SUIT
Internet-Draft
Updates: draft-ietf-suit-manifest (if approved)
Intended status: Standards Track
Expires: 4 September 2025

B. Moran
Arm Limited
H. Tschofenig

3 March 2025

Strong Assertions of IoT Network Access Requirements
draft-ietf-suit-mud-10

Abstract

The Manufacturer Usage Description (MUD) specification describes the access and network functionality required for a device to properly function. This 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 the Software Updates for Internet of Things (SUIT) manifest to a MUD file offering a stronger binding between the two.

Table of Contents

1. Introduction

A Manufacturer Usage Description (MUD) file describes what sort of network communication behavior a device is designed to have. For example, a manufacturer may use a MUD file to describe that a device uses HTTP, DNS and NTP communication but no other protocols. The communication patterns are described in a JSON-based format in the MUD file.

The MUD files do, however, need to be presented by the device to a MUD manager in the operational network where the device is deployed. Under [RFC8520], devices report a MUD URL to a MUD manager in the operational network. The MUD URL is a URL that can be used by the MUD manager to receive the MUD file from a MUD file server to ultimately obtain the MUD file.

Figure 1 shows the MUD architecture, as defined in RFC 8520.

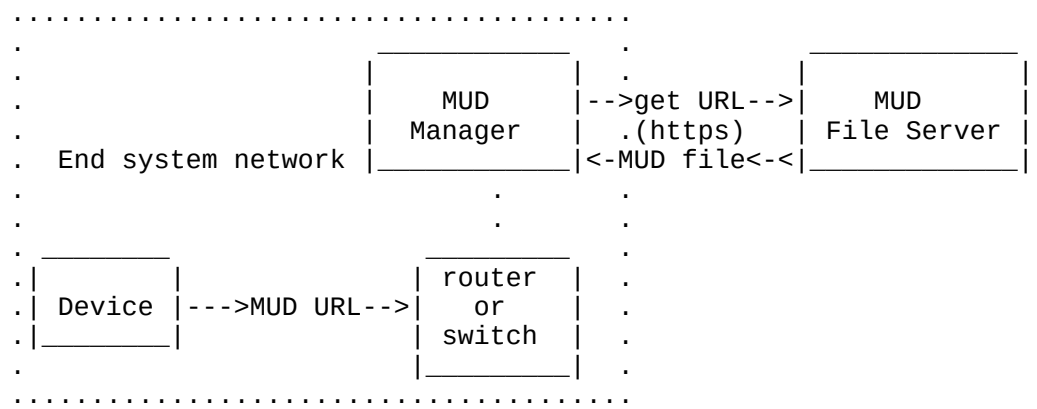

Figure 1: MUD Architecture per RFC 8520.

RFC 8520 envisions different approaches for conveying the MUD URL from the device to the operational network. Section 4 of [I-D.ietf-opsawg-mud-acceptable-urls] provides additional description of the MUD URLs sources, which include:

* DHCP,

* IEEE 802.1AB Link Layer Discovery Protocol (LLDP), and

* Device Certificates, such as IEEE 802.1X, whereby the URL to the MUD File would be contained in the certificate.

The MUD manager must trust the MUD file server from which the MUD file is fetched to return the most up-to-date MUD file. It must also trust the device to report the correct MUD URL. In case of DHCP and LLDP the URL is unprotected and not bound to the device itself.

When the MUD URL is included in a certificate then it is authenticated and integrity protected. However, the certificate only proves possession of a private key and endorsements by the certificate issuer. This does not prove what software is in use, nor does it prove that the MUD file is the correct file for the deployed software: instead, the responsibility falls on the certificate issuer to identify the MUD URL correctly and to supply a MUD Signer correctly. There is a need to bind the entity that creates the software and configuration to the MUD file. The developer is in the best position to describe the communication requirements of the software it developed and configured for a device.

This specification defines an extension to the Software Updates for Internet of Things (SUIT) manifest [I-D.ietf-suit-manifest] to include a MUD URL. A SUIT manifest is a bundle of metadata about code/data for an IoT device, where to find the code/data, the devices to which it applies, and cryptographic information protecting the manifest.

When combining a MUD URL with a manifest used for software/firmware updates then a network operator has more confidence in the description of the communication requirements 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.

This document re-uses the terms defined in [RFC9334] related to remote attestation. Readers of this document are assumed to be familiar with the following terms: Evidence, Claim, Attester, Verifier, and Relying Party (RP).

