         A SUIT Manifest Extension for Concise Software Identifiers
                 draft-birkholz-suit-coswid-manifest-00

Abstract

   This document defines a resource extension for Concise Software
   Identifiers (CoSWID) that represents a SUIT firmware manifest. This
   extension combines the information elements of the SUIT information
   model with the semantic expressiveness of Software Identifiers. In
   consequence, this composite enables the integration of Firmware
   Updates for the Internet of Things (SUIT) in existing work-flows for
   updates of software components in general.

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 January 19, 2019.

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

Table of Contents

1. Introduction

   Firmware updates are quite similar to software update of applications
   - composites of software components - for example, servers or user-
   devices. Attributes and semantic dependencies as defined by Concise
   Software Identifies [I-D.ietf-sacm-coswid] are equivalent. In
   contrast, location and target definitions as well as the
   characteristics that are specific to an update campaign of a SUIT
   require a specific set of addiction information elements. The
   semantics regarding SUIT specific information elements are defined by
   the SUIT Information Model [I-D.ietf-suit-information-model].
   Correspondingly, the CoSWID extension defined in this document uses
   CDDL extension points to add SUIT information elements to the
   standard Concise Software Identifiers.

   The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
   "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and
   "OPTIONAL" in this document are to be interpreted as described in
   BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all
   capitals, as shown here.

2. SUIT Manifest Extension

   The following CDDL data definition is intended to be used as an
   extension to the CoSWID CDDL data definition. Corresponding terms,
   such as Resources, Processors and Targets of input nodes and output
   nodes are covered by the SUIT Information Model and Architecture
   [I-D.ietf-suit-architecture].

3. SUIT Manifest Resource Data Definition

   The following CDDL data definition MUST be added to Payload or
   Evidence Resources via CDDL extension points in order to express SUIT
   Manifests in Concise Software Identifiers.

```
$$payload-extension  //= (suit.manifest-entry,)
$$evidence-extension  //= (suit.manifest-entry,)

suit-manifest = {
  suit.manifest-version,
  suit.digest-info,
  suit.text-reference,
  suit.nonce,
  suit.sequence-number,
  ? suit.pre-condition,
  ? suit.post-condition,
  ? suit.directives,
  ? suit.resources,
  ? suit.processors,
  ? suit.targets,
  ? suit.extensions,
}

suit.manifest-entry = (59: suit-manifest / [ 2* suit-manifest ] )
suit.manifest-version = (60: 1)
suit.digest-info = (61: [ suit.digest-algorithm,
                          ? suit.digest-parameters,
                        ]
               )
suit.digest-algorithm = uint
suit.digest-parameters = bytes
suit.text-reference = (62: bytes)
suit.nonce = (63: bytes)
suit.sequence-number = (64: uint)
suit.pre-condition = (suit.id-condition // suit.time-condition // suit.image-con
dition // suit.custom-condition)
suit.post-condition = (suit.image-condition // suit.custom-condition)
suit.id-condition = (65: [ + [ suit.vendor / suit.class / suit.device,
                               suit.uuid,
                             ]
                         ]
                 )
suit.vendor = 0
suit.class = 1
suit.device = 2
suit.uuid = bstr .size 16
suit.time-condition = (66: [ + [ suit.install-after / suit.best-before,
                                 suit.timestamp,
```

```
                          ]
                        ]
                   )
suit.install-after = 0
suit.best-before = 1
suit.timestamp = uint .size 8
suit.image-condition = (67: [ + [ suit.current-content / suit.not-current-conten
t,
                                  suit.storage-identifier,
                                  ? suit.digest,
                                ]
                        ]
                   )
suit.current-content = 0
suit.not-current-content = 1
suit.digest = bytes
suit.storage-identifier = bytes
suit.custom-condition = (68: [ nint,
                               suit.condition-parameters,
                             ]
                   )
suit.condition-parameters = bytes
suit.directives = (69: { + int => bytes } )
suit.resources = (70: [ + [ suit.resource-type,
                            suit.uri-list,
                            suit.digest,
                            suit.onode,
                            ? suit.size,
                          ]
                      ]
                )
suit.resource-type = suit.payload / suit.dependency / suit.key / suit.alias
suit.payload = 0
suit.dependency = 1
suit.key = 2
suit.alias = 3
suit.uri-list = { + int => text }
suit.size = uint
suit.onode = bytes
suit.processors = (71: [ + [ suit.decrypt / suit.decompress / suit.undiff / suit
.relocate / suit.unrelocate,
                             suit.parameters,
                             suit.inode,
                             suit.onode,
                           ]
                       ]
                 )
suit.decrypt = 0
suit.decompress = 1
suit.undiff = 2
```

```
suit.relocate = 3
suit.unrelocate = 4
suit.parameters = bytes
suit.inode = bytes
suit.targets = (72: [ + [ suit.component-id,
                          suit.storage-identifier,
                          suit.inode,
                          ? suit.encoding,
                        ]
                  ]
              )
suit.component-id = [ + bytes ]
suit.encoding = bytes
suit.extensions = (73: { + int => bytes } )
```

4. Future Updates

    This draft is intended to incorporate the extension registry that
    will be defined by Concise Software Identifiers. Until then, a
    consecutive numbering system in alignment to the labels used in
    Concise Software Identifiers is applied.

5. Security Considerations

    TBD

6. Acknowledgments

    TBD

7. Change Log

    Initial Contribution

8. Contributors

    TBD

9. Normative References

    [I-D.ietf-sacm-coswid]
              Birkholz, H., Fitzgerald-McKay, J., Schmidt, C., and D.
              Waltermire, "Concise Software Identifiers", draft-ietf-
              sacm-coswid-06 (work in progress), July 2018.

   [I-D.ietf-suit-architecture]
             Moran, B., Meriac, M., Tschofenig, H., and D. Brown, "A
             Firmware Update Architecture for Internet of Things
             Devices", draft-ietf-suit-architecture-01 (work in
             progress), July 2018.

   [I-D.ietf-suit-information-model]
             Moran, B., Tschofenig, H., and H. Birkholz, "Firmware
             Updates for Internet of Things Devices - An Information
             Model for Manifests", draft-ietf-suit-information-model-01
             (work in progress), July 2018.

   [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>.

   [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC
             2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174,
             May 2017, <https://www.rfc-editor.org/info/rfc8174>.

Author's Address

   Henk Birkholz
   Fraunhofer SIT
   Rheinstrasse 75
   Darmstadt 64295
   Germany

   Email: henk.birkholz@sit.fraunhofer.de