

SACM Working Group

Internet-Draft

Intended status: Informational National Institute of Standards and Techno

Expires: April 29, 2018

D. Waltermire

S. Banghart

October 26, 2017

Definition of the ROLIE Software Descriptor Extension draft-ietf-sacm-rolie-softwaredescriptor-00

Abstract

This document extends the Resource-Oriented Lightweight Information Exchange (ROLIE) core to add the information type category and related requirements needed to support Software Record and Software Inventory use cases. The 'software-descriptor' information type is defined as a ROLIE extension. Additional supporting requirements are also defined that describe the use of specific formats and link relations pertaining to the new information type.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

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

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

This Internet-Draft will expire on April 29, 2018.

Copyright Notice

Copyright (c) 2017 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in [Section 4](#).e of

the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1.	Introduction	2
2.	Terminology	3
3.	Additional Requirements for the Atom Publishing Protocol . .	3
4.	Additional Requirements for the Atom Syndication Format . . .	4
5.	Information-type Extensions	4
5.1.	The "software-descriptor" information type	4
5.2.	Use of the rolie:format element	5
5.2.1.	The ISO SWID 2016 format	5
5.2.2.	The Concise SWID format	5
6.	rolie:property Extensions	6
6.1.	urn:ietf:params:rolie:property:swd:id	6
6.2.	urn:ietf:params:rolie:property:swd:swname	6
7.	atom:link Extensions	6
8.	Other Registered Extensions	7
9.	IANA Considerations	7
9.1.	software-descriptor information-type	7
9.2.	swd:id property	8
9.3.	swd:swname property	8
10.	Security Considerations	8
11.	Privacy Considerations	9
12.	Normative References	9
Appendix A.	Schema	9
Appendix B.	Examples of Use	9
	Authors' Addresses	10

[1.](#) Introduction

This document defines an extension to the Resource-Oriented Lightweight Information Exchange (ROLIE) protocol to support the publication of software descriptor information. Software descriptor information is information that characterizes:

an installable software package, or

information about static software components that may be installed by a software package or patch.

Software descriptor information includes identifying, versioning, software creation and publication, and file artifact information. Software descriptor information provides data about what might be installed, but doesn't describe where or how a specific software installation is installed, configured, or executed.

Some possible use cases for Software descriptor information include:

- o Software providers can publish software descriptor information so that software researchers and users of software can understand the collection of software produced by a that software provider.
- o Organizations can aggregate and syndicate collections of software descriptor information provided by multiple software providers to support software-related analysis processes (e.g., vulnerability analysis) and value added information (e.g., software configuration checklist repositories) using identification and characterization information derived from software descriptor information.
- o End user organizations can consume sources of software descriptor information, and other related software vulnerability and configuration information to provide the data needed to automate software asset, patch, and configuration management practices.
- o Organizations can use software descriptors to support verification of other entities, thru mechanisms such as RIM or other integrity measurements.

This document supports these use cases by describing the content requirements for Collections of software descriptor information that are to be published to or retrieved from a ROLIE repository. This document also discusses requirements around the use of link relationships and describing the data model formats used in a ROLIE Entry describing a software descriptor information resource.

2. Terminology

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

Definitions for some of the common computer security-related terminology used in this document can be found in [Section 2 of \[RFC5070\]](#).

3. Additional Requirements for the Atom Publishing Protocol

This document does not specify any additional requirements for use of the Atom Publishing Protocol.

4. Additional Requirements for the Atom Syndication Format

This document does not specify any additional requirements on the Atom Syndication Format.

5. Information-type Extensions

This document defines the following information type[s]:

5.1. The "software-descriptor" information type

The "software-descriptor" information type represents any information that describes a piece of software. This document uses the definition of software provided by [[RFC4949](#)]. Note that as per this definition, this information type pertains to static software, that is, code on the disc. The software-descriptor information type is intended to provide a category for information that does one or more of the following:

identifies and characterizes software This software identification and characterization information can be provided by a large variety of data, but always describes software in a pre-installed state.

provides software installer metadata This represents information about software used to install other software. This metadata identifies, and characterizes a software installation package or media.

describes stateless installation metadata Information that describes the software post-deployment, such as files that may be deployed during an installation. It is expected that this metadata is produced generally for a given installation, and may not exactly match the actual installed files on a given endpoint.

Provided below is a non-exhaustive list of information that may be considered to be of a software-descriptor information type.

- o Naming information: IDs and names that aid in the identification of a piece of software
- o Version and patching information: Version numbers, patch identifiers, or other information that
- o Vendor and source information: Includes where the software was developed or distributed from, as well as where the software installation media may be located.

- o Payload and file information: information that describes or enumerates the files and folders that make up the piece of software, and information about those files.
- o Descriptive information and data: Any information that otherwise characterizes a piece of software, such as libraries, runtime environments, target OSes, intended purpose or audience, etc.

Note again that this list is not exhaustive, any information that in is the abstract realm of an incident should be classified under this information-type.

This information type does not include descriptions of running software, or state and configuration information that is associated with a software installation.

5.2. Use of the `rolie:format` element

This document does not contain any additional requirements for the `rolie:format` element, the formats that follow are provided as examples of formats that describe the software descriptor information type.

5.2.1. The ISO SWID 2016 format

The ISO SWID Tag 2016 format is a software descriptor and software record data format. It provides several tags: `primary`, which provides descriptive and naming information about software, `patch`, which describes non-standalone software meant to patch existing software, and `corpus`, which describes the software installation media that installs a given piece of software.

