A Model for Reputation Reporting
draft-ietf-repute-model-04

Abstract

   This document describes a general architecture for a reputation-based
   service and a model for requesting reputation-related data over the
   Internet, where "reputation" refers to predictions or expectations
   about an entity or an identifier such as a domain name. The document
   roughly follows the recommendations of RFC4101 for describing a
   protocol model.

Status of this Memo

Copyright Notice

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

Historically, many Internet protocols have operated between
unauthenticated entities. For example, an email message's author
field (From) [MAIL] can contain any display name or address and is
not verified by the recipient or other agents along the delivery
path. Similarly, a sending email server using [SMTP] trusts that the
[DNS] has led it to the intended receiving server. Both kinds of
trust are easily betrayed, opening the operation to subversion of
some kind, which leads to spam, phishing, and other attacks.

In recent years, explicit identity authentication mechanisms have
begun to see wider deployment. For example, the [DKIM] protocol
permits associating a validated identifier to a message. This
association is cryptographically strong, and is an improvement over
the prior state of affairs, but it does not distinguish between
identifiers of good actors and bad. Even when it is possible to
validate the domain name in an author field (e.g.
"trustworthy.example.com" in "john.doe@trustworthy.example.com")
there is no basis for knowing whether it is associated with a good
actor worthy of trust. As a practical matter, both bad actors and
good adopt basic authentication mechanisms like DKIM. In fact, bad
actors tend to adopt them even more rapidly than the good actors do
in the hope that some receivers will confuse identity authentication
with identity assessment. The former merely means that the name is
being used by its owner or their agent, while the latter makes a
statement about the quality of the owner.

With the advent of these authentication protocols, it is possible to
statisfy the requirement for a mechanism by which mutually trusted
parties can exchange assessment information about other actors. For
these purposes, we may usefully define "reputation" as "the
estimation in which an identifiable actor is held, especially by the
community or the Internet public generally". We may call an
aggregation of individual assessments "reputation input."

While the need for reputation services has been perhaps especially
clear in the email world, where abuses are commonplace, other
Internet services are coming under attack and may have a similar
need. For instance, a reputation mechanism could be useful in rating
the security of web sites, the quality of service of an Internet
Service Provider (ISP), or an Application Service Provider (ASP).
More generally, there are many different opportunities for use of
reputation services, such as customer satisfaction at e-commerce
sites, and even things unrelated to Internet protocols, such as
plumbers, hotels, or books. Just as human beings traditionally rely
on the recommendations of trusted parties in the physical world, so
too they can be expected to make use of such reputation services in a

variety of applications on the Internet.

A full trust architecture encompasses a range of actors and
activities, to enable an end-to-end service for creating, exchanging,
and consuming trust-related information. One component of that is a
query mechanism, to permit retrieval of a reputation. Not all such
reputation services will need to convey the same information. Some
need only produce a basic rating, while others need to provide
underlying detail. This is akin to the difference between check
approval versus a credit report.

An overall reckoning of goodness versus badness can be defined
generically, but specific applications are likely to want to describe
reputations for multiple attributes: an e-commerce site might be
rated on price, speed of delivery, customer service, etc., and might
receive very different ratings on each. Therefore, the model defines
a generic query mechanism and basic format for reputation retrieval,
but allows extensions for each application.

Omitted from this model is the means by which a reputation-reporting
agent goes about collecting such data and the method for creating an
evaluation. The mechanism defined here merely enables asking a
question and getting an answer; the remainder of an overall service
provided by such a reputation agent is specific to the implementation
of that service and is out of scope here.

