



Mobile Spam Reporting Architecture

Candidate Version 1.0 – 20 Apr 2010

Open Mobile Alliance
OMA-AD-SpamRep-V1_0-20100420-C

Use of this document is subject to all of the terms and conditions of the Use Agreement located at <http://www.openmobilealliance.org/UseAgreement.html>.

Unless this document is clearly designated as an approved specification, this document is a work in process, is not an approved Open Mobile Alliance™ specification, and is subject to revision or removal without notice.

You may use this document or any part of the document for internal or educational purposes only, provided you do not modify, edit or take out of context the information in this document in any manner. Information contained in this document may be used, at your sole risk, for any purposes. You may not use this document in any other manner without the prior written permission of the Open Mobile Alliance. The Open Mobile Alliance authorizes you to copy this document, provided that you retain all copyright and other proprietary notices contained in the original materials on any copies of the materials and that you comply strictly with these terms. This copyright permission does not constitute an endorsement of the products or services. The Open Mobile Alliance assumes no responsibility for errors or omissions in this document.

Each Open Mobile Alliance member has agreed to use reasonable endeavors to inform the Open Mobile Alliance in a timely manner of Essential IPR as it becomes aware that the Essential IPR is related to the prepared or published specification. However, the members do not have an obligation to conduct IPR searches. The declared Essential IPR is publicly available to members and non-members of the Open Mobile Alliance and may be found on the “OMA IPR Declarations” list at <http://www.openmobilealliance.org/ipr.html>. The Open Mobile Alliance has not conducted an independent IPR review of this document and the information contained herein, and makes no representations or warranties regarding third party IPR, including without limitation patents, copyrights or trade secret rights. This document may contain inventions for which you must obtain licenses from third parties before making, using or selling the inventions. Defined terms above are set forth in the schedule to the Open Mobile Alliance Application Form.

NO REPRESENTATIONS OR WARRANTIES (WHETHER EXPRESS OR IMPLIED) ARE MADE BY THE OPEN MOBILE ALLIANCE OR ANY OPEN MOBILE ALLIANCE MEMBER OR ITS AFFILIATES REGARDING ANY OF THE IPR'S REPRESENTED ON THE “OMA IPR DECLARATIONS” LIST, INCLUDING, BUT NOT LIMITED TO THE ACCURACY, COMPLETENESS, VALIDITY OR RELEVANCE OF THE INFORMATION OR WHETHER OR NOT SUCH RIGHTS ARE ESSENTIAL OR NON-ESSENTIAL.

THE OPEN MOBILE ALLIANCE IS NOT LIABLE FOR AND HEREBY DISCLAIMS ANY DIRECT, INDIRECT, PUNITIVE, SPECIAL, INCIDENTAL, CONSEQUENTIAL, OR EXEMPLARY DAMAGES ARISING OUT OF OR IN CONNECTION WITH THE USE OF DOCUMENTS AND THE INFORMATION CONTAINED IN THE DOCUMENTS.

© 2010 Open Mobile Alliance Ltd. All Rights Reserved.

Used with the permission of the Open Mobile Alliance Ltd. under the terms set forth above.

Contents

OPEN MOBILE ALLIANCE	1
1. SCOPE (INFORMATIVE)	4
2. REFERENCES	5
2.1 NORMATIVE REFERENCES	5
2.2 INFORMATIVE REFERENCES	5
3. TERMINOLOGY AND CONVENTIONS	6
3.1 CONVENTIONS	6
3.2 DEFINITIONS	6
3.3 ABBREVIATIONS	6
4. INTRODUCTION (INFORMATIVE)	7
4.1 VERSION 1.0	7
5. ARCHITECTURAL MODEL	8
5.1 DEPENDENCIES	8
5.2 ARCHITECTURAL DIAGRAM	8
5.3 FUNCTIONAL COMPONENTS AND INTERFACES/REFERENCE POINTS DEFINITION	9
5.3.1 SpamRep Components	9
5.3.2 SpamRep Interfaces	9
5.3.3 External Components Used by SpamRep Enabler (Informative)	10
5.3.4 External Interfaces Used by SpamRep Enabler (Informative)	11
5.4 SECURITY CONSIDERATIONS	11
APPENDIX A. CHANGE HISTORY (INFORMATIVE)	13
A.1 APPROVED VERSION HISTORY	13
A.2 DRAFT/CANDIDATE VERSION 1.0 HISTORY	13
APPENDIX B. FLOWS (INFORMATIVE)	14
B.1 CALL FLOW FOR REPORT-BY-FINGERPRINT (VARIANT – CORPUS REQUEST)	14
B.2 CALL FLOW FOR USER INQUIRING ABOUT THE SPAM REPORT STATUS	15
B.3 SPAMREP SERVER NOTIFYING SPAM REPORT STATUS CALL FLOW	16
B.4 CALL FLOW FOR RELEASE QUARANTINE	16
B.5 SPAMREP CLIENT REPORTING SPAM MESSAGE CALL FLOW	17

