Network Working Group D. Wilson
Internet-Draft A. Melnikov, Ed.
Intended status: Informational Isode Ltd
Expires: December 3, 2018 June 1, 2018

Multicast Email (MULE) over ACP 142
draft-melnikov-email-over-pmul-05

Abstract

ACP 142 defines P_MUL, which is a protocol for reliable multicast suitable for bandwidth constrained and delayed acknowledgement (Emissions Control or "EMCON") environments running over UDP. This document is a specification of the basic protocol for electronic mail transfer over ACP 142 to enable MTA to MTA transfer. It explains how MULE can be used in conjunction with Simple Mail Transfer Protocol (RFC 5321), including some common SMTP extensions, to provide an alternate MTA to MTA transfer mechanism.

Table of Contents

1. Introduction

P_MUL [ACP142A] is a transport protocol for reliable multicast in bandwidth constrained and delayed acknowledgement environments running on top of UDP. The objectives of this protocol are first to take advantage of the bandwidth saving feature of using the multicast service as supported by modern computer networks and second to allow message transfer under EMCON conditions. EMCON (Emission Control) or "Radio Silence" means that, although receiving nodes are able to receive messages, they are not able to acknowledge the receipt of messages.

The objective of this protocol is to take advantage of multicast communication for the transfer of messages between MTAs (Message Transfer Agents) on a single multicast network under normal - which means dialogue oriented - communication condition and under EMCON condition. EMCON condition means that a receiving node is able to receive messages, but it cannot - for a relatively long time (hours or even days) - acknowledge the received messages.

This illustrates a simple multicast scenario, where the same message has to be sent from MTA A (through G/W) to MTA 1, MTA 2, MTA 3 and MTA 4.

```
                          +-------+                  +-------+
                          | MTA 1 |<-\            /->| MTA 3 |
 +-------+     +-----+    +-------+   \ +-------+ /   +-------+
 | MTA A |<--->| G/W |<---------------->| Router|<
 +-------+     +-----+    +-------+   / +-------+ \   +-------+
                          | MTA 2 |<-/            \->| MTA 4 |
                          +-------+                  +-------+

                        |< -------------- MULE ---------------->|
```

Typical MULE Deployment. The gateway (G/W) and Router might or might not be running on the same system.

Figure 1

Due to multicast use (instead of a unicast communication service) in the above MTA configuration only one message transmission from the gateway to the Router is required in order to reach MTA 1, MTA 2, MTA 3 and MTA 4, instead of 4 as required with unicast. This saves the transmision of 3 message transactions and thus network bandwidth utilisation. Depending on the network bandwidth (in some radio networks less than 9.6 Kb/s) this saving can be of vital importance. The saving in bandwidth utilisation becomes even greater with every additional receiving MTA.

P_MUL employs a connectionless transport protocol to transmit messages, that guarantees reliable message transfer (through ACP 142 retransmissions), even in those cases, when for a certain period of time one or more of the receiving MTAs are not able or allowed to acknowledge completely received messages.

This protocol specification requires fixed multicast groups and a well known knowledge at each participating node (MTA) about the group memberships in one or more multicast groups of each participating node. Membership in multicast groups needs to be established before MULE messages can be sent.

This document defines application protocol MULE (Multicast Email) for transferring Internet Mail messages [RFC5322] over ACP 142 P_MUL.

2. Conventions Used in This Document

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].

This document is also using terminology from [RFC5321] and [RFC5598].

3. MULE

MULE is an electronic mail transport of Internet messages [RFC5322] over ACP 142 P_MUL network. It provided service similar to MTA-to-MTA SMTP [RFC5321]. This document doesn't define a service similar to Message Submission (RFC 6409).

An important feature of MULE is its capability to transport mail across multiple networks, usually referred to as "MULE mail relaying". A network consists of the mutually-ACP142-accessible nodes. Using MULE, a process can transfer mail to another process on the same ACP 142 network or to some other ACP 142 network via a relay or gateway process accessible to both networks.

MULE reuses ESMTP extension framework defined in [RFC5321]. MULE servers MUST support the following ESMTP extensions: DSN [RFC3461], SIZE [RFC1870], 8BITMIME [RFC6152], MT-PRIORITY [RFC6710], DELIVERBY [RFC2852], BINARYMIME and CHUNKING [RFC3030]. (As the message content size can always be determined from the compression wrapper and the size of the envelope, no special handling is needed for binary messages.)

