Workgroup: Network Working Group
Internet-Draft: draft-richardson-mud-qrcode-02
Published: 14 November 2021
Intended Status: Informational
Expires: 18 May 2022
Authors: M. Richardson
Sandelman Software Works
J. Latour
CIRA Labs
H. Habibi Gharakheili
UNSW Sydney

On loading MUD URLs from QR codes

Abstract

This informational document details a protocol to load MUD definitions for devices which have no integrated MUD (RFC8520) support.

This document is published to inform the Internet community of this mechanism to allow interoperability and to serve as a basis of other standards work if there is interest.

RFCEDITOR please remove: Pull requests and edit welcome at:

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 18 May 2022.

Copyright Notice

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

The Manufacturer Usage Description (MUD) [RFC8520] defines a YANG data model to express what sort of access a device requires to operate correctly. That document additionally defines three ways for the device to communicate the URL of the resulting JSON [RFC8259] format file to a network enforcement point: DHCP, within an X.509 certificate extension, and via LLDP.

Each of the above mechanism conveys the MUD URL in-band, and requires modifications to the device firmware. Most small IoT devices do not have LLDP, and often have very restricted DHCP clients. Adding the LLDP or DHCP options requires at least some minimal configuration change, and possibly entire new subsystems. Meanwhile, use of the PKIX certification extension only makes sense as part of a larger IDevID based [ieee802-1AR] deployment such as [I-D.ietf-anima-bootstrapping-keyinfra].

In the above cases these mechanisms can only be implemented by persons with access to modify and update the firmware of the device.

In the meantime there is a chicken or egg problem ([chickenegg]): no manufacturers include MUD URLs in their products as there are no gateways that use them. No gateways include code that processes MUD URLs as no products produce them.

The protocol described here allows any person with physical access to the device to affix a reference to a MUD URL that can later be scanned by an end user.

Such an action can be done by

- *the marketing department of the Manufacturer,
- *an outsourced assembler plant,
- *value added resellers (perhaps in response to a local RFP),
- *a company importing the product (possibly to comply with a local regulation),
- *a network administrator (perhaps before sending devices home with employees, or to remote sites),
- *a retailer as a value added service.

QRcodes are informally described in [qrcode] and formally defined in [isoiec18004]. The protocol described in this document uses a QRcode to encode the MUD URL. Specifically, the protocol leverages the data format from the Reverse Logistics Association's Standardized Quick Response for Logistics [SQRL].

SQRL codes are being put on devices via sticker or via laser etching into the case in order to deal with many situations, but specifically for end-of-life processing for the device. An important idea behind the effort is that clearly identifying a product permits appropriate disposal, refurbishment or recycling of the components of the product.

There are also use cases for SQRL described in which the codes are used as part of regular maintenance for a product.

SQRL is an application of the 12N Data Identifier system specified by the ANSI MH10.8.2 Committee [mh10] in a format appropriate for QRcodes as well as other things like NFCs transmissions.

QRcode generators are available as web services [qrcodewebservice], or as programs such as [qrencode].

Section 4 summarizes the considerations contained in [I-D.ietf-opsawg-mud-acceptable-urls] section 6.1 ("Updating MUD URLs vs

Updating MUD files"). Due to the immutable nature of the QRcode, MUD URLs in this document will need to non-firmware specific.

2. Terminology

Although this document is not an IETF Standards Track publication, it adopts the conventions for normative language to provide clarity of instructions to the implementer. 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.

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. Protocol

This QRcode protocol builds upon the work by [SQRL]. That protocol is very briefly described in the next section. Then the list of needed Data Records to be filled in is explained.

3.1. The SQRL Protocol

[SQRL] documents an octet protocol that can be efficiently encoded into QRcodes using a sequence of ASCII bytes, plus six control codes (see section 3.1 of [SQRL]):

*<RS> Record Separator (ASCII 30)

*<EoT> End of Transmission (ASCII 4)

*<FS> Field Separator (ASCII 28)

*<GS> Group Separator (ASCII 29)

*<US> Unit Separator (ASCII 31),

*Concatenation Operator (ASCII 43: "+").