Figures

Figure 1: SpamRep Architectural Diagram	8
Figure 2: SpamRep Architectural Diagram with external components and interfaces (dashed)	10
Figure 3: Call Flow of a Generic SpamRep Session	14
Figure 4: Report Spam By-Fingerprint with Corpus Request	15
Figure 5: Request for Spam Report Status Call Flow	16
Figure 6: Notifying Spam Report's Status Call Flow – two examples	16
Figure 7: Request Quarantine Release	17
Figure 8: SpamRep Client reporting Spam message Call Flow	18

1. Scope

(Informative)

This document defines the architecture for the Spam Reporting (SpamRep) enabler. This architecture is based on the requirements and use cases described in the SpamRep Requirements Document [SpamRep-RD]. This document describes the functional entities that comprise the enabler, the interfaces between those entities, and the information flow between them.

2. References

2.1 Normative References

- [OSE] “OMA Service Environment”, Open Mobile Alliance™,
URL: <http://www.openmobilealliance.org/>
- [RFC2119] “Key words for use in RFCs to Indicate Requirement Levels”, S. Bradner, March 1997,
URL: <http://www.ietf.org/rfc/rfc2119.txt>
- [SpamRep-RD] “SpamRep Requirements”, Open Mobile Alliance™, OMA-RD-SpamRep-V1_0,
URL: <http://www.openmobilealliance.org/>

2.2 Informative References

- [OMADICT] “Dictionary for OMA Specifications”, Version x.y, Open Mobile Alliance™,
OMA-ORG-Dictionary-Vx_y, URL: <http://www.openmobilealliance.org/>
- [SEC_CF] “OMA Application Layer Security Common Functions V1.0”, Open Mobile Alliance™,
OMA-ERP-SEC_CF-V1_0-20080902-A.zip, URL: <http://www.openmobilealliance.org/>

3. Terminology and Conventions

3.1 Conventions

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].

All sections and appendixes, except “Scope” and “Introduction”, are normative, unless they are explicitly indicated to be informative.

3.2 Definitions

By-Fingerprint	See [SpamRep-RD]
By-Value	See [SpamRep-RD]
Content	See [SpamRep-RD]
Fingerprint	See [SpamRep-RD]
Network Spam Box	See [SpamRep-RD]
Reporter	See [SpamRep-RD]
Spam	See [SpamRep-RD]
SpamRep Client	See [SpamRep-RD]
Spam Report	See [SpamRep-RD]
SpamRep Server	See [SpamRep-RD]
User	See [OMADICT]

3.3 Abbreviations

OMA	Open Mobile Alliance
------------	----------------------

4. Introduction

(Informative)

The SpamRep enabler provides a mechanism whereby a Reporter can designate received content as “spam” and provide a report to a network entity. The purpose for this enabler is twofold. First, it provides Service Providers the relevant information they need to analyze and address abuse in their networks and keep their network-based anti-abuse elements, such as content filters and blacklists, up-to-date with the latest information necessary to prevent proliferation of messaging abuse. Second, allowing Users to proactively participate in fighting messaging abuse gives them a sense of satisfaction and an appreciation of progress as unwanted and unsolicited messages are progressively reduced.

The scope of the SpamRep enabler is intentionally narrow and includes only the message transfer between the mobile device and network entity. The architecture consists of a SpamRep Client, a server in the network, and an interface between them. Client and server functionality beyond that associated with message transfer between the two is out of scope.

4.1 Version 1.0

The SpamRep enabler defines a common mechanism for Users to report content received by various means as Spam. It specifies types and format of messages sent by the SpamRep Client autonomously or on behalf of the User to the SpamRep Server, SpamRep Server responses and the underlying transport(s) used for the message exchange.

5. Architectural Model

The architecture model of the SpamRep enabler is based on the requirements defined in [SpamRep-RD]. It is very simple and consists of only a SpamRep Server and a SpamRep client. The SpamRep Server exposes a single interface. The SpamRep Client uses this interface.

SpamRep Server may interact with various messaging systems (e.g. e-mail, SMS, MMS etc.), but what those specific systems are and how the interaction is conducted are outside of the scope of the enabler.

SpamRep Server may interact with various spam filtering and reporting systems, but what those specific systems are and how the interaction is conducted are outside of the scope of the enabler.

SpamRep Client may interact with various messaging clients (e.g. e-mail, SMS, MMS etc.), but what those specific clients are and how the interaction are conducted is outside of the scope of the enabler.