This document also uses terms defined in [RFC8520], such as MUD, MUD file, MUD manager, MUD URL, etc.

3. Workflow

Figure 2 shows the architectural extensions introduced by combining SUIT and MUD. The key elements are that the developer, who produces the firmware is also generating a manifest and the MUD file. Information about the MUD file is embedded into the SUIT manifest and provided to the device via firmware update mechanism. Once this information is available on the device it can be presented during device onboarding, during network access authentication, or as part of other interactions that involve the conveyance of Evidence to the operational network. After retrieving the manifest, the MUD file can be obtained as well.

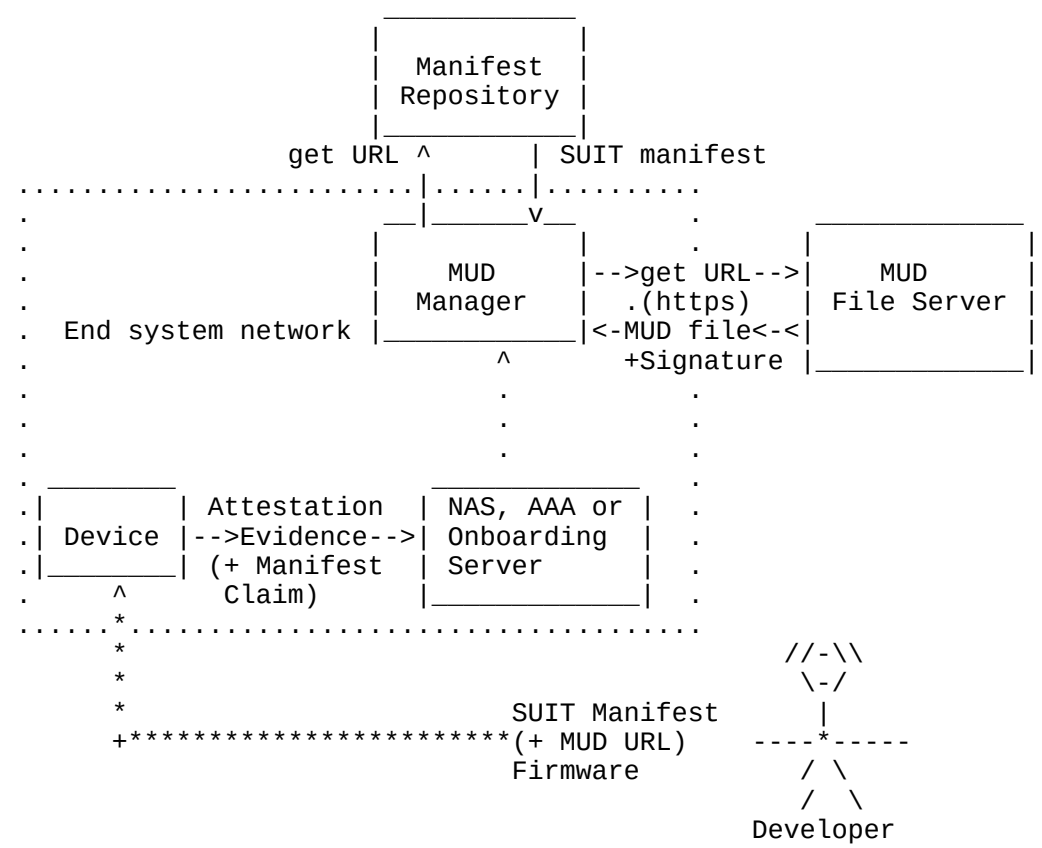

Figure 2: SUIT-MUD Architecture.

The intended workflow is as follows, and assumes an attestation mechanism between the device and the MUD Manager:

* At the time of onboarding, devices report their manifest in use to the MUD Manager via some form of attestation Evidence and a conveyance protocol. The device thereby acts as an Attester. The normative specification of these mechanisms is out of scope for this document.

* An example of an Evidence format is the Entity Attestation Token (EAT) [I-D.ietf-rats-eat], which offers a rich set of claims. This specification assumes that Evidence includes a link to the SUIT manifest via the "manifests" claim (see Section 4.2.15 of [I-D.ietf-rats-eat]) or that the manifest itself is embedded in the Evidence. This Evidence is conveyed to the operational network via some protocol, such as network access authentication

protocol (for example using the EAP-TLS 1.3 method [RFC9190] utilizing the attestation extensions [I-D.fossati-tls-attestation]) or an onboarding protocol like FIDO Device Onboard (FDO) [FDO] or Bootstrapping Remote Secure Key Infrastructure (BRSKI) [RFC8995].