Relaying a message using MULE is performed as follows:

1. The message is reassembled from one or more DATA_PDUs [ACP142A].

2. If the contentType-ShortForm value is 25, the BSMTP-like payload is extracted from compressedContent field and uncompressed as specified in Section 3.2. If the contentType-ShortForm value is not 25, it is handled as described in [ACP142A]. This document doesn't discuss further any cases where contentType-ShortForm value is not 25.

3. The list of recipients is extracted from RCPT-lines (see Section 3.1). If the receiving node is not responsible (directly or inderectly) for any of the recipients, the message is discarded and no further processing is done.

4. The relay adds trace header fields, for example the Received header field. See Section 4.4. of [RFC5321] and [RFC7601].

5. The set of ACP 142 destinations for the message is created by extracting right hand sides (hostnames) of each RCPT-line, eliminating duplicates and then converting each hostname into next ACP 142 destination using static configuration.

6. For each unique ACP 142 destination, the following steps are performed:

   A. A new BSMTP-like payload is formed, as described in Section 3.1, which only contains RCPT-lines that correspond to recipients that can receive mail through the ACP 142 destination.

   B. The created payload is compressed and encoded as specified in Section 3.2.

   C. The compressed payload is sent by P_MUL as a series of Address_PDU and one or more DATA_PDUs. When the message has an associated MT-PRIORITY value [RFC6710], the MappedPriority(value) is included as the Priority field of corresponding ACP 142 PDUs, including Address_PDU, DATA_PDUs, DISCARD_MESSAGE_PDU. Here MappedPriority(x) is defined as "6 - x".

3.1. BSMTP-like Payload construction

MULE uses BSMTP-like payload which differs from BSMTP [RFC2442]. As with BSMTP, ESMTP capability negotiation is not used, since receiver EMCON restrictions prohibit such real-time interaction. For that reason, there is no point in including EHLO capabilities. "MAIL FROM:" and "RCPT TO:" prefixes can also be eluded in order to save a few bytes.

For each received message, the corresponding BSMTP-like payload is constructed as follows (Lines are terminated using CR LF).:

The first line is what would be used for the data following "MAIL FROM:" in the SMTP dialogue. I.e. it contains the return-path address, within <>'s followed by any ESMTP extension parameters to the MAIL FROM command.

After that, there is a separate line for each recipient of the message. The value is what would follow "RCPT TO:" in the SMTP dialogue, i.e. the recipient address within <>'s followed by any ESMTP extension parameters to the corresponding RCPT TO command.

The list of recipients is terminated by an empty line (i.e. just CR LF)

The message content follows the empty line. There is no need for transparency ("dot stuffing") or terminating with a sequence "CR LF . CR LF", as the end of the message content is indicated by the end of the data (See Section 3.2 for more details).

An example of BSMTP-like payload follows

```
<from@example.com> MT-PRIORITY=4 BODY=8BITMIME RET=HDRS ENVID=QQ314159
<to1@example.net> NOTIFY=SUCCESS,FAILURE
ORCPT=rfc822;Bob@enterprise.example.net
<to2@example.net> NOTIFY=SUCCESS,FAILURE

From: from@example.com
To: To1 <to1@example.net>, To2 <to2@example.net>
Date: 27 Apr 2017 16:17 +0100
Subject: a test
MIME-Version: 1.0
Content-type: text/plain; charset=utf-8
Content-transfer-encoding: 8bit

This is worth <poundsign>100
```

ABNF [RFC5234] for the BSMTP-like payload is:

```
bsmtp-like-payload = envelope CRLF payload
envelope = FROM-line 1*RCPT-line
FROM-line = reverse-path [SP mail-parameters] CRLF
RCPT-line = forward-path [SP rcpt-parameters] CRLF

payload = *OCTET
          ; Conforms to message syntax as defined in RFC 5322 and extended in
MIME

OCTET = <any 0-255 octet value>
reverse-path = <as defined in RFC 5321>
forward-path = <as defined in RFC 5321>
mail-parameters = <as defined in RFC 5321>
rcpt-parameters = <as defined in RFC 5321>
```

3.2. Payload compression

BSMTP-like payload (Section 3.1) is first compressed using zlibCompress [RFC1951] and the compressed payload is placed in the compressedContent field of the CompressedContentInfo element defined in Section 4.2.6 of [STANAG-4406]. This is then encoded as BER encoding [ITU.X690.2002] of the CompressedData ASN.1 structure. For convenience, the original definition of ASN.1 of the CompressedData structure is included below. The contentType-ShortForm value used by

MULE MUST be 25. (The contentType-OID alternative is never used by MULE.)

The above procedure is similar to how X.400 messages are sent using Annex E of STANAG 4406 Ed 2. This makes it easier to implement MTAs that support both Internet messages and X.400 messages in the same code base.