5.1 Dependencies

SpamRep Enabler does not depend on any other OMA enabler or external functional entity.

5.2 Architectural Diagram

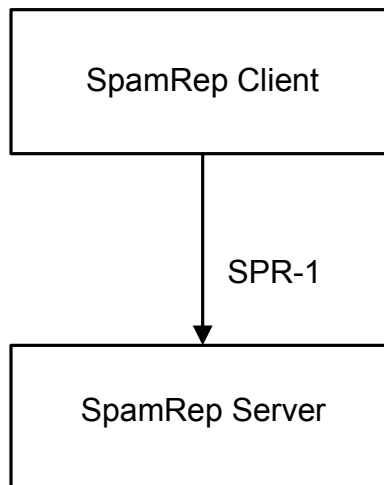


Figure 1: SpamRep Architectural Diagram

5.3 Functional Components and Interfaces/reference points definition

5.3.1 SpamRep Components

5.3.1.1 SpamRep Client

The SpamRep Client implements the functionality necessary to communicate with a SpamRep Server. Users and possibly various messaging systems interact with the SpamRep Client in order to submit a Spam Report or request an action. The SpamRep Client may be implemented in a mobile device, but messaging servers may implement it as well in order to communicate with a SpamRep Server.

SpamRep Client provides the following functionality:

- Applies policies to determine applicable Spam Report settings;
- Creates User-initiated or autonomous Spam Reports and sends them to the SpamRep Server;
- Provides additional information about the Spam message when requested by the SpamRep Server;
- Assembles and sends action requests (e.g. Block/Unblock Sender, Release Quarantined Messag, Opt Out) to the SpamRep Server;
- Queries the SpamRep Server about the status of Spam Reports;
- Queries the SpamRep Server about messages quarantined by messaging enablers in the network;
- Receives responses from the SpamRep Server;

5.3.1.2 SpamRep Server

SpamRep Server is a network entity which receives Spam Reports and action requests from SpamRep Clients. It may interact with various messaging and anti-spam systems.

SpamRep Server provides the following functionality:

- Applies policies to determine applicable Spam Report handling;
- Receives and validates Spam Reports;
- Processes action requests and responds to queries about Spam Report status;
- Responds to queries about quarantined messages;
- Notifying the Reporter the status of spam report automatically (if specified by policy);
- Possibly forwarding and sharing of Spam Reports.

5.3.2 SpamRep Interfaces

5.3.2.1 Interface SPR-1: SpamRep Server

The SPR-1 interface is exposed by the SpamRep Server. It provides the following functions:

- Sending of Spam Reports and retrieving of information about the Spam messages;
- Requesting actions;
- Querying the status of Spam Reports;

- Querying about quarantined messages.

5.3.3 External Components Used by SpamRep Enabler (Informative)

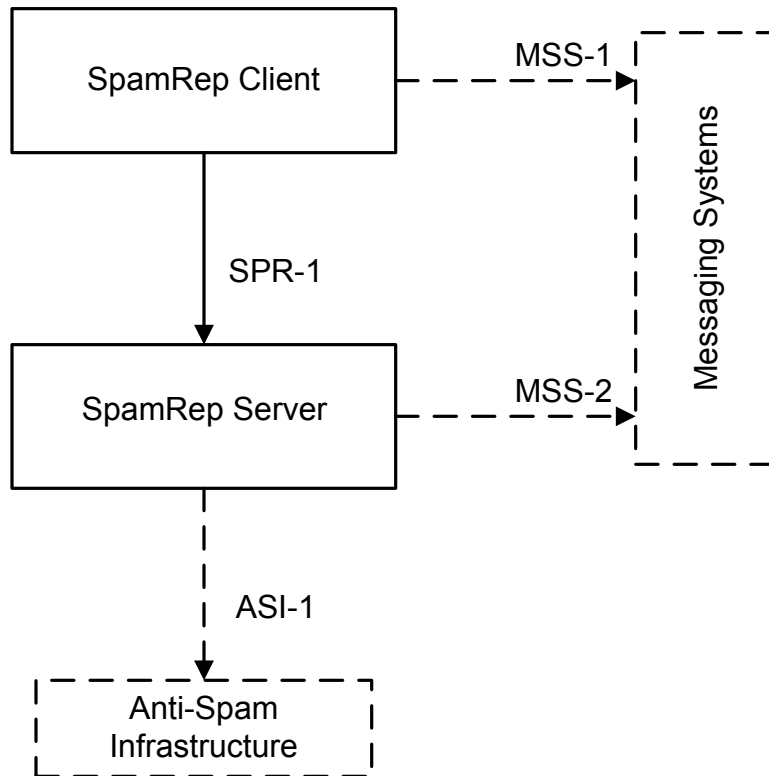


Figure 2: SpamRep Architectural Diagram with external components and interfaces (dashed)

5.3.3.1 Messaging Systems

Messaging Systems represent all the external systems used for sending, receiving and managing various kinds of messages (e.g. email, SMS, MMS etc.). Each of the Messaging Systems typically is comprised of both server and client components, but since they are entirely out of scope of the SpamRep enabler they are represented in a simplified fashion.