For a more complete overview as well as normative requirements, refer to `TODO(ref?):ISO/IEC 19770-2`

5.2.2. The Concise SWID format

The Concise SWID format is an alternative representation of the ISO SWID Tag 2016 format using a CBOR encoding defined by a CDDL specification. It provides the same features and attributes as are specified in ISO 19770-2, plus:

- o a straight forward method to sign and encrypt SWID Tags using COSE, and
- o additional attributes that provide an improved structure to include file hashes intended to be used as Reference Integrity Measurements (RIM).

6. rolie:property Extensions

This document registers new valid rolie:property names as follows:

6.1. urn:ietf:params:rolie:property:swd:id

This property provides an exposure point for an identification field from the associated software descriptor. The value of this property SHOULD be uniquely identifying information generated from the software descriptor linked to by the entry's atom:content element. swd:id property values SHOULD have a one-to-one mapping to individual pieces of SWD content.

6.2. urn:ietf:params:rolie:property:swd:sname

This property provides an exposure point for the plain text name of the software being described. Due to the great variance in naming schemes, this property should be considered informative.

7. atom:link Extensions

This section defines the requirements around the implementation of link relationships in the atom:link element in Entries. Each relationship is named, described, and given a requirement level.

Name	Description	Conformance
ancestor	Links to a software descriptor resource that defines an ancestor of the software being described by this Entry.	MAY
patches	Links to a software descriptor resource that defines the software being patched by this software	MAY
requires	Links to a software descriptor resource that defines a piece of software required for this software to function properly.	MAY
installs	Links to a software descriptor resource that defines the software being installed by this software.	MAY
installationrecord	Provides a link to a resource that describes an installation of this software.	MAY

Table 1: Link Relations for Resource-Oriented Lightweight Indicator Exchange

8. Other Registered Extensions

There are no additional extensions to define.

9. IANA Considerations

9.1. software-descriptor information-type

IANA has added an entry to the "ROLIE Security Resource Information Type Sub-Registry" registry located at <https://www.iana.org/assignments/rolie/category/information-type> .

The entry is as follows:

name: software-descriptor

index: TBD

reference: This document, [Section 5.1](#)

9.2. swd:id property

IANA has added an entry to the "ROLIE URN Parameters" registry located in <<https://www.iana.org/assignments/rolie/>>.

The entry is as follows:

name: property:swd:id

Extension IRI: urn:ietf:params:rolie:property:swd:id

Reference: This document, [Section 6.1](#)

Subregistry: None

9.3. swd:sname property

IANA has added an entry to the "ROLIE URN Parameters" registry located in <<https://www.iana.org/assignments/rolie/>>.

The entry is as follows:

name: property:swd:sname

Extension IRI: urn:ietf:params:rolie:property:swd:sname

Reference: This document, [Section 6.2](#)

Subregistry: None

10. Security Considerations

Use of this extension implies dealing with the security implications of both ROLIE and of software descriptors in general. As with any SWD information, care should be taken to verify the trustworthiness and veracity of the descriptor information to the fullest extent possible.

Ideally, software descriptors should have been signed by the software manufacturer, or signed by whichever agent processed the source code. SWD documents from these sources are more likely to be accurate than those generated by scraping installed software.

These "authoritative" sources of SWD content should consider additional security for their ROLIE repository beyond the typical recommendations, as the central importance of the repository is likely to make it a target.

Version information is often represented differently across manufacturers and even across product releases. If using SWD version information for low fault tolerance comparisons and searches, care should be taken that the correct version scheme is being utilized.

11. Privacy Considerations

This extension does not introduce any privacy considerations above or beyond that of the core ROLIE document. Any implementations using this extension should understand the privacy considerations of ROLIE and the Atom Publishing Protocol.

12. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC4949] Shirey, R., "Internet Security Glossary, Version 2", FYI 36, [RFC 4949](#), DOI 10.17487/RFC4949, August 2007, <<https://www.rfc-editor.org/info/rfc4949>>.
- [RFC5070] Danyliw, R., Meijer, J., and Y. Demchenko, "The Incident Object Description Exchange Format", [RFC 5070](#), DOI 10.17487/RFC5070, December 2007, <<https://www.rfc-editor.org/info/rfc5070>>.

Appendix A. Schema

This document does not require any schema extensions.

Appendix B. Examples of Use

Use of this extension in a ROLIE repository will not typically change that repository's operation. As such, the general examples provided by the ROLIE core document would serve as examples. Provided below is a sample SWD ROLIE entry:


```
<?xml version="1.0" encoding="UTF-8"?>
<entry xmlns="http://www.w3.org/2005/Atom"
  xmlns:rolie="urn:ietf:params:xml:ns:rolie-1.0">
  <id>dd786dba-88e6-440b-9158-b8fae67ef67c</id>
  <title>Sample Software Descriptor</title>
  <published>2015-08-04T18:13:51.0Z</published>
  <updated>2015-08-05T18:13:51.0Z</updated>
  <summary>A descriptor for a piece of software published by this
  organization. </summary>
  <link rel="self" href="http://www.example.org/provider/SWD/123456"/>
  <category
    scheme="urn:ietf:params:rolie:category:information-type"
    term="software-descriptor"/>
  <rolie:format ns="urn:example:COSWID"/>
  <content type="application/xml"
    src="http://www.example.org/provider/SWD/123456/data"/>
</entry>
```

Authors' Addresses

David Waltermire
National Institute of Standards and Technology
100 Bureau Drive
Gaithersburg, Maryland 20877
USA

Email: david.waltermire@nist.gov

Stephen Banghart
National Institute of Standards and Technology
100 Bureau Drive
Gaithersburg, Maryland 20877
USA

Email: stephen.banghart@nist.gov