The Compressed Data Type (CDT) consists of content of any type that is compressed using a specified algorithm. The following object identifier identifies the Compressed Data Type:

```
            id-mmhs-CDT ID ::= { iso(1) identified-organization(3) nato(26)
stanags(0)
                                  mmhs(4406) object-identifiers(0) id-mcont(4)
2 }
```

The Compressed Data Type are defined by the following ASN.1 type:

```
DEFINITIONS ::=
BEGIN
CompressedData ::= SEQUENCE {
                     compressionAlgorithm CompressionAlgorithmIdentifier,
                     compressedContentInfo CompressedContentInfo
                   }
CompressionAlgorithmIdentifier ::= CHOICE {
                     algorithmID-ShortForm [0] AlgorithmID-ShortForm,
                     algorithmID-OID [1] OBJECT IDENTIFIER
                   }
AlgorithmID-ShortForm ::= INTEGER { zlibCompress (0) }
CompressedContentInfo ::= SEQUENCE {
                     CHOICE {
                       contentType-ShortForm [0] ContentType-ShortForm,
                       contentType-OID [1] OBJECT IDENTIFIER
                     },
                     compressedContent [0] EXPLICIT OCTET STRING
                   }
ContentType-ShortForm ::= INTEGER {
                     unidentified (0),
                     external (1), -- identified by the object-identifier
                                   -- of the EXTERNAL content
                     p1 (2),
                     p3 (3),
                     p7 (4)
                   }
END
```

3.3. Error handling

As MULE doesn't allow next hop MTA/MDA to return immediate Response Codes for FROM-line or any of recipients in RCPT-line, MTAs/MDAs that are compliant with this specification that receive a message that can't be relayed further or delivered MUST generate a non delivery DSN report [RFC6522] message which includes message/delivery-status body part [RFC3464] and submit it using MULE to the FROM-line return-path address.

MULE relays (unlike MULE MDAs) don't need to verify that they understand all FROM-line and/or RCPT-line parameters. This keeps relay-only implementations simpler and avoids the need to upgrade them when MULE MDAs are updated to support extra SMTP extensions.

4. Gatewaying from Internet Mail to MULE

A gateway from Internet Mail to MULE acts as SMTP server on the receiving side and as MULE client on the sending side.

When the content type for a message is an Internet message content type (which may be 7bit, 8bit or binary MIME), this is transported using ACP 142 [ACP142A] as follows:

1. For each mail message a BSMTP-like payload is formed, as described in Section 3.1.

2. The created payload is compressed and encoded as specified in Section 3.2.

3. The compressed payload is sent by P_MUL as a series of Address_PDU and one or more DATA_PDUs. When the message has an associated MT-PRIORITY value [RFC6710], the MappedPriority(value) is included as the Priority field of corresponding ACP 142 PDUs, including Address_PDU, DATA_PDUs, DISCARD_MESSAGE_PDU. Here MappedPriority(x) is defined as "6 - x".

The set of ACP 142 destinations for the message is derived from the next hop MTAs for each of the recipients.

4.1. Use of BDAT

If a message is received by a gateway, through SMTP transfers using the CHUNKING [RFC3030] extension, the message is rebuilt by the receiving MTA into its complete form and is then used as a single MULE message payload. Use of BINARYMIME [RFC3030] extension is conveyed by inclusion of BODY=BINARY parameter in the FROM-line.

5. Gatewaying from MULE to Internet Mail

A gateway from MULE to Internet Mail acts as a MULE server on the receiving side and as an SMTP client on the sending side.

Gatewaying from ACP 142 environment to Internet Email is the reverse of the process specified in Section 4.

1. The ACP 142 message is reassembled from one or more DATA_PDUs.

2. If the contentType-ShortForm value is 25, the BSMTP-like payload is extracted from compressedContent field and uncompressed as specified in Section 3.2. If the contentType-ShortForm value is not 25, it is handled as described in [ACP142A].

3. The BSMTP-like payload is converted to SMTP transaction (see Section 3.1). (The first line of the BSMTP-like payload is prepended with "MAIL FROM:" and each following line (until the empty line is encountered) is prepended with "RCPT TO:". After skipping the empty delimiting line, the rest of the payload is the message body. This can be either sent using DATA or a series of BDAT commands, depending on capabilities of the receiving SMTP system. For example, presence of BODY=BINARY parameter in FROM-line would necessitate use of BDAT or downconversion of the message to 7-bit compatible representation.)