Both SpamRep Server and SpamRep Client may interact with Messaging Systems in order to accomplish Spam Reporting tasks. Messaging Systems may provide the following functionality to the SpamRep enabler:

- Provide messages deemed as Spam for inclusion into Spam Reports either By-Value, By-Reference or By-Fingerprint;
- Perform actions (e.g. Block/Unblock Sender, Opt Out) on behalf of the User/Reporter, as relayed through the SpamRep Server;
- May deliver the Spam Report;
- Provide part of the message deemed spam to SpamRep Client for integration into a Spam Report.

5.3.3.2 Anti-Spam Infrastructure

Anti-Spam Infrastructure represents the external systems used to filter quarantine and report Spam. Some of those systems may be fully integrated with the Messaging Systems, but logically they are a separate entity and they provide the following functionality to the SpamRep Enabler:

- Accept Spam Report information from the SpamRep Server;
- Provide status information to the SpamRep Server;
- Analyze the Spam Report to determine if it is spam.

5.3.4 External Interfaces Used by SpamRep Enabler (Informative)

5.3.4.1 Interface MSS-1: Messaging Systems

MSS-1 interface is exposed by Messaging Systems. This interface is used by SpamRep Client in order to obtain messages for inclusion into Spam Reports. It provides the following functionality:

- Returns messages for inclusion into Spam Reports By-Value, By-Reference or By-Fingerprint.

5.3.4.2 Interface MSS-2: Messaging Systems

MSS-2 interface is exposed by Messaging Systems. This interface is used by SpamRep Server in order to request actions. It provides the following functionality:

- Accepts requests and returns responses related to actions (e.g. Block/Unblock Sender, Release Quarantined Message, Opt Out);
- Accepts requests and returns responses related to retrieving list of quarantined messages.

5.3.4.3 Interface ASI-1: Anti-Spam Infrastructure

ASI-1 interface is exposed by Anti-Spam Infrastructure. This interface is used by SpamRep Server in order to report Spam. It provides the following functionality:

- Accept information from SpamRep Server;
- Answer queries about status.

5.4 Security Considerations

The primary purpose of the SpamRep enabler is to provide a mechanism to assist in the battle against mobile messaging abuse. Some messaging abuse is thought to be motivated by pure maliciousness, but the vast majority is known to be motivated by money. Email spam, for example, is known to be a highly lucrative enterprise for the message originators. As such, there is a strong motivation on the part of the originators of abusive messages to disrupt spam-control mechanisms in any way possible. An important security consideration in the SpamRep enabler, therefore, is the ability to prevent denial-of-service attacks. One way to do this is to ensure that only authorized entities may issue Spam Reports.

Another key security consideration is protection of the Spam Reporter's privacy. In a typical usage scenario a User would voluntarily issue a Spam Report in response to receiving an unsolicited and unwanted message. This message would be transmitted to the SpamRep Server in the operator's network, and what is subsequently done with that message, including possibly sharing the message with another network operator, is out of scope of the SpamRep enabler. Messages must be

transmitted in a way that preserves the integrity of their contents and protects the identity of the Reporter to ensure that Users will be inclined to use the service.

Security mechanisms that may be employed to prevent denial-of-service attacks and preserve Reporter privacy may include:

- Authentication of SpamRep Client
- Authentication of SpamRep Server
- Authorization of Reporters
- Message integrity protection
- Message confidentiality protection

Suitable authentication, message integrity and confidentiality protection mechanisms MAY be found in [SEC_CF].

Appendix A. Change History

(Informative)

A.1 Approved Version History

Reference	Date	Description
n/a	n/a	No prior version –or- No previous version within OMA

