SUIT
Internet-Draft
Intended status: Standards Track
Expires: 14 September 2023

B. Moran
H. Tschofenig
Arm Limited

13 March 2023

Strong Assertions of IoT Network Access Requirements
draft-ietf-suit-mud-03

Abstract

The Manufacturer Usage Description (MUD) specification describes the access and network functionality required for a device to properly function. The MUD description has to reflect the software running on the device and its configuration. Because of this, the most appropriate entity for describing device network access requirements is the same as the entity developing the software and its configuration.

A network presented with a MUD file by a device allows detection of misbehavior by the device software and configuration of access control.

This document defines a way to link a SUIT manifest to a MUD file offering a stronger binding between the two.

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 14 September 2023.

Copyright Notice

Table of Contents

1. Introduction

Under [RFC8520], devices report a URL to a MUD manager in the network. RFC 8520 envisions different approaches for conveying the information from the device to the network such as:

* DHCP,

* IEEE802.1AB Link Layer Discovery Protocol (LLDP), and

* IEEE 802.1X whereby the URL to the MUD file would be contained in the certificate used in an EAP method.

The MUD manager then uses the ~~the~~ URL to fetch the MUD file, which contains access and network functionality required for a device to properly function.

The MUD manager must trust the service from which the ~~URL~~ MUD file is fetched ~~and~~ to return an authentic copy of the MUD file. This concern may be mitigated using the optional signature reference in the MUD file. The MUD manager must also trust the device to report a correct URL. In case of DHCP and LLDP the URL is unprotected. When the URL to the MUD file is included in a certificate then it is authenticated and integrity protected. A certificate created for use with network access authentication is typically not signed by the entity that wrote the software and configured the device, which leads to conflation of local network access rights with rights to assert all network access requirements.

There is a need to bind the entity that creates the software and configuration to the MUD file because only that entity can attest the network access requirements of the device. This specification defines an extension to the SUIT manifest to include a MUD file (~~per~~by reference or by value). When combining a ~~manufacturer usage description~~MUD with a manifest used for software/firmware updates (potentially augmented with attestation) then a network operator can get more confidence in the description of the access and network functionality required for a device to properly function.

2. Terminology

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.

3. Workflow

The intended workflow is as follows:

* At the time of onboarding, devices report their SUIT manifests in use to the MUD manager.

* If the SUIT_MUD_container (see Section 5) has been severed, the MUS manager can use the suit-reference-uri ~~can be used~~ to retrieve the complete SUIT manifest.

* The manifest authenticity is verified by the MUD manager, which enforces that the MUD file presented is also authentic and as intended by the device software vendor.

Commented [DT1]: Why? That is, why does the service need to trust the client for anything?

Commented [DT2]: What is typical? Is there a reference that can be cited?

Commented [DT3]: Each device can have any number of SUIT manifests in use (e.g., one per "app"/service), so this should be plural.

Commented [DT4]: Not correct. The device software vendor might not be the entity responsible for the configuration, only the software. It has to know it's as intended by the *configuration owner.*

* Each time a device is updated, rebooted, or otherwise substantially changed, it will execute an attestation.
  - Among other claims in the Entity Attestation Token (EAT) [I-D.ietf-rats-eat], the device will report its software digest(s), configuration digest(s), primary manifest URI, and primary manifest digest to the MUD manager.
  - The MUD manager can then validate these attestation reports in order to check that the device is operating with the expected version of software and configuration.
  - Since the manifest digest is reported, the MUD manager can look up the corresponding manifest.
* If the MUD manager does not already have a full copy of the manifest, it can be acquired using the reference URI.
* Once a full copy of the manifest is provided, the MUD manager can verify the device attestation report
* The MUD manager acquires the MUD file from the MUD ~~url~~URL.
* The MUD manager verifies the MUD file signature using the provided Subject Key Identifier.
* Then, the MUD manager can apply any appropriate policy as described by the MUD file.

4. Advantages over previous MUD ~~url~~ URL reporting mechanisms

Binding within the firmware manifest has several advantages over other MUD ~~url~~ URL reporting mechanisms:

* The MUD ~~url~~ URL is tightly coupled to device firmware version.
* The device does not report the ~~url~~URL, so the device cannot tamper with the ~~url~~URL.
* The onus is placed on the software author to provide a MUD file that describes their device.
* The Manifest Author (software signer) explicitly authorizes a key to sign MUD files, providing a tight coupling between the party that knows device behavior best (the manifest author) and the party that declares device behavior (MUD file signer).

