Internet Engineering Task Force (IETF)
Request for Comments: 7072
Category: Standards Track
ISSN: 2070-1721

N. Borenstein
Mimecast
M. Kucherawy

November 2013

A Reputation Query Protocol

Abstract

This document defines a mechanism to conduct queries for reputation information over the HyperText Transfer Protocol (HTTP) using JavaScript Object Notation (JSON) as the payload meta-format.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 5741.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at http://www.rfc-editor.org/info/rfc7072.

Table of Contents

1. Introduction

This document defines a method to query a reputation data service for information about an entity, using the HyperText Transfer Protocol (HTTP) as the transport mechanism and JSON as the payload meta-format.

The mechanism is a two-stage query:

1. A client retrieves a template from a server that describes the construction of a Universal Resource Identifier (URI) that will be the actual query;

2. The client then uses the constructed URI to request the reputation data from the server.

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 [RFC7070] and [RFC7071].

3. Description

3.1. Overview

The components to the question being asked are the following:

- The subject of the query;
- The name of the host, or the IP address, at which the reputation service is available;
- The name of the reputation application, i.e., the context within which the subject is being evaluated;
- Optionally, names of the specific reputation assertions or attributes that are being requested.

There is no discovery protocol for finding reputation services. These are typically subscription services, negotiated between operators through some out-of-band method.

Assertions are discussed in [RFC7071].

The name of the application, if given, is expected to be one registered with IANA in the "Reputation Applications" registry, which is defined in [RFC7071]. A server receiving a query about an application it does not recognize or explicitly support (e.g., by virtue of private agreements or experimental extensions) MUST return a 404 error code.

A reputation query made via [HTTP] encodes the question being asked in an HTTP GET method. The specific syntax of the query itself is specified by retrieving a URI template from the reputation service, completing the template, and then issuing the query.

3.2. URI Template

The template file is retrieved by requesting the [WELL-KNOWN-URI] "repute-template" from the host providing reputation service, using HTTP. (The registration for this well-known URI is in Section 4.) The server returns the template file in a reply that MUST use the

text/plain media type (see [MIME]) and SHOULD include an Expires field (see Section 14.21 of [HTTP]) indicating a duration for which the template is to be considered valid by clients and not re-queried.

If an Expires field is present, the client SHOULD NOT send another query to the same server prior to the timestamp in the field. If no Expires field is present, the client SHOULD wait at least one day before sending another query to the same server (i.e., the client assumes a default expiration of one day).

The template file might contain more than one template. Such a file MUST have each template separated by a carriage return (ASCII 0x0D) and newline (ASCII 0x0A) character, as is typical for most text-based Internet protocols.

Each template in the file is expanded using the variables that are the parameters to the query. These parameters are either the subject about which reputation information is sought (or details associated with it) or other parameters that are established out-of-band with the reputation service; they are not established by any automated discovery described here. The client then attempts to query each expanded template that uses a URI scheme it is capable of querying, in the order presented in the file, until the client finds one to which it can establish a usable connection and issue the query.

For example, given the following template:

http://{service}/{application}/{subject}/{assertion}

A query about the use of the domain "example.org" in the "email-id" application context to a service run at "example.com", where that application declares a required "subject" parameter, requesting the "SPAM" reputation assertion, would be formed as follows:

http://example.com/email-id/example.org/spam

3.3. Syntax

The syntax for the [URI] of the query is constructed using a template as per [URI-TEMPLATE]. (See Section 3.2.) Clients MUST provide the following values in the expansion of the template:

application: The name of the application reputation in whose context the request is being made. These names are registered with IANA, and conform to the ABNF "token" found in [MIME].

service: The hostname or IP address to which the query is being sent. This MUST be the same as the host to which the template query was issued.

subject: The subject of the query, extracted from some content to be evaluated. The subject portion of the template conforms to the ABNF "value" found in [MIME].

The following variable can also be provided. It is not mandatory in this model, but a specific application (defined in its own extension document) might declare it mandatory in a specific context:

assertion: The name of the specific assertion of interest to the client. Assertion names conform to the ABNF "token" found in [MIME]. If absent, the client is indicating that it requests all available assertion information.