A.2 Draft/Candidate Version 1.0 History

Document Identifier	Date	Sections	Description
Draft Versions	24 Jun 2009	1, 2.1, 4	Initial skeleton draft
OMA-AD-SpamRep-V1_0	06 Aug 2009	3.2, 4.1, 5, 5.2	Incorporates input to committee: OMA-MWG-SpamRep-2009-0006R01-CR_Definitions OMA-MWG-SpamRep-2009-0007-CR_ModelDiagram
	11 Sep 2009	5.3, 5.4	Incorporates input to committee: OMA-MWG-SpamRep-2009-0008R03-CR_ComponentsIntf OMA-MWG-SpamRep-2009-0016-CR_AD_Security_Intro OMA-MWG-SpamRep-2009-0017R01-CR_Additional_Functionalities
	01 Oct 2009	2.2, 5.4, 5.3.1.1, 5.3.1.2	Incorporates input to committee: OMA-MWG-SpamRep-2009-0015R01-CR_AD_Security_Considerations OMA-MWG-SpamRep-2009-0026R01-CR_AD_Quarantined_Message_List_Retrieval_Function
	15 Oct 2009	Appendix B.	Incorporates input to committee: OMA-MWG-SpamRep-2009-0040R01-CR_AD_Call_Flow_Generic OMA-MWG-SpamRep-2009-0032-CR_AD_Call_Flow_By_Fingerprint_variant_corpus_request
	16 Oct 2009	Appendix B.	Incorporates input to committee: OMA-MWG-SpamRep-2009-0030R02-Call_Flow_for_Inquiring_about_the_status_of_Spam_Report
	20 Oct 2009	Appendix B.	Incorporates input to committee: OMA-MWG-SpamRep-2009-0029R05-INP_Notifying_Spam_Report_Status_Call_Flow OMA-MWG-SpamRep-2009-0046R01-CR_Call_Flow_Release_Quarantine
	30 Oct 20 09	5.1	Incorporates input to committee: OMA-MWG-SpamRep-2009-0047-CR_Dependencies
	04 Dec 2009	5.3.1.1, 5.3.1.2, 5.3.2.1, 2.1, 3.2, Appendix B, B.4, 5.3.4.2, 5.4, 5.2, 5.3.3	Incorporates input to committee: OMA-MWG-SpamRep-2009-0053R01-CR_Server2ClientRequest OMA-MWG-SpamRep-2009-0054R01-CR_ServerIntfAdd OMA-MWG-SpamRep-2009-0056-CR_Refs_Defs_Fix OMA-MWG-SpamRep-2009-0057R01-CR_AppendixB_Fix OMA-MWG-SpamRep-2009-0058R01-CR_Fix_for_Section_5.3_5.4 OMA-MWG-SpamRep-2009-0059-CR_NewArchDiagram
	18 Dec 2009	4, 5	Incorporates input to committee: OMA-MWG-SpamRep-2009-0062-CR_ADRR_Comments_A04_A07
	15 Jan 2010	5.3.4.1, 5.3.3.1, 5.3.1.2	Incorporates editorial changes as requested in comment A033 from OMA-ADRR-SpamRep-V1_0-20091218-D
	09 Feb 2010	4.0, 5.0, 5.3.2.1	Incorporates input to committee: OMA-MWG-SpamRep-2010-0010-CR_ADRR_Comments_A005_A006_A019
	03 Mar 2010	B5, 5.3.1.1, 5.3.1.2	Incorporates input to committee: OMA-MWG-SpamRep-2009-0024R01-INV_SpamRep_Client_reports_Spam_message_Call_Flow OMA-MWG-SpamRep-2010-0031-CR_ADRR_Comments_A010_A015_A016
Candidate version: OMA-AD-SpamRep-V1_0	20 Apr 2010	All	Status changed to Candidate by TP: OMA-TP-2010-0172-INP_SpamRep_V1_0_AD_for_Candidate_approval Editorial fixes: 2010 template

Appendix B. Flows (informative)

This informative section presents several examples of SpamRep call flows. SpamRep call flows occur within the context of a session. Figure 3 illustrates a generic SpamRep session. A session consists of one or more command-response transactions. Transactions begin with a command being sent by the SpamRep Client to the SpamRep Server, and conclude with a single SpamRep Server response. A SpamRep Client or Server may perform any number of unspecified actions (e.g., forwarding a Spam Report) before, during, and/or after a transaction. Note that a session also may include optional flows which establish authentication and/or encryption.

