Workgroup: Network Working Group
Internet-Draft: draft-richardson-mud-qrcode-07
Published: 21 March 2022
Intended Status: Informational
Expires: 22 September 2022
A   M. Richardson   J. Latour
uSandelman Software Works   CIRA Labs
t
h
o
r
s
:
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 Manufacturer Usage Description (MUD) as described in RFC8520.

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.

RFC-EDITOR-please-remove: This work is tracked at https://github.com/mcr/mud-qrcode

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 22 September 2022.

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 to communicate to a network enforcement point the MUD URL, i.e., the URL of the resulting MUD file in JSON [RFC8259]: 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 [RFC8995].

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]): manufacturers are not motivated to (and thus likely do not) include MUD URLs in their products, as they believe that there are no gateways using those URLs. At the same time, gateways have little incentive to (and thus likely do not) include code that processes MUD URLs, as it is believed that no products have and disseminate them.

The protocol described in this document 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.

The QR-based protocol is presented as a convenient alternative when the mechanisms from RFC 8520 are not available to use, on the device or the gateway.

Affixing a sticker 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 5 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 be 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.

Readers should be familiar with the terminology in [RFC8520], including: MUD file, MUD URL, Manufacturer and MUD manager and controller.

3. Protocol

This QRcode protocol builds upon the work by [SQRL]. That protocol is very briefly described in Section 3.1. 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 Separator characters are used in SQRL to signify the beginning of a new unit of data. A MUD specific parser that encounters a Field Separator 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 is the Product Name in ASCII. Its 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.

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. MUD files can contain arbitrary UTF8 for the model-name, while the SQRL files are expected to be 7-bit US-ASCII.

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. The Security Considerations section of [RFC3986] applies, particularly section 7.1.

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. Device MAC Address

If a MAC address is used as a unique device identifier (which is RECOMMENDED if possible), then it MUST be included in this Data Record.

[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) This document RECOMMENDS use of upper-case hexadecimal letters.

Parsers that find punctuation (such as colons (":"), dashes ("-"), or white space) MUST skip over it. Parses MUST tolerate hexadecimal in both upper, lower and even mixed case. Systems SHOULD canonicalize it to upper case.

4. Applicability

The use of stickers to convey MUD URLs would appear to have little value when the stickers are applied by the end user organization and consumed by the same. This is particularly the case when the QR code does not include the device MAC address. In such a situation the installer handling the device would scan the QR code to get the appropriate MUD file reference, and have to input the associated MAC address as well.

In such a case, one might wonder why the installer couldn't just enter the appropriate MAC address and select the appropriate ACLs for the device. No MUD file or QR code to convey it would be useful at all.

The use of a MUD file (or QR code other other way to convey it) has the advantage that it offers several layers of indirection:

1. The list of ACLs for a given device may be added or removed.

2. The ACLs may refer to DNS names, which may map to IPv4 or IPv6 addresses.

3. The entire file may be replaced, and may also include supply chain information, such as Software Bill of Materials (SBOM).

In addition, the mechanism to install a new device (MAC address) to MUD file mapping does not need to permit any other network security settings to be alterable by the person doing the installation.

5. 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) enforced in the network 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. To recap, if a feature is removed from the firmware, and the MUD file still permits it then there is a potential hole that could perhaps be exploited. The opposite situation, where a MUD file wrongly forbids something leads to false positives in the security system, and evidence is that this results in the entire system being ignored. Preventing attacks on core infrastructure may be more important than getting the ACL perfect.

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.

6. 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 to 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, and to be applied in the field.

7. 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 be 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.

The use of URL shorting services discussed in Section 3.2.4 may result in trading convenience and efficiency with privacy, since the service provider might leverage per-device or per-customer short URLs to track and correlate requests.

8. Security Considerations

8.1. QR codes are not assurances

The mere presence of a 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 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.

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 be seen as a trusted device (even though such trust is not justified based on that evidence.) An attacker that wishes to infiltrate their own devices need only suitably camouflage the device with an appropriate sticker in order to convey legitimacy.

8.2. MUD files can have signatures

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.

8.3. URL Shortening services can change content

If a URL shorterning service is used, it is possible that the MUD Controller is redirected to another MUD file with different content. The use of MUD signatures can detect attacks on the integrity of the file. To do this, the MUD controller needs to be able to verify the signature on the file.

If a Trust On First Use (TOFU) policy is used for signature trust anchors, then the URL shortening service can still attack, if it substitutes content and signature on the first use. MUD controllers and the people operating them need to be cautious when using TOFU.

8.4. MUD QR code stickers could be confused

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.

Careful examination of stickers is in order!

8.5. QR code can include MAC address

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 address is not included, then a network operator, having the device in their hands, has to associate the policy with the device through some other interface.

Despite the significant advantage of having the MAC address included, it is unlikely that third party stickers will include that. Including the MAC address requires that a unique sticker with a QRcode 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 increases the possibility that the wrong policy will be applied to a device.

9. IANA Considerations

This document makes no request for IANA actions.

10. Acknowledgements

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

11. References

11.1. 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>.

[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>.

11.2. Informative References

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

[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>.

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

[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>.

[RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, DOI 10.17487/RFC3986, January 2005, <https://www.rfc-editor.org/info/rfc3986>.

[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>.

[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/info/rfc8995>.

[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