Section 7.2 of [SQRL] gives the details, which can be summarized as:

1. The QR code header starts with:

"[)>" <RS> "06" <GS> "12N"

1. Include one or more Data Records. This consists of a four letter Field Identifiers followed by ASCII characters terminated with a <Unit Separator>.

2. End with:

<RS><EoT>

There are additionally optional flags that may be present in every Data Record as described in section 7.4 of [SQRL]. These flags have no bearing on MUD processing. A parser which is only collecting MUD URLs will not need to parse those flags. A general purpose SQRL parser will need more complexity.

Field Seperator characters are used in SQRL to signify the beginning of a new unit of data. A MUD specific parser that encounters a Field Seperator and has not yet collected the right MUD information MUST ignore the characters collected so far and then restart.

Environment records, as described in [SQRL] section 7.4, look and act exactly as fields, with a special Field Identifier. They serve no purpose when looking for MUD information, and MAY be ignored.

3.2. Manufacturer Usage Descriptions in SQRL

3.2.1. B000 Company Name

The B000 Data Record is mandatory in [SQRL]. It MUST be in ASCII representation. It should be a representation of the company or brand name. It SHOULD match the ietf-mud/mud/mfg-name in the MUD file, however the MUD file can contain arbitrary UTF8 for this name, while the SQRL files are expected to be 7-bit US-ASCII.

3.2.2. B001 Product Name

The B001 Data Record is optional in [SQRL]. It's presence is RECOMMENDED. Some third parties that create QRcode stickers might not know the product name with 100% certainty, and MAY prefer to omit this rather than create further confusion. It is the Product Name in ASCII.

3.2.3. B002 Model Number

The B002 Data Record is optional in [SQRL], but is MANDATORY in this profile. It is the Model Name in ASCII. It SHOULD match the optional ietf-mud/mud/model-name in the MUD file if that entry is present in the MUD file.

If a third party that is creating QRcodes can not locate an official model number when creating their MUD file and QRcode, then the third party SHOULD make one up.

3.2.4. MUD URL Data Record

A new Field Identifier has been assigned by the Reverse Logistics Association (RLA), which is "M180" This record MUST be filled with the MUD URL.

Short URLs are easier to encode into QRcode because they require fewer pixels of QRcode. More content in the QRcode requires a bigger image.

Use of URL shortening services (see [URLshorten]) can be useful provided that the service is stable throughout the lifetime of the device and QRcode, and that the privacy stance of the service is well understood.

Section 8.1 of [SQRL] also has some good advice on longevity concerns with URLs.

The URL provided MUST NOT have a query (?) portion present. If one is present, the query portion MUST be removed before processing.

3.2.5. MUD Device MAC Address

In order for the MUD controller to associate the above policy with a specific device, some unique identifier must be provided to the MUD controller. The most actionable identifier is the Ethernet MAC address. [SQRL] section 9.10 defines the Data Record: "M06C" as the MAC address. No format for the MAC address is provided in that document.

This document RECOMMENDS 12 (or 16) hex octets are used with no spaces or punctuation. (16 octets are used in the IEEE OUI-64 format used in 802.15.4, and some next generation Ethernet proposals)

Parsers that find punctuation (such as colons (":"), dashes ("-"), or white space) MUST skip over it.

4. Generic URL or Version Specific URL

MUD URLs which are communicated in-band by the device, and which are programmed into the device's firmware may provide a firmware specific version of the MUD URL. This has the advantage that the resulting Access Control Lists (ACLs) implemented are specific to the needs of that version of the firmware.

A MUD URL which is affixed to the device with a sticker, or etched into the case can not be changed.