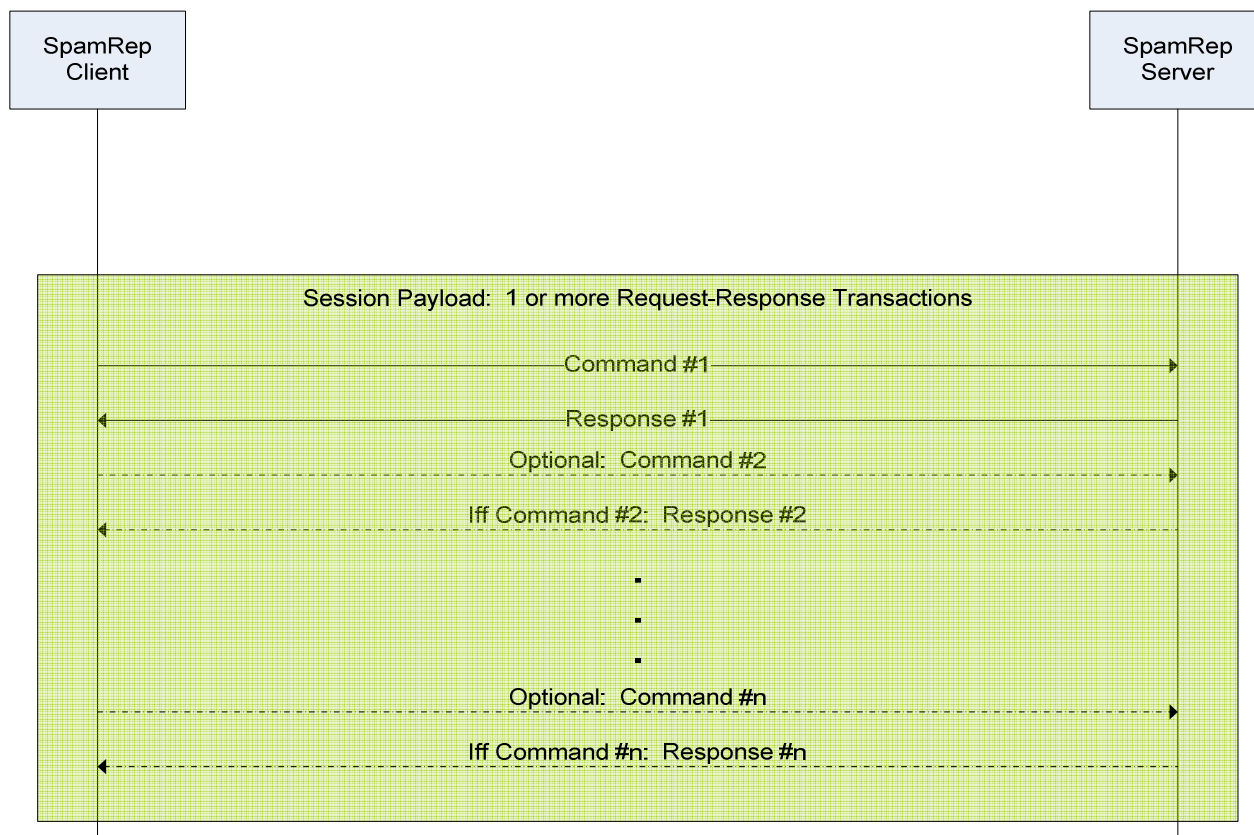


Figure 3: Call Flow of a Generic SpamRep Session

The remainder of this section presents several SpamRep transactions as examples. These examples are not the complete set of SpamRep transactions, nor do they illustrate all possible variants of these transactions.

B.1 Call flow for Report-by-Fingerprint (variant – Corpus Request)

Role/Where Used

The “By-Fingerprint” report method is used to save network bandwidth and/or server resources by representing a message as one or more compact “fingerprints.” However, in some cases (such as a previously-unseen fingerprint) and anti-spam infrastructure architectures, fingerprints may not provide enough information and it may be necessary for the SpamRep

Client to send the entire message. This may occur as an exception in situations where, after the SpamRep Client has reported Spam using the “By-Fingerprint” method, the SpamRep Server responds with a request that the SpamRep Client transmit the full message corpus.

Detailed Flow

1. The SpamRep Client sends a “By-Fingerprint” spam report to the SpamRep Server.
2. The SpamRep Server requests that the SpamRep Client transmit the full message to the SpamRep Server.
3. The SpamRep Client sends a spam report using a different method (e.g., with full message corpus) to the SpamRep Server.
4. The SpamRep Server replies with a status message (e.g., indicating success/failure and optionally any other appropriate parameters).

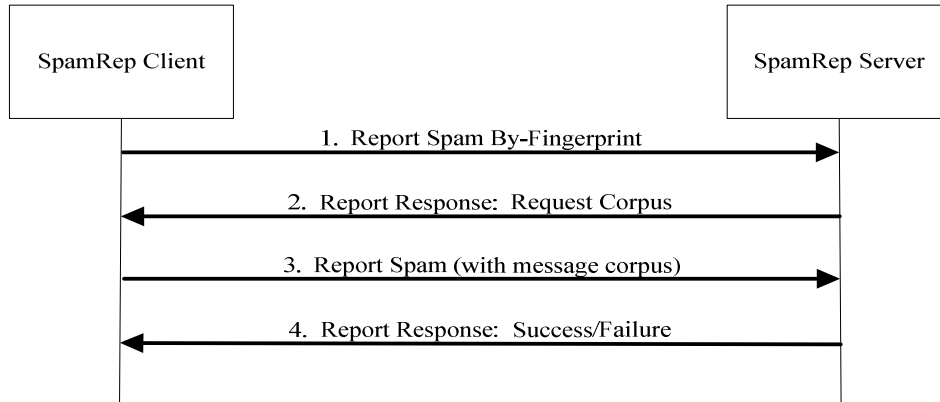


Figure 4: Report Spam By-Fingerprint with Corpus Request

B.2 Call flow for User inquiring about the Spam Report status

This call flow is triggered when the SpamRep Client has reported a Spam message and wishes to inquire about the status of the Spam Report.

