipType, defined in the MAEC Package schema.

The MAEC MalwareSubjectRelationshipTypeVocab-1.1 type extends ControlledVocabularyStringType defined in CybOX Common. Thus, Type fields that make use of this vocabulary are restricted to the enumerated entries contained in the maecVocabs:MalwareSubjectRelationshipTypeEnum-1.1; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	string	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Malware Subject Relationship Types*.'
vocab_reference	anyURI	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#MalwareSubjectRelationshipTypeVocab-1.1*.'

2.7.3.1 MalwareSubjectRelationshipTypeEnum-1.1

The MalwareSubjectRelationshipTypeEnum is a non-exhaustiveenumeration of relationships between Malware_Subjects.

Enumeration Value	Description
downloads	Specifies that the Malware Subject downloads one or more other Malware Subject (s).
downloaded by	Specifies that the current Malware Subject was downloaded by one or more other Malware Subject(s).
drops	Specifies that the Malware Subject drops (or writes to disk) one or more other Malware Subject(s).
dropped by	Specifies that the current Malware Subject was dropped (or written to disk) by one or more other Malware Subject(s).
extracts	Specifies that the Malware Subject extracts (from an embedded archive or another container) one or more other Malware Subject(s).
extracted from	Specifies that the current Malware Subject was extracted from one or more other Malware Subject(s).
direct descendant of	Specifies that the current Malware Subject is a direct descendant (i.e. in terms of development lineage)

	of one or more other Malware Subject(s).
direct ancestor of	Specifies that the current Malware Subject is a direct ancestor (i.e. in terms of development lineage) of one or more other Malware Subject(s).
memory image of	Specifies that the current Malware Subject represents a memory image associated with one or more other Malware Subject(s).
contained in memory image	Specifies that the current Malware Subject is a malware binary or component contained in one or more other Malware Subject(s) that represent memory images.
disk image of	Specifies that the current Malware Subject represents a disk image associated with one or more other Malware Subject(s).
contained in disk image	Specifies that the current Malware Subject is a malware binary or component contained in one or more other Malware Subject(s) that represent disk images.
network traffic capture of	Specifies that the current Malware Subject represents captured network traffic associated with one or more other Malware Subject(s).
contained in network traffic capture	Specifies that the current Malware Subject is a malware binary or component contained in one or more other Malware Subject(s) that represent captures of network traffic.
packed version of	Specifies that the current Malware Subject represents a packed version (in terms of executable binary packing) of one or more other Malware Subject(s).
unpacked version of	Specifies that the current Malware Subject represents an unpacked version (in terms of executable binary packing) of one or more other Malware Subject(s).
installs	Specifies that the current Malware Subject installs one or more other Malware Subject(s).
installed by	Specifies that the current Malware Subject is installed by one or more other Malware Subject(s).
64-bit version of	Specifies that the current Malware Subject is a 64-bit version of one or more other Malware Subject(s).
32-bit version of	Specifies that the current Malware Subject is a 32-bit version of one or more other Malware Subject(s).
encrypted version of	Specifies that the current Malware Subject is an encrypted version of one or more other Malware Subject(s).
decrypted version of	Specifies that the current Malware Subject is a decrypted version of one or more other Malware Subject(s).

2.8 Package-Related Default Vocabularies

The default vocabularies in this section are related to MAEC Packages.

2.8.1 GroupingRelationshipTypeVocab-1.0

The `GroupingRelationshipTypeVocab` is the default MAEC vocabulary for the grouping relationship types in a MAEC Package, which are captured in `Grouping_Relationships` fields via the `Type` field of `GroupingRelationshipType` defined in the MAEC Package schema.

The MAEC `GroupingRelationshipActionNameVocab-1.0` type extends `ControlledVocabularyStringType` defined in CybOX Common. Thus, `Type` fields that make use of this vocabulary are restricted to the enumerated entries contained in the `maecVocabs:GroupingRelationshipTypeEnum-1.0`; extended fields are shown below.

Field	Type	Multiplicity	Description
vocab_name	`string`	0..1	Specifies the name of the vocabulary. The fixed value is '*MAEC Default Grouping Relationship Types*.'
vocab_reference	`anyURI`	0..1	Specifies the URI associated with the vocabulary. The fixed value is '*https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd#GroupingRelationshipTypeVocab-1.0*.'

2.8.1.1 GroupingRelationshipTypeEnum-1.0

The `GroupingRelationshipTypeEnum` is a non-exhaustive enumeration of Malware Subject grouping relationships.

Enumeration Value	Description
same malware family	Indicates that the Malware Subjects in the MAEC Package are all part of the same malware family.
clustered together	Indicates that the Malware Subjects in the MAEC Package were clustered together by some algorithm or other mechanism.
observed together	Indicates that the Malware Subjects in the MAEC Package were observed together, such as on a host system, in some archive, etc. Note that there may not be any relationship between the Malware Subjects beyond co-location.
part of intrusion set	Indicates that the Malware Subjects in the MAEC Package were found as part of the same malware intrusion set.
same malware toolkit	Indicates that the Malware Subjects in the MAEC Package were all created using the same malware toolkit, independent of toolkit version.

Appendix – References

References made in this document are listed below.

