

REPUTE
Internet-Draft
Intended status: Informational
Expires: May 30, 2014

M. Kucherawy
November 26, 2013

Considerations Regarding Third-Party Reputation Services
draft-kucherawy-repute-consid-00

Abstract

Reputation services offer quality assessments about likely future behavior, based on past behaviors. The use of these services has become a common tool in many applications that seek to apply collected intelligence about traffic sources. Often this is done because it is common or even expected operator practice. It is therefore important to be aware of a number of considerations for both operators and consumers of the data. This document includes a collection of the best advice available regarding providers and consumers of reputation data, based on experience to date. Much of this is based on experience with email reputation systems, but the concepts are generally applicable.

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 May 30, 2014.

Copyright Notice

Copyright (c) 2013 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.	Background	3
3.	Using Reputation Services	4
4.	Providing Reputation Services	6
5.	Evolution	8
6.	Security Considerations	8
7.	IANA Considerations	8
8.	Informative References	8
Appendix A.	Acknowledgments	9

[1.](#) Introduction

Reputation services involve collecting feedback from the community about sources of Internet traffic and aggregating that feedback into a rating of some kind. Common examples include feedback about traffic associated with specific email addresses, URIs or parts of URIs, IP addresses, etc. The specific collection, analysis, and rating methods vary from one service to the next and one problem domain to the next, but several operational concepts appear to be common to all of these.

The promise of the protection that relying on reputation services offers can be enticing, and many users and operators alike typically engage those services merely because it is expected of them. A critical notion, however, is that use of such a service explicitly involves a third party in the flow of data being received. This is often taken for granted, with potentially disastrous results.

This document highlights this and other considerations in providing and consuming reputation data services.

[2.](#) Background

The anti-abuse community has historically focused on identifying sources that misbehave, i.e., that earn negative reputations. For email, this means identifying sources of spam; for security, it means identifying sources of penetration attacks. The purpose here is to identify and filter traffic from bad actors. This grew out of operational need. As the Internet grew, so did the occurrence of problematic traffic, especially in email. The pragmatics of email (i.e., the fact that the total IP address space is more constrained than the total email address space) drove the focus on using IP addresses as the focus of reputation, in addition to the fact that IP addresses have a degree of validation (via the TCP/IP infrastructure) where email addresses have had none.

The major considerations around a third-party reputation service are:

Raw data: The method of obtaining the information that will be analyzed;

Rating method: The techniques used on the collected data to compute a rating or other expression of expected behavior;

Publication: How consumers obtain the computed ratings.

A specific example of a publication method in common use in the email space is the DNS blacklist [[DNSBL](#)]. In particular, the operator of a

reputation service computes reputations of IP addresses and stores them in a database. Via a DNSBL query, a consumer can query the database as to whether mail should be accepted from a particular source of incoming [[SMTP](#)], based on previous observations and feedback. The service uses the IP address of the source as the basis for a query to the database, accessed through the Domain Name System [[DNS](#)]. [[DNSBL](#)] includes several points in its Security Considerations document that are repeated and further developed here.

However, regardless of the identifier used for a reputation, bad actors can evade detection or its consequences by changing identifiers (e.g., move to a new IP address, register a new domain name, use a sub-domain). This makes the problem space effectively boundless, especially as IPv6 rolls out, with its vastly larger address space.

A framework for reputation services is introduced in [[REPUTE](#)] and the documents it references.

[3.](#) Using Reputation Services

Operators that choose to make use of reputation services to influence content allowed to pass into or through their infrastructures need to understand that they are granting a third party (the reputation service provider, or RSP) the ability to affect the handling of incoming traffic, for better or worse. Of course, this is the whole point of engaging an RSP when everything is working properly, but a number of issues are worthy of consideration before establishing such a relationship.

Some cases have occurred where an RSP made the unilateral decision to terminate its service. To encourage its clients to stop issuing queries, it began reporting a maximally negative reputation about all subjects, causing rejection of all incoming traffic during the incident period. Although one would hope such incidents to be rare, automated means to detect such unfortunate returns (malicious or otherwise) and take remedial should be considered.

RSPs will be the subject of attacks once it is understood that success in doing so will allow malicious content to evade detection and filtering. Users of RSPs need to plan for possible interruptions in service availability or quality.

Similarly, some actors will try to "game" the service, which is to say that such actors will attempt to determine patterns of behavior that result in the reporting of favorable reputations, and in doing so, acquire artificially inflated reputations. One could reasonably assume that a reputation service is inherently fragile. For

operational clients, this should prompt balanced and comparative, rather than unilateral, use of the service.

It is suggested that, when engaging an RSP, an operator should try to learn the following things about the RSP in order to understand the exposure potential:

- o the RSP's basis for listing or not listing particular subjects;
- o if an RSP is paid by its listees, the rate and criteria for rejection from being listed;
- o how the RSP collects data about subjects;
- o how many data points are input to the reported reputation;
- o whether reputation is based on a reliable identifier;
- o how the RSP establishes reliability and authenticity of those data;
- o how continuing data validity is maintained (e.g., on-going

monitoring of the reported data and sources);

- o how actively data validity is tracked (e.g., how changes are detected);
- o how disputed reputations are handled;
- o how often input data expire;
- o whether older information is more or less influential than newer;
- o whether the reported reputation is a scalar, a Boolean value, a collection of values, or something else;
- o when transitioning among RSPs, the differences between them among these above points; that is, whether a particular score from one means the same thing from another.