1. SpamRep Client sends a request inquiring about the status of the Spam Report to the SpamRep Server. The request may contain some contextual information such as:
 - Information that identifies the Spam Report
 - Information that identifies the User.
2. SpamRep Server processes the request from the SpamRep Client and provides a response to the SpamRep Client. The response message contains:
 - the status of the Spam Report (e.g. processed or not, type of abuse, etc)

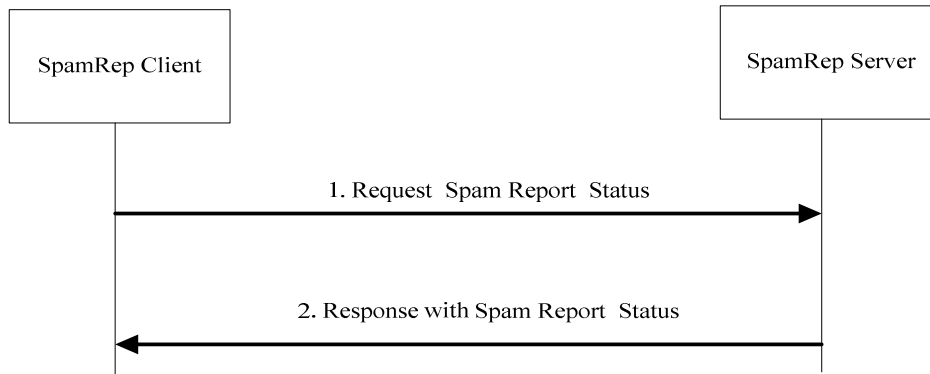


Figure 5: Request for Spam Report Status Call Flow

B.3 SpamRep Server notifying Spam Report status Call Flow

This call flow is triggered when the Spam Report has been processed and per policy. This may be done in different ways. For example, SpamRep Server can send the status notification message using the same messaging (e.g., SMS) system which delivered the offending message; Additionally, if SpamRep Client supports it, when there is an existing session between SpamRep Client and SpamRep Server, the SpamRep Server can send the notification message in the same session.

1. SpamRep Server notifies the user of the Spam Report status.

The Notification may contain some contextual information such as:

- The status of Spam Report (e.g., type of abuse)

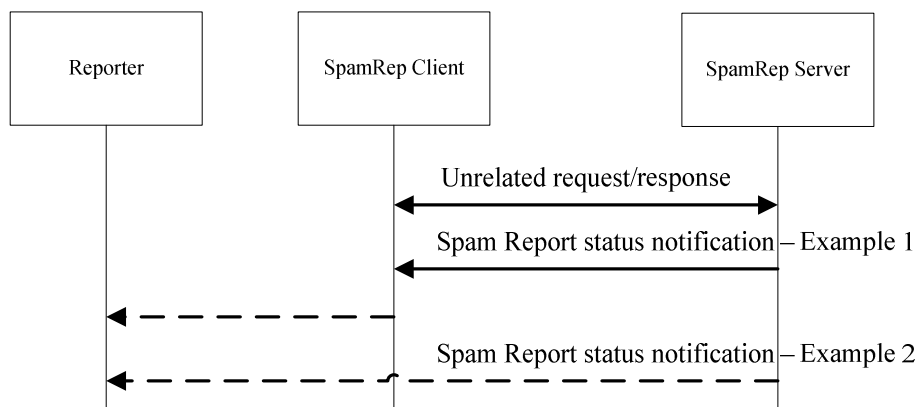


Figure 6: Notifying Spam Report's Status Call Flow – two examples

B.4 Call flow for Release Quarantine

The call flow presented in this section applies to all other action requests (e.g., retrieve list of quarantined messages, block senders, etc) as well. In all cases, it can reuse the interface of the Message Box of the Messaging system.

Role/Where Used

A SpamRep Client may inform the SpamRep Server that it has reason to believe that the messaging infrastructure may be blocking and/or quarantining messages inappropriately, and/or that the SpamRep Server should release any quarantined messages. Note that the SpamRep Server's interpretation of this message and its ultimate decision as to whether or not to release messages falls outside of the scope of the SpamRep 1.0 enabler. However, the SpamRep Server must respond with status information, which may include an indication of messaging server actions.