2. High-Level Architecture

A reputation mechanism functions as a component of a service, such as
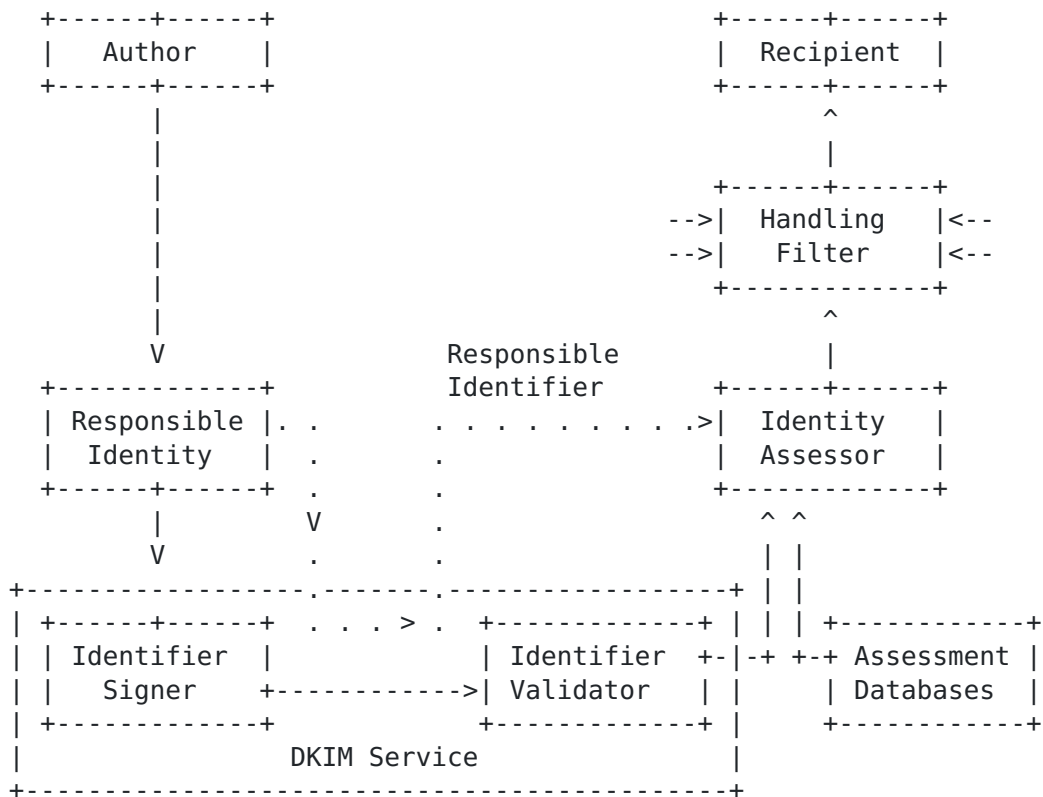that depicted in Figure 1 of [RFC5863], reproduced here:

```
      +------+------+                         +------+------+
      |   Author    |                         |  Recipient  |
      +------+------+                         +------+------+
             |                                       ^
             |                                       |
             |                                +------+------+
             |                            -->|  Handling   |<--
             |                            -->|   Filter    |<--
             |                                +------------+
             |                                       ^
             |                                       |
             V                  Responsible          |
      +------------+            Identifier     +------+------+
      | Responsible |. .      . . . . . . . .>|  Identity   |
      |  Identity   | .           .           |  Assessor   |
      +------+------+ .           .           +------------+
             |          V         .                 ^ ^
             V          .         .                 | |
   +-----------------.-------.-----------------+     | |
   | +------+------+  . . . > .  +------------+ |  |  | | +------------+
   | | Identifier  |            | Identifier  +-|-+ +-+ | Assessment |
   | |   Signer    +------------>|  Validator  | |     |  Databases  |
   | +------------+              +------------+ |     +------------+
   |               DKIM Service                 |
   +-------------------------------------------+
```

         Figure 1: [RFC5683](RFC5683) 'Actors in a Trust Sequence Using DKIM'

   Here, the reputation mechanism is shown only as a query by an
   Identity Assessor, made to Assessment Databases.

   This memo outlines the query and response mechanism. It provides the
   following definitions:

   o Vocabulary for the current work and work of this type;

   o The types and content of queries that can be supported;

   o The extensible range of response information that can be provided;

   o A query/response protocol;

   o Query/response transport conventions.