A.1 MAEC Documents

[MAEC$_{O}$] MAEC Overview
http://maec.mitre.org/about/docs/MAEC_Overview.pdf

[MAEC$_{S}$] Characterizing Malware with MAEC and STIX
http://maec.mitre.org/about/docs/Characterizing_Malware_MAEC_and_STIX_v1.0.pdf

[SPEC$_{B}$] MAEC Bundle Specification
http://maec.mitre.org/language/version4.1/MAEC_Bundle_Spec_v4_1.pdf

[SPEC$_{P}$] MAEC Package Specification
http://maec.mitre.org/language/version4.1/MAEC_Package_Spec_v2_1.pdf

[SPEC$_{C}$] MAEC Container Specification
http://maec.mitre.org/language/version4.1/MAEC_Container_Spec_v2_1.pdf

[SPEC$_{V}$] MAEC Default Vocabularies Specification
http://maec.mitre.org/language/version4.1/MAEC_Vocabs_Spec_v1_1.pdf

[REQ] Requirements and Recommendations for MAEC Compatibiity
http://maec.mitre.org/compatible/Requirements_for_MAEC_Compatibility_V1.1.pdf

A.2 MAEC Web Pages

[EXAM$_{W}$] MAEC v4.1 Release Examples
http://maec.mitre.org/language/version4.1/#samples

[EXAM$_{G}$] MAEC Examples (GitHub repository)
https://github.com/MAECProject/schemas/tree/master/examples

[MAEC] MAEC Web Site
https://maec.mitre.org

[MAEC$_{C}$] MAEC Community
https://maec.mitre.org/community/index.html

[MAEC$_{L}$] MAEC Discussion List Signup
http://maec.mitre.org/community/discussionlist.html

[MAEC$_{H}$] MAEC Handshake (send email to maec@mitre.org for access)
https://handshake.mitre.org/

[REL4] MAEC v4.1 Release
https://maec.mitre.org/language/version4.1/

[TERM] MAEC Terminology
http://maec.mitre.org/about/terminology.html

[TIES] Ties to Existing Standards
http://maec.mitre.org/about/standards.html

[FAQ] MAEC FAQ
http://maec.mitre.org/about/faqs.html

[TOU] MAEC Terms of Use
https://maec.mitre.org/about/termsofuse.html

[VER] Versioning Policy
http://maec.mitre.org/language/versioning_policy.html

A.3 MAEC Schema

[REL$_{B}$] MAEC Bundle Model
https://maec.mitre.org/language/version4.1/maec_bundle_schema.xsd

[REL$_{P}$] MAEC Package Model
https://maec.mitre.org/language/version4.1/maec_package_schema.xsd

[REL$_{C}$] MAEC Container Model
https://maec.mitre.org/language/version4.1/maec_container_schema.xsd

[REL$_{D}$] MAEC Default Vocabularies
https://maec.mitre.org/language/version4.1/maec_default_vocabularies.xsd

A.4 MAEC Development

[DEV] MAEC GitHub Repositories
https://github.com/MAECProject/

[DEV$_P$] MAEC Python Library
https://github.com/MAECProject/python-maec

[DEV$_S$] MAEC Schema Development
https://github.com/MAECProject/schemas

[DEV$_U$] MAEC Utilities
https://github.com/MAECProject/utils

A.5 Other References

[CPE] Common Platform Enumeration (CPE)
http://nvd.nist.gov/cpe.cfm (Official CPE Dictionary)
http://csrc.nist.gov/publications/PubsNISTIRs.html (CPE Specifications)

[CUCKOO] Cuckoo Sandbox
http://www.cuckoosandbox.org/

[CVE] Common Vulnerabilities and Exposures (CVE)
http://cve.mitre.org

[CVSS] Common Vulnerability Scoring System
http://www.first.org/cvss

[CYBOX] Cyber Observable eXpression (CybOX)
http://cybox.mitre.org

[IOC] Open Indicators of Compromise (OpenIOC)
http://openioc.org/

[MMDEF] IEEE ICSG's Malware Metadata Exchange Format
http://standards.ieee.org/develop/indconn/icsg/mmdef.html

[OVAL] Open Vulnerability and Assessment Language (OVAL)
http://oval.mitre.org

[RFC2119] RFC 2119 – Key words for use in RFCs to Indicate Requirement Levels
http://www.ietf.org/rfc/rfc2119.txt

[STIX] Structured Threat Information eXpression (STIX)
http://stix.mitre.org

[W3C$_0$] W3C Namespaces in XML 1.0 (Third Edition)
http://www.w3.org/TR/REC-xml-names/

[W3C$_1$] W3C Recommendation for Hex-Encoded Binary Data
http://www.w3.org/TR/xmlSchema-2/#hexBinary

[W3C$_2$] W3C Recommendation for Boolean Data
http://www.w3.org/TR/xmlSchema-2/#boolean

[W3C$_3$] W3C Recommendation for Double Data
http://www.w3.org/TR/xmlschema-2/#double

[W3C$_4$] W3C Recommendation for Float Data
http://www.w3.org/TR/xmlSchema-2/#float

[W3C$_5$] W3C Recommendation for Integer Data
http://www.w3.org/TR/xmlSchema-2/#integer

[W3C$_6$] W3C Recommendation for XML Qualified Names
http://www.w3.org/TR/xmlSchema-2/#QName

[W3C$_7$] W3C Recommendation for String Data
http://www.w3.org/TR/xmlSchema-2/#string

[W3C$_8$] W3C Recommendation for unsigned int Data
http://www.w3.org/TR/xmlschema-2/#unsignedInt

[W3C$_9$] W3C Recommendation for URI Data
http://www.w3.org/TR/xmlschema-2/#anyURI