

REPUTE Working Group
Internet-Draft
Intended status: Standards Track
Expires: October 8, 2012

N. Borenstein
Mimecast
M. Kucherawy
Cloudmark
April 6, 2012

A Media Type for Reputation Interchange
draft-ietf-repute-media-type-02

Abstract

This document defines a media type for exchanging reputation information about an arbitrary class of object.

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 <http://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 October 8, 2012.

Copyright Notice

Copyright (c) 2012 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 (<http://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 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	3
2.	Terminology and Definitions	3
2.1.	Key Words	3
2.2.	Other Definitions	3
3.	Description	3
3.1.	Reputon Keys	4
3.2.	Reputon Structure	5
3.3.	Example Reply	6
4.	Scores	7
5.	IANA Considerations	8
5.1.	application/reputon Media Type Registration	8
5.2.	Reputation Applications Registry	9
6.	Security Considerations	11
7.	References	11
7.1.	Normative References	11
7.2.	Informative References	11
Appendix A.	Acknowledgments	12
Appendix B.	Public Discussion	12
	Authors' Addresses	12

1. Introduction

This document defines a media type for use when answering a reputation query using the "long form" query defined in [[I-D.REPUTE-QUERY-HTTP](#)], which uses [[HTTP](#)].

Also included is the specification for an IANA registry to contain definitions and symbolic names for known reputation applications and corresponding response sets.

2. Terminology and Definitions

This section defines terms used in the rest of the document.

2.1. Key Words

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 [[KEYWORDS](#)].

2.2. Other Definitions

Other terms of importance in this document are defined in [[I-D.REPUTE-MODEL](#)], the base document in this document series.

3. Description

A "reputon" is a single independent object containing reputation information. A particular query about a subject of interest will receive one or more reputons in response, depending on the nature of the data collected and reported by the server.

The format selected for the representation of a reputon is Javascript Object Notation (JSON), defined in [[JSON](#)]. Accordingly, a new media type, "application/reputon+json", is defined for the JSON representation of reputational data, typically in response to a client making a request for such data about some subject. This media type has one optional parameter, "app", which defines the specific reputation application in whose context the query is made and the response returned. If absent, a generic reputation query is assumed for which only a simple reply is allowed.

The body of the media type consists of a JSON document that contains the reputation information requested. A detailed description of the expected structure of the reply is provided below.

3.1. Reputon Keys

The key pieces of data found in a reputon for all reputation applications are defined as follows:

RATER: The identity of the entity providing the reputation information, typically expressed as a DNS domain name.

ASSERTION: A keyword indicating the specific assertion or claim being rated. In the absence of an "app" parameter on the media type, the reputon can only indicate generic goodness, with the default assertion "IS-GOOD," but each application is expected to define additional ASSERTIONS.

RATED: The identity of the entity being rated. The nature of this field is application-specific; it could be domain names, email addresses, driver's license numbers, or anything that uniquely identifies the entity being rated. Documents that define specific reputation applications are required to define syntax and semantics for this field.

RATING: The overall rating score for that entity, expressed as a floating-point number between 0.0 and 1.0 inclusive. See [Section 4](#) for discussion.

The following are OPTIONAL for all applications, to be used in contexts where they are appropriate:

CONFIDENCE: The level of confidence the reputation provider has in the value presented being accurate, expressed as a floating-point number between 0.0 and 1.0 inclusive.

RATER-AUTHENTICITY: The level of confidence in that identity being genuine, expressed as a floating-point number between 0.0 and 1.0 inclusive.

SAMPLE-SIZE: The number of data points used to compute that score, possibly an approximation. Expressed as an unsigned 64-bit integer. The units are deliberately not specified, since not all reputation service providers will collect data the same way. Consumers will need to determine out-of-band the units being reported and apply this value accordingly within their local policies.

UPDATED: A timestamp indicating when this value was generated. Expressed as the number of seconds since January 1, 1970 00:00 UTC.

A particular application that registers itself with IANA MAY also define additional application-specific attribute/value pairs beyond these standard ones.

Further, particular application service providers MAY provide local extensions to registered applications. Syntax for these will need to be specified and accommodated privately between clients and servers.

3.2. Reputon Structure

A reputon expressed in JSON consists of an object that itself contains zero or more objects whose names are "reputon". Each reputon object is a set of key-value pairs, where the keys are the names of particular properties that comprise a reputon (as listed above, or as provided with specific applications), and values are the content associated with those keys. The set of keys that make up a reputon within a given application are known as that application's "response set".