   It provides an extremely simple query/response model that can be
   carried over a variety of transports, including the Domain Name
   System. (Although not typically thought of as a 'transport', the DNS
   provides generic capabilities and can be thought of as a mechanism
   for transporting queries and responses that have nothing to do with

addresses.) Each specification for Repute transport is independent
of any other specification. A diagram of the basic query service is
found in Figure 2.

```
    +-----------+         Query                +----------+
    |           +. . . . . . . . . . . . .>|          |
    |  Client   |                              |  Server  |
    |           <. . . . . . . . . . . . +          |
    +-----+-----+         Response             +-------+--+
          |                                        ^   |
          V                                        |   |
    +------+----+                   +-----------+  |   | Response
    | Transport |---------------->| Transport |--+   | Set
    +-----------+    DNS            +-----------+      |
                     TCP                               V
                     UDP                       +------------+
                     ...                       | Reputation |
                                               | Database   |
                                               +------------+
```
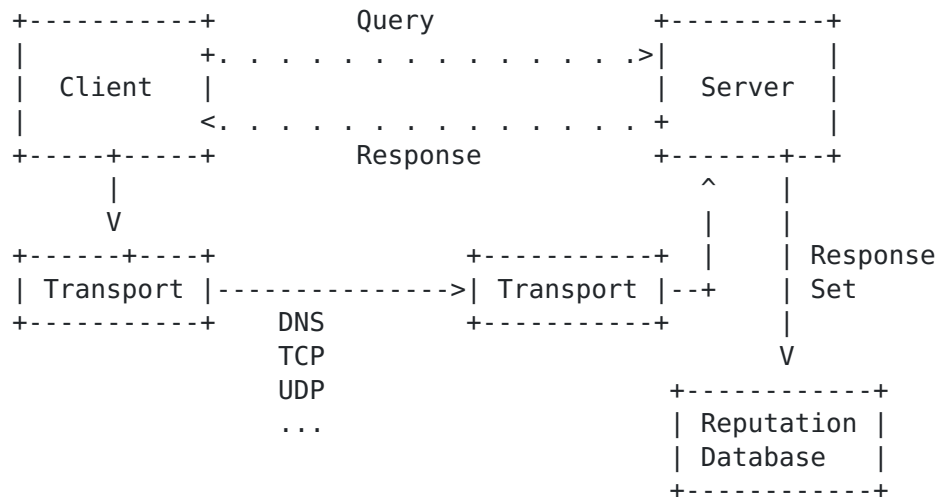
                  Figure 2: Basic Reputation Query Service

   Both the query and response are application-specific. An application
   of the model defines the parameters available to queries of that
   type, and also defines the data returned in response to any query.

3. Terminology and Definitions

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

3.1. Response Set

   A "Response Set" comprises those data that are returned in response
   to a reputation query about a particular entity. The types of data
   are specific to an application; the data returned in the evaluation
   of email senders would be different than the reputation data returned
   about a movie or a baseball player.

   Response Sets have symbolic names, and these have to be registered
   with IANA to prevent name collisions. IANA registries are created in
   a separate memo. Each definition of a Response Set needs to define
   its registry entry.

4. Information Represented in a Response Set

   The basic information to be represented in the protocol is fairly

simple, and includes the following:

o the identity of the entity providing the reputation information;

o the identity of the entity being rated;

o the overall rating score for that entity;

o the level of confidence in the accuracy of that rating; and

o the number of data points underlying that score.

Beyond this, arbitrary amounts of additional information might be
provided for specific uses of the service. The entire collection is
the Response Set for that application. The query/response protocol
defines a syntax for representing such Response Sets, but each
application defines its own Response Set. Thus, the basic information
also includes the name of the application for which the reputation
data is being expressed.

Each application requires its own specification of the Response Set.
For example, a specification might be needed for a reputation
Response Set for an "email-sending-domain"; the Response Set might
include information on how often spam was received from that domain.
Additional documents define a [MIME] type for reputation data, and
protocols for exchanging such data.

5. Information Flow in the Reputation Query Protocol

The basic Response Set could be wrapped into a new MIME media type
[MIME] or a DNS RR, and transported accordingly. It also could be
the integral payload of a purpose-built protocol. For a basic
request/response scenario, one entity (the Client) will ask a second
entity (the Server) for reputation data about a third entity (the
Target), and the second entity will respond with that data.

An applications might benefit from an extremely lightweight
mechanism, supporting constrained queries and responses, while others
might need to support larger and more complex responses.

6. IANA Considerations

This memo presents no actions for IANA.

[RFC Editor: Please remove this section prior to publication.]