* The MUD Manager, acting as a Relying Party, relays the Evidence to the Verifier and receives an Attestation Result in response. This allows the MUD Manager to check that the device is operating with the expected version of software and configuration.

* Since a URL to the manifest is contained in the Evidence, the MUD Manager can look up the corresponding manifest.

* The MUD Manager acquires the MUD file from the MUD URL found in the SUIT manifest. The SUIT manifest contains the MUD URL and not the MUD file primarily to due the size of the MUD file. This also allows the MUD file to be updated rapidly in response to evolving threats.

* The MUD Manager verifies the MUD file signature using the Subject Key Identifier (SKI) provided in the SUIT manifest.

* Then, the MUD Manager can apply any appropriate policy as described by the MUD file.

Each time a device is updated, rebooted, or otherwise substantially changed, it will execute the remote attestation procedures again.

4. Operational Considerations

This specification assumes that the software/firmware author provides a MUD file that describes the behavior of the software running on a device.

4.1. Pros

The approach described in this document has several advantages over the RFC 8520 MUD URL reporting mechanisms:

* The MUD URL is tightly coupled to device software/firmware version.

* The device does not report the MUD URL, so the device cannot tamper with the MUD URL.

* The author explicitly authorizes a key to sign MUD files, providing a tight coupling between the party that knows device behavior best (the author of the software/firmware) and the party that declares device behavior (MUD file signer).

* Network operators do not need to know, a priori, which MUD 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 in a SUIT manifest:

  - By providing a SUIT manifest that overrides the MUD URL.

  - By replacing the MUD URL in their network infrastructure.

* Devices can be quarantined if the Attestation Result indicates that an out-dated or compromised software/firmware version has been used.

* Devices cannot lie about which MUD URL to use.

4.2. Cons

This mechanism relies on the use of SUIT manifests to encode the MUD URL. Conceptually, the MUD file is similar to a Software Bill of Material (SBOM) but focuses on the external visible communication behavior, which is essential for network operators, rather than describing the software libraries contained within the device itself.

* MUD Manager must be aware of the Status Tracker or vice versa so that the MUD Manager can obtain MUD URLs and MUD Signer SKIs from the Status Tracker. This implies a new API in the MUD manager or Status Tracker.

* The MUD manager requires a failover mechanism to trigger the status tracker to obtain a copy of the SUIT Manifest in order to extract the MUD URL if it is not already aware of a device. This could be done, for example, as a part of an onboarding flow.

* Attestation Evidence may convey the SUIT Manifest, in which case the Status Tracker becomes a Relying Party since it depends on Attestation Evidence. This workflow is expected, however.

* This approach explicitly moves the decisions about device behaviour away from the Network Operator and towards the Manifest Author. While this is appropriate when the Manifest Author is trusted, not all IoT devices are fully trusted, and MUD files enable a Network Operator to restrict their capabilities. For a

Network Operator to override a device's manufacturer-provided MUD URL will require the MUD manager to have a mechanism to enable this override, which adds complexity

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 is added to the SUIT manifest along with a subject key identifier (ski). Note that the subject key identifier refers to a more generic version of SubjectPublicKeyInfo defined in [RFC5280], which refers to an X.509-based ski. The subject key identifier MUST be generated according to the process defined in [I-D.ietf-cose-key-thumbprint] and the SUIT_Digest structure MUST be populated with the selected hash algorithm and obtained fingerprint. The subject key identifier corresponds to the key used in the MUD signature file described in Section 13.2 of [RFC8520].

Note: A key need not be in COSE Key format to create a COSE Key Thumbprint of it.

The following Concise Data Definition Language (CDDL) [RFC8610] describes the extension to the SUIT_Manifest structure:

The extension to the SUIT_Manifest is described here:

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

The SUIT_MUD_container structure is defined as follows:

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

6. Security Considerations

This specification links MUD files to SUIT manifests for improving security protection and ease of use. By including MUD URLs in SUIT manifests an extra layer of protection has been created and synchronization risks can be minimized.

Used in this way, the MUD manager presents an additional layer of security on networks where they are enabled. The MUD manager configures the L2/L3 infrastructure of a Local Area Network to apply restrictive policies to certain devices. The MUD manager only has

the ability to elevate or restrict the network privileges of a device. Therefore, attacks on the MUD Manager cannot compromise devices, they can only enable a compromised device to access more of the network. Further security considerations related to the MUD Manager are covered in [RFC8520].