5.1. Handling of ESMTP extensions and Error handling

ESMTP extension parameters to MAIL FROM and RCPT TO SMTP commands obtained from BSMTP-like payload are processed according to specifications of the corresponding ESMTP extensions, including dealing with absence of support for ESMTP extensions that correspond to MAIL FROM/RCPT TO parameters found in the BSMTP-like payload.

Failures to extract or uncompress BSMTP-like payload are handled according to ACP 142.

6. IANA Considerations

IANA is requested to create a new registry "Multicast Email SMTP extensions". Registration procedure for the new registry is "Specification Required" [RFC8126], but the registration reviewer(s) will be appointed and managed by the editors of this document together with the Independent Submissions Editor. Selected Designated Expert(s) should (collectively) have good knowledge of SMTP protocol (and its extensions/extensibility mechanisms), and ACP 142 and its limitations.

Designate Expert verifies that

1. the requested SMTP extension is already registered in the "SMTP Service Extensions" registry in the "Mail Parameters" section of the IANA Website or is well documented on a stable, publicly accessible web page.

2. the requested SMTP extension has the correct status as specified in Section 6.1. When deciding on status, the Designated Expert(s) is provided with the following guidelines:

   1. If the SMTP extension only affects commands other than MAIL FROM/RCPT TO, then the status should be "N/A".

   2. If the SMTP extension only applies to SMTP submission (and not to SMTP relay or final SMTP delivery), then the status should be "N/A".

   3. If the SMTP extension changes which commands are allowed during an SMTP transaction (e.g. if it adds commands alternative to DATA or declares commands other than MAIL FROM/RCPT TO/DATA/BDAT to be a part of SMTP transaction), then the status should be "Disallowed" or "Special".

   4. If the SMTP extension adds extra round trips during SMTP transaction, then the status should be "Disallowed" or "Special".

Registration requests should include SMTP extension name, status (see Section 6.1), specification reference and may include an optional note. (At IANA's discretion the new registry can instead be represented as an extra column in the existing "SMTP Service Extensions" registry.)

6.1. SMTP Extension Support in MULE

The following table summarizes how different SMTP extensions can be used with MULE. Each extension has one of the following statuses: "Required" (required to be supported by MULE relays, SMTP-to-MULE gateway or MULE-to-SMTP gateway), "Disallowed" (incompatible with MULE), "N/A" (not relevant, because they affect commands other than MAIL FROM and/or RCPT TO, or only defined for SMTP Submission. Such extensions can still be used on the receiving SMTP side of SMTP-to-MULE gateway) "Supported" (can be used with MULE, but requires bilateral agreement between sender and receiver), or "Special". "Special" needs to be accompanied by an explanation.

SMTP Extension Support in MULE:

SMTP Extension Keyword	Reference	Status
SIZE	[RFC1870]	Required
8BITMIME	[RFC6152]	Required
DSN	[RFC3461]	Required
MT-PRIORITY	[RFC6710]	Required
DELIVERBY	[RFC2852]	Required
BINARYMIME	[RFC3030]	Required
CHUNKING	[RFC3030]	Special (*)
ENHANCEDSTATUSCODES	[RFC2034]	Special (**)
RRVS	[RFC7293]	Supported
SUBMITTER	[RFC4405]	Supported
PIPELINING	[RFC2920]	N/A
STARTTLS	[RFC3207]	N/A
AUTH	[RFC4954]	Special (***)
BURL	[RFC4468]	N/A
NO-SOLICITING	[RFC3865]	N/A
CHECKPOINT	[RFC1845]	Disallowed
CONNEG	[RFC4141]	Disallowed

(*) - SMTP CHUNKING MUST be supported on the receiving SMTP side of a SMTP-to-MULE gateway and MAY be used on the sending side of MULE-to-SMTP gateway. MULE relay doesn't need to do anything special for this extension.

(**) - ENHANCEDSTATUSCODES extension is supported by including relevant status codes in DSN [RFC3461] reports.

(***) - The AUTH parameter to MAIL FROM command is "supported", but the rest of AUTH extension is not applicable to MULE.

Note that the above table is not exhaustive. Future RFCs can define how SMTP Extensions not listed above can be used in MULE.

6.2. SMTP Extension Support in MULE

IANA is requested to add the value "MULE" to the "WITH protocol types" subregistry of under the "Mail Transmission Types" registry. The new entry should point to this document.

7. Security Considerations

As MULE provides service similar to SMTP, many of Security Considerations from [RFC5321] apply to MULE as well, in particular Sections 7.1, 7.2, 7.4, 7.6, 7.7, 7.9 of [RFC5321] still apply to MULE.

As MULE doesn't support capability negotiation or SMTP HELP command, Section 7.5 of [RFC5321] ("Information Disclosure in Announcements") doesn't apply to MULE.