Given the considerations of [I-D.ietf-opsawg-mud-acceptable-urls] section 6.1 ("Updating MUD URLs vs Updating MUD files"), it is prudent to use a MUD URL which points to a MUD file which will only have new features added over time, and never have features removed.

When the firmware eventually receives built-in MUD URL support, then a more specific URL may be used.

Note that in many cases it will be third parties who are generating these QRcodes, so the MUD file may be hosted by the third party.

5. Crowd Supply of MUD Files

At the time of writing, the IETF MUD is a new IETF Proposed Standard. Hence, IoT device manufacturers have not yet provided MUD profiles for their devices. A research group at the University of New South Wales (UNSW Sydney) has developed an open-source tool, called MUDgee ([MUDgee]), which automatically generates a MUD file (profile) for an IoT device from its traffic trace in order to make this process faster, easier, and more accurate. Note that the generated profile completeness solely depends on the completeness of the input traffic traces. MUDgee assumes that all the activity seen is intended and benign.

UNSW researchers have applied MUDgee about 30 consumer IoT devices from their lab testbed, and publicly released their MUD files ([MUDfiles]). MUDgee can assist IoT manufacturers in developing and verifying MUD profiles, while also helping adopters of these devices to ensure they are compatible with their organisational policies.

Similar processes have been done in a number of other public and private labs. One of the strong motivations for this specification is to allow for this work to leave the lab, to be applied in the field.

6. Privacy Considerations

The presence of the MUD URL in the QR code reveals the manufacturer of the device, the type or model of the device, and possibly the firmware version of the device.

The MAC address of the device will also need to be present, and this is potentially Personally Identifiable Information (PII). Such QRcodes should not be placed on the outside of the packaging, and only on the device itself, ideally on a non-prominent part of the device. (e.g., the bottom).

The QR code sticker should not placed on any part of the device that might become visible to machine vision systems in the same area. This includes security systems, robotic vacuum cleaners, anyone taking a picture with a camera. Such systems may store the picture(s) in such a way that a future viewer of the image will be able to decode the QR code, possibly through assembly of multiple pictures. Of course, the QR code is not, however, a certain indicator that the device is present, only that the QR code sticker that came with the device is present.

7. Security Considerations

The mere presence of at QRcode on a device does not in itself create any security issues on its own. Neither an attached paper sticker or a laser etched code in a plastic case will not affect the device operation. The QRcode is not active, it is not in general able to communicate on nearby networks. It is conceivable that something more active is concealed in the sticker: an NFC or RFID tag for instance. But, any sticker could contain such a thing: on some university campuses stickers are often used as part of political campaigns, and can be found attached all over the place.

Security issues that this protocol create are related to assumptions that the presence of the QRcode might imply. The presence of the QRcode may imply to some owners or network operators that the behaviour of the device has been vetted by some authority. It is here that some caution is required.

The network operator who takes the MUD file designated by the QRcode needs to be careful that they are validating the signature on the MUD file. Not only that the file is intact, but that the signer of the file is authorized to sign MUD files for that vendor, or that the network operator has some trust if the MUD file is a crowd sourced definition. At the time of writing, [RFC8520] does not define any infrastructure to authenticate or authorize MUD file signers.

A possibly bigger risk from application of MUD file stickers to devices is that they may begin to convey a sense of safety to users of the device. The presence of the sticker, possibly with the logo of the physical establishment in which the device is located could convey to occupants of the establishment that this device is an official device. For instance, a university which only deploys sensors on the university campus that have been vetted for compliance against a MUD definition.

The risk is then of social engineering: any device with a reasonable looking QRcode may become a trusted device. An attacker that wishes

to infiltrate their own devices need only suitably camoflage the device with an appropriate sticker in order to convey legitimacy.

Another issue with the stickers is that the wrong sticker could be applied to a device by a reseller or other trusted party, either in error, or via some physical or socially engineered attack against that party. The network operator now onboards a device, and applies what they think is a legitimate network policy for the device in their hands, only it is in fact a policy for another kind of device.