If a template contains a variable that is not required and the client does not have a value to insert, it substitutes the empty string into the template in place of that variable. Service providers crafting templates MUST do so such that a client doing an empty variable expansion will still produce a syntactically and semantically valid and unambiguous URI. For example, given this template:

```
http://{service}/{application}/{subject}/{assertion}/{a}/{b}
```

If "{a}" and "{b}" are optional and "{a}" expands to the empty string, then the resulting URI will have adjacent backslash ("/", ASCII 0x2F) characters and one path component after the assertion. If the server interpreting the URI's path component removes or ignores adjacent backslash characters (such as is done with the UNIX filesystem), the server will be unable to distinguish an empty "{a}" from an empty "{b}", and it could serve the wrong response. Where possible, the template needs to be constructed such that expansion of optional variables yields an unambiguous result. For example, an unambiguous version of the above would be:

```
http://{service}/{application}/{subject}/{assertion}/a={a}/b={b}
```

...or, even better, using URI template set expansions:

```
http://{service}/{application}/{subject}/{assertion}{?a,b}
```

Every application space has a set of assertions applicable to its own context. [RFC7071] defines a single assertion assumed to exist in any application that does not define its own assertion set.

Reputation applications can extend the set of optional or required query parameters as part of their IANA registration actions. The set enumerated above establishes the base set common to all of them. Further, additional required or optional extension query parameters might be defined by specific reputation service providers, though these are private arrangements between client and server and will not be registered with IANA.

Authentication between reputation client and server is outside the scope of this specification. It could be provided through a variety of available transport-based or object-based mechanisms, including a later extension of this specification.

3.4. Response

The response is expected to be contained in a media type designed to deliver reputons. A media type designed for this purpose, "application/reputon+json", is defined in [RFC7071].

If the server generates responses that contain an Expires field (see Section 14.21 of [HTTP]), that timestamp MUST align with the "expires" field within the response, if any. Failing to do so can result in a state where the response has expired, but the HTTP reply has not, and the client would in that case be unable to get a fresh answer from the reputation server.

3.5. Protocol Support

A client has to implement HTTP in order to retrieve the query template as described in Section 3.2. Accordingly, a server can assume the client will be able to handle a URI template that produces a URI for the query using the "http" URI scheme. The template could yield a query string that uses some other URI scheme, in which case the client could try that URI as well if it supports issuing queries with that URI scheme.

A server SHOULD include support for providing service over HTTP, and publish templates indicating support for this, as a baseline for interoperability with arbitrary clients.

4. IANA Considerations

This document registers the "repute-template" well-known URI in the "Well-Known URI" registry as defined by [WELL-KNOWN-URI], as follows:

URI suffix: repute-template

Change controller: IETF

Specification document(s): [RFC7072]

Related information: none

5. Security Considerations

This document defines particular uses of existing protocols for a specific application. In particular, the basic protocol used for this service to retrieve a URI template from a well-known location is basic HTTP, which is not secure without certain extensions. Security issues relevant to use of URI templates are discussed in [URI-TEMPLATE], and those relevant to well-known URI definitions and retrieval are discussed in [WELL-KNOWN-URI].

The reputation service itself will use HTTP or other transport methods to issue queries and receive replies. Those protocols have registered URI schemes and, as such, presumably have documented security considerations. The protocol described here operates atop those URI schemes, and does not itself present new security considerations.

Reputation mechanisms represent an obvious security concern, in terms of the validity and use of the reputation information. These issues are beyond the scope of this specification. General information pertaining to using or providing reputation services can be found in [CONSIDERATIONS].

The security considerations applicable to HTTP (see Section 15 of [HTTP] apply, since this query mechanism for reputation uses that protocol. If it is desirable to conceal the content of the query and its response, use of encryption techniques such as HTTP over TLS [HTTPS] can be used.

6. References

6.1. Normative References

[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.

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

[MIME] Freed, N. and N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies", RFC 2045, November 1996.

[RFC7070] Borenstein, N., Kucherawy, M., and A. Sullivan, "An Architecture for Reputation Reporting", RFC 7070, November 2013.

[RFC7071] Borenstein, N. and M. Kucherawy, "A Media Type for Reputation Interchange", RFC 7071, November 2013.

[URI-TEMPLATE] Gregorio, J., Fielding, R., Hadley, M., Nottingham, M., and D. Orchard, "URI Template", RFC 6570, March 2012.

[URI] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, January 2005.

[WELL-KNOWN-URI] Nottingham, M. and E. Hammer-Lahav, "Defining Well-Known Uniform Resource Identifiers (URIs)", RFC 5785, April 2010.

6.2. Informative References

[CONSIDERATIONS] Kucherawy, M., "Operational Considerations Regarding Reputation Services", Work in Progress, May 2013.

[HTTPS] Rescorla, E., "HTTP Over TLS", RFC 2818, May 2000.

Appendix A. Acknowledgements

The authors would like to thank the following for their contributions to this work: Simon Hunt, Mark Nottingham, David F. Skoll, and Mykyta Yevstifeyev.

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
270 Upland Drive
San Francisco, CA 94127
USA

EMail: superuser@gmail.com