Commented [DT5]: Undefined term. What's this? If a device has 2 apps installed on top of a common OS/firmware, each of which has its own SUIT manifest, what does "primary" mean?

Commented [DT6]: Why firmware rather than software? If the firmware and software have different manifests, I think this is wrongly worded. This gets to the problem with the word "primary" above.

Commented [DT7]: Maybe, but the software author can't provide one that reflects a specific configuration by the owner. So shouldn't the device owner provide a more constrained MUD file?

* Network operators do not need to know, a priori, which MUD ~~url~~ URL to use for each device; this can be harvested from the device's manifest and only replaced if necessary.

* A network operator can still replace a MUD ~~url~~URL:

  - By providing a manifest that overrides the MUD ~~url~~URL.

  - By replacing the MUD ~~url~~ URL in network infrastructure.

* Devices can be quarantined if they do not attest a known software version.

* Devices cannot lie about which MUD ~~url~~ URL to use.

5. Extensions to SUIT

To enable strong assertions about the network access requirements that a device should have for a particular software/configuration pair a MUD ~~url~~ URL is added to a SUIT manifest along with a subject-key identifier, according to [RFC7093], mechanism 4 (the keyIdentifier is composed of the hash of the DER encoding of the SubjectPublicKeyInfo value). The Subject Key Identifier MUST be constructed with mechanism 4.

The following CDDL describes the extension to the SUIT_Manifest structure:

```
$$severable-manifest-members-choice-extensions //= (
  suit-manifest-mud => SUIT_Digest / SUIT_MUD_container
)
```

The SUIT_Envelope is also amended:

```
$$SUIT_severable-members-extensions //= (
  suit-manifest-mud => bstr .cbor SUIT_MUD_container
)

SUIT_MUD_container = {
    suit-mud-url => #6.32(tstr),
    suit-mud-ski => SUIT_Digest,
}
```

Commented [DT8]: Can't a device owner do the same thing too?

Commented [DT9]: Having a hard time parsing grammar around this phrase

Commented [DT10]: What does "ski" mean?

6. Security Considerations

This specification links MUD files to other IETF technologies, particularly to SUIT manifests, for improving security protection and ease of use. By including MUD files (~~per~~ by reference or by value) in SUIT manifests an extra layer of protection has been created and synchronization risks can be minimized. If the MUD file and the software/firmware loaded onto the device gets out-of-sync a device may be firewalled and, with firewalling by networks in place, the device may stop functioning.

7. IANA Considerations

suit-manifest-mud must be added as an extension point to the SUIT manifest registry.

Commented [DT11]: Please update to meet the requirements listed at https://www.iana.org/help/protocol-registration#registrations

8. Normative References

[I-D.ietf-rats-eat]
Lundblade, L., Mandyam, G., O'Donoghue, J., and C. Wallace, "The Entity Attestation Token (EAT)", Work in Progress, Internet-Draft, draft-ietf-rats-eat-19, 19 December 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-rats-eat-19>.

[I-D.ietf-suit-manifest]
Moran, B., Tschofenig, H., Birkholz, H., Zandberg, K., and O. Rønningstad, "A Concise Binary Object Representation (CBOR)-based Serialization Format for the Software Updates for Internet of Things (SUIT) Manifest", Work in Progress, Internet-Draft, draft-ietf-suit-manifest-22, 27 February 2023, <https://datatracker.ietf.org/doc/html/draft-ietf-suit-manifest-22>.

[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>.

[RFC7093] Turner, S., Kent, S., and J. Manger, "Additional Methods for Generating Key Identifiers Values", RFC 7093, DOI 10.17487/RFC7093, December 2013, <https://www.rfc-editor.org/info/rfc7093>.

[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>.

[RFC8520] Lear, E., Droms, R., and D. Romascanu, "Manufacturer Usage Description Specification", RFC 8520, DOI 10.17487/RFC8520, March 2019, <https://www.rfc-editor.org/info/rfc8520>.

Authors' Addresses

Brendan Moran
Arm Limited
Email: brendan.moran.ietf@gmail.com

Hannes Tschofenig
Arm Limited
Email: hannes.tschofenig@gmx.net