7. Privacy Considerations

Some kinds of reputation data are sensitive, and should not be shared
publicly. For cases that have such sensitivity, it is imperative to
protect the information from unauthorized access and viewing. The
model described here neither suggests nor precludes any particular
transport mechanism for the data. However, for the purpose of
illustration, a reputation service that operates over HTTP might
employ any of its well-known mechanisms to solve these problems,
which include OpenPGP [OPENPGP], Transport Layer Security [TLS], and
S/MIME [SMIME].

8. Security Considerations

This memo introduces an overall protocol model, but no implementation
details. As such, the security considerations presented here are
very high-level. The detailed analyses of the various specific
components of the protocol can be found the documents that
instantiate this model.

8.1. Biased Reputation Agents

As with [VBR], an agent seeking to make use of a reputation reporting
service is placing some trust that the service presents an unbiased
"opinion" of the object about which reputation is being returned.
The result of trusting the data is, presumably, to guide action taken
by the reputation client. It follows, then, that bias in the
reputation service can adversely affect the client. Clients
therefore need to be aware of this possibility and the effect it
might have. For example, a biased system returning a reputation
about a DNS domain found in email messages could result in the
admission of spam, phishing or malware through a mail gateway (by
rating the domain name more favourably than warranted) or could
result in the needless rejection or delay of mail (by rating the
domain more unfavourably than warranted). As a possible mitigation
strategy, clients might seek to interact only with reputation
services that offer some disclosure of the computation methods for
the results they return. Such disclosure and evaluation is beyond
the scope of the present memo.

Similarly, a client placing trust in the results returned by such a
service might suffer if the service itself is compromised, returning
biased results under the control of an attacker without the knowledge
of the agency providing the reputation service. This might result
from an attack on the data being returned at the source, or from a
man-in-the-middle attack. Protocols, therefore, need to be designed
so as to be as resilient against such attacks as possible.

8.2. Malformed Messages

   Both clients and servers of reputation systems need to be resistant
   to attacks that involve malformed messages, deliberate or otherwise.
   Malformations can be used to confound clients and servers alike in
   terms of identifying the party or parties responsible for the content
   under evaluation. This can result in delivery of undesirable or even
   dangerous content.

8.3. Further Discussion

   Numerous other topics related to use and management of reputation
   systems can be found in [I-D.REPUTE-CONSIDERATIONS].

9. Informative References

   [DKIM]   Allman, E., Callas, J., Delany, M., Libbey, M., Fenton,
               J., and M. Thomas, "DomainKeys Identified Mail (DKIM)
               Signatures", RFC 4871, May 2007.

   [DNS]   Mockapetris, P., "Domain names - implementation and
               specification", STD 13, RFC 1035, November 1987.

   [I-D.REPUTE-CONSIDERATIONS]
               Kucherawy, M., "Operational Considerations Regarding
               Reputation Services", draft-ietf-repute-considerations
               (work in progress), November 2012.

   [MAIL]   Resnick, P., "Internet Message Format", RFC 5322,
               October 2008.

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

   [OPENPGP]   Callas, J., Donnerhacke, L., Finney, H., Shaw, D., and R.
               Thayer, "OpenPGP Message Format", RFC 4880, November 2007.

   [RFC5863]   Hansen, T., Siegel, E., Hallam-Baker, P., and D. Crocker,
               "DomainKeys Identified Mail (DKIM) Development,
               Deployment, and Operations", RFC 5863, May 2010.

   [SMIME]   Ramsdell, B. and S. Turner, "Secure/Multipurpose Internet
               Mail Extensions (S/MIME) Version 3.2: Message
               Specification", RFC 5751, January 2010.

   [SMTP]   Klensin, J., "Simple Mail Transfer Protocol", RFC 5321,

                    October 2008.

   [TLS]   Dierks, T. and E. Rescorla, "The Transport Layer Security
               (TLS) Protocol Version 1.2", RFC 5246, August 2008.

   [VBR]   Hoffman, P., Levine, J., and A. Hathcock, "Vouch By
               Reference", RFC 5518, April 2009.

Appendix A. Public Discussion

   Public discussion of this suite of memos 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

   Email: superuser@gmail.com

   Andrew Sullivan (editor)
   Dyn, Inc.
   150 Dow St.
   Manchester, NH 03101
   USA

   Email: asullivan@dyn.com