Inclusion of the device specific MAC address (described in Section 3.2.5) in the QRcode makes use of the MUD code much easier as it identifies the device specifically. If the MAC addresss is not included, then a network operator, having the device in their hands, then has to associate the policy with the device through some other interface.

Despite the significant advantage of having the MAC address include, it is unlikely that third party stickers will include that. Including the MAC address requires that the QRcode for that device requires that a unique sticker be created for each device. This is possible if the sticker is applied by a manufacturer: it is already common to have a serial number and MAC address on the outside of the device. In that case, if the QRcode is part of that sticker, then the customization problem is not that complex.

For cases where a third party has produced the QRcode, it is likely that every device of a particular model will have the same QRcode applied, omitting the MAC address. This makes it more likely that a policy will be applied to the wrong device.

8. IANA Considerations

This document makes no request for IANA actions.

9. Acknowledgements

This work was supported by the Canadian Internet Registration Authority (cira.ca).

10. References

10.1. Normative References

[qrcode] Wikipedia, "QR Code", December 2019, <https://en.wikipedia.org/wiki/QR_code>.

[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>.

[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>.

[SQRL] Reverse Logistics Association, "SQRL Codes: Standardized Quick Response for Logistics, Using the 12N Data Identifier", February 2017, <https://rla.org/resource/12n-documentation>.

10.2. Informative References

[chickenegg] Wikipedia, "Chicken or the egg", December 2019, <https://en.wikipedia.org/wiki/Chicken_or_the_egg>.

[I-D.ietf-anima-bootstrapping-keyinfra] Pritikin, M., Richardson, M. C., Eckert, T., Behringer, M. H., and K. Watsen, "Bootstrapping Remote Secure Key Infrastructure (BRSKI)", Work in Progress, Internet-Draft, draft-ietf-anima-bootstrapping-keyinfra-45, 11 November 2020, <https://www.ietf.org/archive/id/draft-ietf-anima-bootstrapping-keyinfra-45.txt>.

[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-04, 6 October 2021, <https://www.ietf.org/archive/id/draft-ietf-opsawg-mud-acceptable-urls-04.txt>.

[ieee802-1AR] IEEE Standard, "IEEE 802.1AR Secure Device Identifier", 2009, <http://standards.ieee.org/findstds/standard/802.1AR-2009.html>.

[isoiec18004] ISO/IEC, "Information technology - Automatic identification and data capture techniques - QR Code bar

code symbology specification (ISO/IEC 18004)", February 2015.

[mh10] "ANSI MH10.8.2 Committee", May 2021, <https://webstore.ansi.org/Standards/MHIA/ANSIMH102016>.

[MUDfiles] UNSW Sydney, ., "MUD Profiles", July 2019, <https://iotanalytics.unsw.edu.au/mud/>.

[MUDgee] Hamza, A., "MUDgee", July 2019, <https://github.com/ayyoob/mudgee>.

[qrcodewebservice] Internet, "QR Code Generators", December 2019, <https://duckduckgo.com/?q=QR+code+web+generator>.

[qrencode] Fukuchi, K., "QR encode", December 2019, <https://fukuchi.org/works/qrencode/index.html.en>.

[RFC8259] Bray, T., Ed., "The JavaScript Object Notation (JSON) Data Interchange Format", STD 90, RFC 8259, DOI 10.17487/RFC8259, December 2017, <https://www.rfc-editor.org/info/rfc8259>.

[URLshorten] Wikipedia, "URL shortening", May 2021, <https://en.wikipedia.org/wiki/URL_shortening>.

Authors' Addresses

Michael Richardson
Sandelman Software Works

Email: mcr+ietf@sandelman.ca

Jacques Latour
CIRA Labs

Email: Jacques.Latour@cira.ca

Hassan Habibi Gharakheili
UNSW Sydney

Email: h.habibi@unsw.edu.au