

MEDIAMAN
Internet-Draft
Updates: [6838](#) (if approved)
Intended status: Standards Track
Expires: 25 August 2022

M. Sporny
A. Guy
Digital Bazaar
February 2022

Media Types with Multiple Suffixes
draft-ietf-mediamaan-suffixes-01

Abstract

This document updates [RFC 6838](#) "Media Type Specifications and Registration Procedures" to describe how to interpret subtypes with multiple suffixes.

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 5 August 2022.

Copyright Notice

Copyright (c) 2022 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Revised BSD License text as described in Section 4.e of the [Trust Legal Provisions](#) and are provided without warranty as described in the Revised BSD License.

Internet-Draft

Media Types with Multiple Suffixes

February 2022

Table of Contents

1.	Introduction	2
1.1.	Conventions Used in This Document	2
2.	Media Types with Multiple Suffixes	2
2.1.	Processing Multiple Suffixes	3
2.2.	Security Considerations	4
2.2.1.	Media Type Fibbing	4
3.	Normative References	4
Appendix A.	Acknowledgements	5
	Authors' Addresses	5

[1.](#) Introduction

As written, [RFC 6838](#) [[RFC6838](#)] permits the registration of media type subtype names which contain any number of occurrences of the "+" character. [RFC 6838](#) defines the characters following the final "+" to be a structured syntax suffix, but does not define anything further about how to interpret subtype names containing more than one "+" character.

This document updates [RFC 6838](#) to clarify how to interpret subtype names containing more than one "+" character as subtypes with multiple suffixes.

As registration of media types which use a structured suffix has become widely supported, this enables further specialization of media types that build on already registered and well-defined media types which themselves use a structured suffix.

[1.1.](#) Conventions Used in This Document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [[RFC2119](#)] when they appear in ALL CAPS. They may also appear in lower or mixed case as plain English words, without any normative meaning.

[2.](#) Media Types with Multiple Suffixes

The following paragraphs are additions to [RFC 6838](#).

Media types MAY be registered with more than one suffix appended to the base subtype name. The suffixes MUST be interpreted as ordered. Valid media type names containing a structured suffix are built from right to left (not left to right). Characters on the left-most side of the left-most "+" in a subtype name specify the base subtype name. Characters to the right of each "+" in a subtype name denote additional structured syntax suffixes.

Media types with more than one suffix MUST be registered according to the procedure defined in [RFC6838]. A new base subtype name MUST only be registered with suffix combinations that are already registered in their own right in the Structured Syntax Suffixes registry (<https://www.iana.org/assignments/media-type-structured-suffix/media-type-structured-suffix.xhtml>).

For example, a media type that uses two suffixes, such as "application/foo+xml+gzip" is only permitted insofar as "+gzip" and "+xml" are already registered structured syntax suffixes.

2.1. Processing Multiple Suffixes

Registered media types have clear processing rules. In cases where specific handling of the exact media type is not required, receivers of the media type MAY do generic processing on the underlying representation according to their ability to process any subset of the suffix(es) from right to left inclusive. In other words, an application can choose to ignore the base subtype name and left-most "+" from a media type with multiple suffixes, and process according to the remaining media type suffix(es).

This sort of generic processing MAY be utilized in a processing pipeline where each segment of the pipeline handles a particular structured syntax suffix by applying decoding rules associated with the structured syntax suffix in the Structured Syntax Suffixes Registry (<https://www.iana.org/assignments/media-type-structured-suffix/media-type-structured-suffix.xhtml>). The segment of the

pipeline could then remove the structured syntax suffix from the media type and then pass the output of the decoding operation as well as the modified media type further down the pipeline.

For example, for the media type "application/did+ld+json", applications can choose to process the underlying representation according any of the following processing models: 1) application/did+ld+json (as specified in the Media Type Registry (<https://www.iana.org/assignments/media-types/media-types.xhtml>)), 2) +ld+json (as specified in the Structured Syntax Suffixes Registry (<https://www.iana.org/assignments/media-type-structured-suffix/media-type-structured-suffix.xhtml>)), or 3) +json (as specified in the

Structured Syntax Suffixes Registry (<https://www.iana.org/assignments/media-type-structured-suffix/media-type-structured-suffix.xhtml>)). As a further example, for the media type "image/svg+xml+gzip", applications can choose to process the underlying representation according any of the following processing models: 1) image/svg+xml+gzip (as specified in the Media Type Registry (<https://www.iana.org/assignments/media-types/media-types.xhtml>)), 2) +gzip (as specified in the Structured Syntax Suffixes Registry (<https://www.iana.org/assignments/media-type-structured-suffix/media-type-structured-suffix.xhtml>)), and then +xml (as specified in the Structured Syntax Suffixes Registry (<https://www.iana.org/assignments/media-type-structured-suffix/media-type-structured-suffix.xhtml>)).

If an application choses to utilize a portion of the media type that is a structured syntax suffix, the specification referred to in the the "Encoding Considerations" entry of the Structured Syntax Suffixes Registry (<https://www.iana.org/assignments/media-type-structured-suffix/media-type-structured-suffix.xhtml>) MUST be used for both encoding and decoding the byte stream associated with the media type.

[2.2.](#) Security Considerations

[2.2.1.](#) Media Type Fibbing

It is possible for attacker to utilize multiple structured suffixes in a way that tricks unsuspecting toolchains into skipping important security checks and allowing viruses to propagate. For example, an attacker might utilize an "application/vnd.ms-

excel.addin.macroEnabled.12+zip" structured suffix to trigger an unzip process that would then invoke Microsoft Excel directly, bypassing anti-virus tooling that would otherwise block a macro-enabled MS Excel file containing a virus of some kind from being scanned or opened.

While the likelihood of these sorts of attacks are low, they are not zero and enterprising attackers might take advantage of applications that carelessly register themselves in a structured suffix processing toolchain. These sorts of toolchains need to ensure that the incoming media type is not blindly trusted and that proper magic header or file structure checking is performed before allowing the encoded data to drive operations that might negatively impact the application environment or operating system.

3. 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>>.

[RFC6838] Freed, N., Klensin, J., and T. Hansen, "Media Type Specifications and Registration Procedures", [BCP 13](#), [RFC 6838](#), DOI 10.17487/RFC6838, January 2013, <<https://www.rfc-editor.org/info/rfc6838>>.

Appendix A. Acknowledgements

The editors would like to thank the following individuals for feedback on the specification (in alphabetical order): Martin J. Duerst, Ivan Herman, Graham Klyne, Murray S. Kucherawy, and Ted Thibodeau Jr.

Authors' Addresses

Manu Sporny
Digital Bazaar
203 Roanoke Street W.

Blacksburg, VA 24060
United States of America
Email: msporny@digitalbazaar.com
URI: <http://manu.sporny.org/>

Amy Guy
Digital Bazaar
203 Roanoke Street W.
Blacksburg, VA 24060
United States of America
Email: rhiaro@digitalbazaar.com
URI: <https://rhiaro.co.uk/>