Detailed Flow

1. The SpamRep Client sends a Release Quarantine request to the SpamRep Server. On receipt, the SpamRep Server may relay that request to the logical entity (e.g., Messaging Server) over MSS-2, where the quarantined messages are managed. If so decided, the SpamRep Server may identify the Spam message Box type and user's subscription, and send an "Update Spam message status" request to the Messaging System by reusing the interface of Message Box (e.g. MMS Box, PoC Box). The timing of this release may be prior to, during, and/or after Step 2. The Messaging System responds with a status message to the SpamRep Server.
2. The SpamRep Server responds with a status message, optionally indicating any actions of the Messaging System.

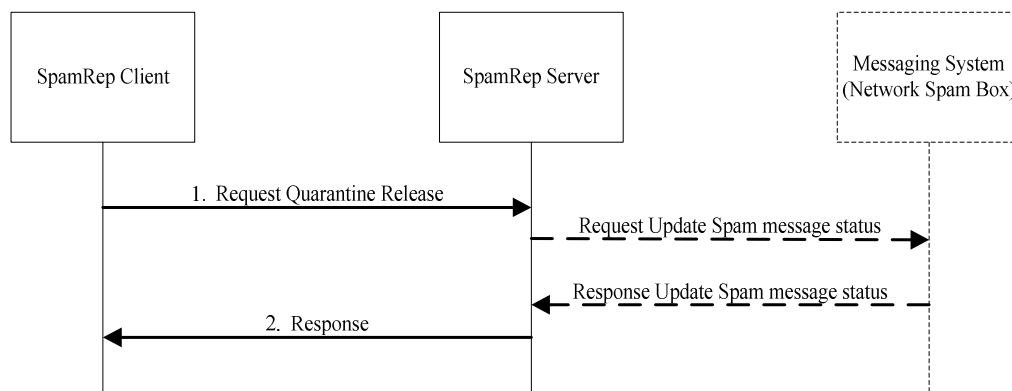


Figure 7: Request Quarantine Release

B.5 SpamRep Client Reporting Spam message Call Flow

This call flow is triggered by SpamRep user or SpamRep Client automatically on detection of a messaging abuse issue such as receiving a spam message.

1. SpamRep Client sends Spam Report to the SpamRep Server. The delivery may be By-Reference and By-Fingerprint in addition to By-Value. The report may contain some contextual information such as:
 - SpamRep Report ID
 - Date and time of a Spam Report submission
 - Type of abuse report, including: spam, phishing, malware, not spam, miscategorized, unauthorized, sender authentication failure, other, and unspecified
 - Permission to share Spam Reports with third parties

- List of third parties with which the SpamRep Server is allowed to share the Spam Report
- Content from the original message deemed abusive by the Reporter
- Data describing the delivery path of the abusive message
- Data that identifies the actual or purported originating address of the abusive message

SpamRep Server receives the report and identifies it's a Spam report. SpamRep Server may optionally processes the report (Inspect the Spam Report and analyse the obtained spam message content to identify if it is a spam message), and may possibly forward and share the Spam Report before and/or after the next step.

2. SpamRep Server responds to the SpamRep Client with the success or failure of the report and other information, optionally including disposition of the report.

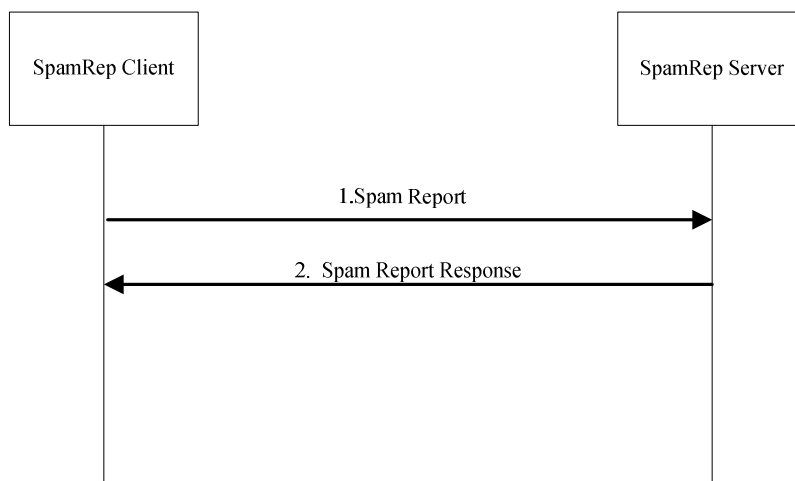


Figure 8: SpamRep Client reporting Spam message Call Flow