Thus, the following simple example:

Content-type: application/reputon+json

```
{
  "reputon":
  {
    "rater": "RatingsRUs.example.com",
    "rater-authenticity": 1.0,
    "assertion": "IS-GOOD",
    "rated": "Alex Rodriguez",
    "rating": 0.99,
    "sample-size": 50000
  }
}
```

...indicates we are absolutely sure (1.0) that the entity "RatingsRUs.example.com" consolidated 50000 data points (perhaps from everyone in Yankee Stadium) and concluded that Alex Rodriguez is very very good (0.99) at something. It doesn't tell us what he's good at, and while it might be playing baseball, it could just as well be paying his taxes on time.

A more sophisticated usage would define a baseball application with a response set of specific assertions, so that this example:


```
Content-type: application/reputon+json; app="baseball"
```

```
{
  "reputon":
  {
    "rater": "baseball-reference.example.com",
    "rater-authenticity": 1.0,
    "assertion": "HITS-FOR-POWER",
    "rated": "Alex Rodriguez",
    "rating": 0.99,
    "sample-size": 50000
  }
}
```

...would indicate that 50000 fans polled by the entity baseball-reference.example.com rate A-Rod very highly in hitting for power, whereas this example:

```
Content-type: application/reputon+json; app="baseball"
```

```
{
  "reputon":
  {
    "RATER": "baseball-reference.example.com",
    "RATER-AUTHENTICITY": 1.0,
    "ASSERTION": "CLUTCH-HITTER",
    "RATED": "Alex Rodriguez",
    "RATING": 0.4,
    "SAMPLE-SIZE": 50000
  }
}
```

...would indicate that a similar poll indicated a somewhat weaker consensus that A-Rod tends to choke in critical baseball situations.

In practice, most usage of reputons is expected to make use of the "app" parameter to target an application-specific set of assertions.

3.3. Example Reply

The following is an example reputon generated using this schema, including the media type definition line:


```
Content-Type: application/reputon+json; app="email-id"
```

```
{
  "reputon":
  {
    "rater": "rep.example.net",
    "rater-authenticity": 0.95,
    "assertion": "SPAM",
    "identity": "DKIM",
    "rated": "example.com",
    "rating": 0.012,
    "sample-size": 16938213,
    "updated": 1317795852
  }
}
```

Here, reputation agent "rep.example.net" is asserting within the context of the "email-id" application that "example.com" appears to be associated with spam 1.2% of the time, based on just short of 17 million messages analyzed or reported to date. The "email-id" application has declared the extension key "identity" to indicate how the subject identifier was used in the observed data, establishing some more specific semantics for the "rating" value. In this case, the extension is used to show the identity "example.com", the subject of the query, is extracted from the analyzed messages using the [\[DKIM\]](#) "d=" parameter for messages where signatures validate. The reputation agent is 95% confident of this result. (See [\[I-D.REPUTE-EMAIL-IDENTIFIERS\]](#) for details about the registered email identifiers application.)

4. Scores

The score presented as the value in the RATING parameter appears as a floating point value between 0.0 and 1.0 inclusive. The intent is that the definition of an assertion within an application will declare what the anchor values 0.0 and 1.0 specifically mean. Generally speaking, 1.0 implies full agreement with the assertion, while 0.0 indicates no support for the assertion.

The definition will also specify the type of scale in use when generating scores, to which all reputation service providers for that application space must adhere. This will allow a client to change which reputation service provider is being queried for a given without having to learn through some out-of-band method what the new provider's values mean. For example, a registration might state that ratings are linear, which would mean a score of "x" is twice as strong as a value of "x/2".

5. IANA Considerations

This document presents two actions for IANA, namely the creation of the new media type "application/reputon+json" and the creation of a registry for reputation application types. Another document in this series creates an initial registry entry for the latter.

5.1. application/reputon Media Type Registration

This section provides the media type registration application from [\[MIME-REG\]](#) for processing by IANA:

To: ietf-types@iana.org

Subject: Registration of media type application/reputon

Type name: application

Subtype name: reputon+json

Required parameters: none

Optional parameters:

app: Names the reputation application in use within the reputon, which defines the valid assertions and any extensions that may also be valid (i.e., the response set) for that application. These MUST be registered with IANA.

Encoding considerations: "7bit" encoding is sufficient and MUST be used to maintain readability when viewed by non-MIME mail readers.

Security considerations: See [Section 6](#) of [this document].

Interoperability considerations: Implementers MUST ignore any "app" values, attribute/value pairs, or response set items they do not support.