An operator using an RSP would be wise to ensure it has the capability to give preference to local policies, for cases where the client expects to disagree with the reported reputation.

An operator should be able to limit the impact of a negative reputation on content acceptance. For example, rather than rejecting content outright when a negative reputation is returned, simply subject it to additional (i.e., more thorough) local analysis before permitting the

traffic to pass. In other words, the reputation may simply allow certain layers of a multi-layered filtering system to be bypassed when that reputation is favorable.

A sensible default should apply when the RSP is not available. This can also be a query to a different RSP known to be less robust than the primary one.

Recent proposals such as the experimental system implemented in [\[OPENDKIM\]](#) have focused on tailoring operation to prefer or emphasize content whose sources have positive reputations. See [Section 5](#) for discussion of this notion. As stated in [Section 1](#), negative reputations are easy to shed, while the universe of things that will earn and maintain positive reputations is relatively small. Designing a filtering system that observes these notions is expected

to be more lightweight to operate and harder to game.

One choice is to query and cross-reference multiple RSPs. This can help to detect which ones under comparison are reliable, and offsets the effect of anomalous replies. More generally, a robust mechanism that is using a third-party service needs to contain an array of mechanisms, and to limit its dependence on any one mechanism, as well as protect against for misbehavior by an individual mechanism.

[4.](#) Providing Reputation Services

Operators intending to provide a reputation service need to consider that there are many flavors of clients. There will be clients that are prepared to make use of a reputation service blindly, while others will be interested in understanding more fully the nature of the service being provided. These can be likened to a consumer credit check that only seeks a yes-or-no reply versus wanting to review a detailed credit report. An operator of an RSP should be prepared to answer as many of the questions identified in [Section 3](#) as possible, not only because wise clients will ask, but also because they reflect issues that have arisen over the years, and diligent exploration of the points they raise will result in a better reputation service.

Obviously, in computing reputations via traffic analysis, some private algorithms may come into play. For some RSPs, such "secret sauce" comprises their competitive advantage over others in the same space. This document is not suggesting that all private algorithms need to be exposed for a reputation service to be acceptable. Instead, it is anticipated that enough of the above details need to be available to ensure consumers (and in some cases, industry or the general public) that the RSP can be trusted to influence key local policy decisions.

Reputations should be based on accurate identifiers, i.e., some property of the content under analysis that is difficult to falsify. For example, in the realm of email, the address found in the From: header field of a message is typically not verifiable, while the domain name found in a validated domain-level signature is. In this case, constructing a reputation system based on the domain name is more useful than one based on the From: field.

The biggest frustration with most RSPs to date has been the challenge of dealing with errors: there often is no visible, accessible, and transparent process for remediating the errant addition of an identifier to a negative reputation list. An RSP in widespread use is perceived to have enormous power when its results are used to reject traffic outright; when a "bad" entry is added referencing a good actor, it can have destructive effects, so an effective mechanism to fix such problems needs to exist.

Clients with varying sensitivities need to be accommodated. The mechanism that is used to access the RSP should provide an ability to request that query results include details about the basis for producing those results. This will help the user to decide how to apply those results. For example, it should be possible for the reply to contain:

- o the result itself;
- o the number of data points used to compute the result;
- o the age range of the data;
- o source diversity of the input data;
- o currency of the result (i.e., when it was computed);
- o basis of the result (i.e., which identifier was used).

The systems and algorithms used by the RSP to compute the reported reputation will need to be hardened as much as practicable against gaming or other forms of data poisoning. Larger source diversities are harder to overcome with poisoned input, but are expensive to build in terms of both infrastructure and time.

Systems focused on assigning positive reputations rather than negative ones are promising since positive reputations, if made difficult to earn, put a large cost on bad actors, which may be enough to dissuade them entirely.

Recent consideration of reputation efforts is evolving toward the identification of good actors rather than bad actors, and giving them preferential treatment. This drastically reduces the problem space: There are vastly more IP addresses and email addresses used by bad actors to generate problematic traffic than are used by good actors to generate desirable traffic.

Moreover, good actors tend to be represented by stable names and addresses, allowing users to rely on these to identify and give preferential treatment to their traffic. Good actors have no need to hop around to different addresses, and already work to keep their traffic clean. In addition, good actors are willing and able to collaborate in the assessment process, such as by supplying validated identifiers that are associated with their traffic.

This new approach of focusing on identification of good actors has only been tried to date using manually edited whitelists, but has shown promising results on that scale.

6. Security Considerations

Several points are raised above that can be described as threats to the delivery of valid user data. This document highlights and discusses those matters, but introduces no new security issues.

7. IANA Considerations

This memo contains no actions for IANA.

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

8. Informative References

- [DNS] Mockapetris, P., "Domain Names -- Concepts and Facilities", [RFC 1034](#), November 1987.
- [DNSBL] Levine, J., "DNS Blacklists and Whitelists", [RFC 5782](#), February 2010.
- [OPENDKIM] "OpenDKIM (Open Source DKIM)", July 2013, <<http://www.opendkim.org>>.
- [REPUTE] Borenstein, N. and M. Kucherawy, "An Architecture for Reputation Reporting", [RFC 7070](#), November 2013.
- [SMTP] Klensin, J., "Simple Mail Transfer Protocol", [RFC 5321](#),

October 2008.

[Appendix A](#). Acknowledgments

The author wishes to acknowledge the following for their review and constructive criticism of this proposal: Chris Barton, Dave Crocker, Vincent Schonau

Author's Address

Murray S. Kucherawy

EMail: superuser@gmail.com