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. This is, however, not a concern specific to this specification but rather to the use of MUD in general. Below are two mitigations:

* A manufacturer must update the MUD file in advance of network service or product changes so that the new services can be supported. Because the MUD file is accessed by a URL means that it can be subsequently updated. This requires a MUD file being retrieved again. This handles the case when the device is already deployed and in use.

* There is a possibility that an IoT device has remained on-shelf inventory for an extended period, resulting in its MUD file being inaccessible at its previous location. This necessitates a decision on how to implement a fail-safe tailored to the particular environment.

7. IANA Considerations

IANA is requested to add a new value to the SUIT manifest elements registry created with [I-D.ietf-suit-manifest]:

* Label: TBD1 [[Value allocated from the standards action address range]]

* Name: Manufacturer Usage Description (MUD)

* Reference: [[TBD: This document]]

8. References

8.1. Normative References

[I-D.ietf-cose-key-thumbprint]
Isobe, K., Tschofenig, H., and O. Steele, "CBOR Object Signing and Encryption (COSE) Key Thumbprint", Work in Progress, Internet-Draft, draft-ietf-cose-key-thumbprint-06, 6 September 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-cose-key-thumbprint-06>.

[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-31, 6 September 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-rats-eat-31>.

[I-D.ietf-suit-manifest]
Moran, B., Tschofenig, H., Birkholz, H., Zandberg, K., and O. R淡nningstad, "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-33, 24 February 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-suit-manifest-33>.

[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/rfc/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/rfc/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/rfc/rfc8520>.

[RFC8610] Birkholz, H., Vigano, C., and C. Bormann, "Concise Data Definition Language (CDDL): A Notational Convention to Express Concise Binary Object Representation (CBOR) and JSON Data Structures", RFC 8610, DOI 10.17487/RFC8610, June 2019, <https://www.rfc-editor.org/rfc/rfc8610>.

[RFC9334] Birkholz, H., Thaler, D., Richardson, M., Smith, N., and W. Pan, "Remote ATtestation procedureS (RATS) Architecture", RFC 9334, DOI 10.17487/RFC9334, January 2023, <https://www.rfc-editor.org/rfc/rfc9334>.

8.2. Informative References

[FDO] FIDO Alliance, "FIDO Device Onboard Specification 1.1", April 2022, <https://fidoalliance.org/specifications/download-iot-specifications/>.

[I-D.fossati-tls-attestation]
Tschofenig, H., Sheffer, Y., Howard, P., Mihalcea, I., Deshpande, Y., Niemi, A., and T. Fossati, "Using Attestation in Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", Work in Progress, Internet-Draft, draft-fossati-tls-attestation-08, 21 October 2024, <https://datatracker.ietf.org/doc/html/draft-fossati-tls-attestation-08>.

[I-D.ietf-opsawg-mud-acceptable-urls]
Richardson, M., Pan, W., and E. Lear, "Authorized update to MUD URLs", Work in Progress, Internet-Draft, draft-ietf-opsawg-mud-acceptable-urls-12, 6 September 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-opsawg-mud-acceptable-urls-12>.

[RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, DOI 10.17487/RFC5280, May 2008, <https://www.rfc-editor.org/rfc/rfc5280>.

[RFC8995] Pritikin, M., Richardson, M., Eckert, T., Behringer, M., and K. Watsen, "Bootstrapping Remote Secure Key Infrastructure (BRSKI)", RFC 8995, DOI 10.17487/RFC8995, May 2021, <https://www.rfc-editor.org/rfc/rfc8995>.

[RFC9190] Preu Mattsson, J. and M. Sethi, "EAP-TLS 1.3: Using the Extensible Authentication Protocol with TLS 1.3", RFC 9190, DOI 10.17487/RFC9190, February 2022, <https://www.rfc-editor.org/rfc/rfc9190>.

Acknowledgements

We would like to thank Roman Danyliw for his excellent review as the responsible security area director, Bahcet Sarikaya for his Genart review, Michael Richardson for his IoT directorate review and Susan Hares for her Opsdir review. During the IESG review Robert Wilton, Eliot Lear, Zaheduzzaman Sarker, Francesca Palombini, John Scudder, Paul Wouters, ric Vyncke, and Murray Kucherawy.

Authors' Addresses

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

Hannes Tschofenig
Email: hannes.tschofenig@gmx.net