Published specification: [this document]

Applications that use this media type: Any application that wishes to query a service that provides reputation data using the "long form" defined in [\[I-D.REPUTE-QUERY-HTTP\]](#). The example application is one that provides reputation expressions about DNS domain names found in email messages.

Additional information: The value of the "app" parameter MUST also be registered with IANA.

Person and email address to contact for further information:

Nathaniel Borenstein <nps@guppylake.com>

Murray S. Kucherawy <msk@cloudmark.com>

Intended usage: COMMON

Author:

Nathaniel Borenstein

Murray S. Kucherawy

Change controller: IESG

5.2. Reputation Applications Registry

IANA is requested to create the "Reputation Applications" registry. This registry will contain names of applications used with the application/reputon+json media type (and other media types that carry reputons), as defined by this document.

New registrations or updates MUST be published in accordance with the "Specification Required" guidelines as described in [[IANA-CONSIDERATIONS](#)].

New registrations and updates MUST contain the following information:

1. Name of the application being registered or updated
2. Short description of the application (i.e., the class of entity about which it reports reputation data)
3. The document in which the application is defined
4. New or updated status, which MUST be one of:

current: The application is in current use

deprecated: The application is in current use but its use is discouraged

historic: The application is no longer in current use

5. A description of the subject of a query within this reputation, and a legal syntax for the same
6. An optional table of query parameters that are specific to this application; each table entry must include:

Name: Name of the query parameter

Status: (as above)

Description: A short description of the purpose of this parameter

Syntax: A reference to a description of valid syntax for the parameter's value

Required: "yes" if the parameter is mandatory, "no" otherwise

A document creating a reputation application MUST include:

1. A list of one or more assertions registered within this application; each table entry must include:

Name: Name of the assertion

Description: A short description of the assertion, with specific meanings for values of 0.0 and 1.0

Scale: A short description of the scale used in computing the value (see [Section 4](#) of this document)

A document creating a reputation application MAY include:

1. A list of one or more response set extension keys for use within this application; each table entry must include:

Name: Name of the extension key

Description: A short description of the key's intended meaning

Syntax: A description of valid values that can appear associated with the key

6. Security Considerations

This section describes security considerations introduced by the media type and content syntax defined here.

[TBD]

7. References

7.1. Normative References

[I-D.REPUTE-MODEL]

Borenstein, N. and M. Kucherawy, "A Model for Reputation Interchange", [draft-ietf-repute-model](#) (work in progress), June 2011.

[I-D.REPUTE-QUERY-HTTP]

Borenstein, N. and M. Kucherawy, "Reputation Data Interchange using HTTP and XML", [draft-ietf-repute-query-http](#) (work in progress), November 2011.

[JSON]

Crockford, D., "The application/json Media Type for JavaScript Object Notation (JSON)", [RFC 4627](#), July 2006.

[KEYWORDS]

Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.

7.2. Informative References

[DKIM]

Crocker, D., Ed., Hansen, T., Ed., and M. Kucherawy, Ed., "DomainKeys Identified Mail (DKIM) Signatures", [RFC 6376](#), September 2011.

[HTTP]

Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., and T. Berners-Lee, "Hypertext Transfer Protocol -- HTTP/1.1", [RFC 2616](#), June 1999.

[I-D.REPUTE-EMAIL-IDENTIFIERS]

Borenstein, N. and M. Kucherawy, "A Reputation Vocabulary for Email Identifiers", [draft-ietf-repute-email-identifiers](#) (work in progress), November 2011.

[IANA-CONSIDERATIONS]

Narten, T. and H. Alvestrand, "Guidelines for Writing an

IANA Considerations Section in RFCs", [RFC 5226](#), May 2008.

[MIME-REG]

Freed, N. and J. Klensin, "Media Type Specifications and Registration Procedures", [RFC 4288](#), December 2005.

[Appendix A](#). Acknowledgments

The authors wish to acknowledge the contributions of the following to this specification: Frank Ellermann, Tony Hansen, Jeff Hodges, John Levine, David F. Skoll, and Mykyta Yevstifeyev.

[Appendix B](#). Public Discussion

Public discussion of this suite of documents takes place on the domainrep@ietf.org mailing list. See <https://www.ietf.org/mailman/listinfo/domainrep>.

Authors' Addresses

Nathaniel Borenstein
Mimecast
203 Crescent St., Suite 303
Waltham, MA 02453
USA

Phone: +1 781 996 5340
Email: nsb@guppylake.com

Murray S. Kucherawy
Cloudmark
128 King St., 2nd Floor
San Francisco, CA 94107
USA

Phone: +1 415 946 3800
Email: msk@cloudmark.com