As MULE doesn't support VRFY or EXPN SMTP commands, Section 7.3 of [RFC5321] ("VRFY, EXPN, and Security") that talks about email harvesting doesn't apply to MULE.

Arguably, it is more difficult to cause application layer Denial-of-Service attack on a MULE server than on an SMTP server. This is partially due to fact that ACP 142 is used in radio/wireless networks with relatively low bandwidth and very long round trip time (especially if EMCON is in force). However, as MULE is using multicast, multiple MULE nodes can receive the same message and spend CPU processing it, even if the message is addressed to recipients that are not going to be handled by such nodes. As MULE lacks transport layer source authentication, this can be abused by malicious senders.

8. References

8.1. Normative References

[ACP142A] "Common Messaging strategy and procedures", ACP 142(A), August 2008.

[ITU.X690.2002]
International Telecommunications Union, "Information Technology - ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)", ITU-T Recommendation X.690, July 2002.

[RFC1870] Klensin, J., Freed, N., and K. Moore, "SMTP Service Extension for Message Size Declaration", STD 10, RFC 1870, DOI 10.17487/RFC1870, November 1995, <https://www.rfc-editor.org/info/rfc1870>.

[RFC1951] Deutsch, P., "DEFLATE Compressed Data Format Specification version 1.3", RFC 1951, DOI 10.17487/RFC1951, May 1996, <https://www.rfc-editor.org/info/rfc1951>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC2852] Newman, D., "Deliver By SMTP Service Extension", RFC 2852, DOI 10.17487/RFC2852, June 2000, <https://www.rfc-editor.org/info/rfc2852>.

[RFC3030] Vaudreuil, G., "SMTP Service Extensions for Transmission of Large and Binary MIME Messages", RFC 3030, DOI 10.17487/RFC3030, December 2000, <https://www.rfc-editor.org/info/rfc3030>.

[RFC3461] Moore, K., "Simple Mail Transfer Protocol (SMTP) Service Extension for Delivery Status Notifications (DSNs)", RFC 3461, DOI 10.17487/RFC3461, January 2003, <https://www.rfc-editor.org/info/rfc3461>.

[RFC3464] Moore, K. and G. Vaudreuil, "An Extensible Message Format for Delivery Status Notifications", RFC 3464, DOI 10.17487/RFC3464, January 2003, <https://www.rfc-editor.org/info/rfc3464>.

[RFC5234] Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, DOI 10.17487/RFC5234, January 2008, <https://www.rfc-editor.org/info/rfc5234>.

[RFC5321] Klensin, J., "Simple Mail Transfer Protocol", RFC 5321, DOI 10.17487/RFC5321, October 2008, <https://www.rfc-editor.org/info/rfc5321>.

[RFC5322] Resnick, P., Ed., "Internet Message Format", RFC 5322, DOI 10.17487/RFC5322, October 2008, <https://www.rfc-editor.org/info/rfc5322>.

[RFC5598] Crocker, D., "Internet Mail Architecture", RFC 5598, DOI 10.17487/RFC5598, July 2009, <https://www.rfc-editor.org/info/rfc5598>.

[RFC6152] Klensin, J., Freed, N., Rose, M., and D. Crocker, Ed., "SMTP Service Extension for 8-bit MIME Transport", STD 71, RFC 6152, DOI 10.17487/RFC6152, March 2011, <https://www.rfc-editor.org/info/rfc6152>.

[RFC6522] Kucherawy, M., Ed., "The Multipart/Report Media Type for the Reporting of Mail System Administrative Messages", STD 73, RFC 6522, DOI 10.17487/RFC6522, January 2012, <https://www.rfc-editor.org/info/rfc6522>.

[RFC6710] Melnikov, A. and K. Carlberg, "Simple Mail Transfer Protocol Extension for Message Transfer Priorities", RFC 6710, DOI 10.17487/RFC6710, August 2012, <https://www.rfc-editor.org/info/rfc6710>.

[RFC7601] Kucherawy, M., "Message Header Field for Indicating Message Authentication Status", RFC 7601, DOI 10.17487/RFC7601, August 2015, <https://www.rfc-editor.org/info/rfc7601>.

[RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <https://www.rfc-editor.org/info/rfc8126>.

[STANAG-4406] "STANAG 4406 Edition 2: Military Message Handling System", STANAG 4406 Ed. 2, March 2005.

8.2. Informative References

[RFC1845] Crocker, D., Freed, N., and A. Cargille, "SMTP Service Extension for Checkpoint/Restart", RFC 1845, DOI 10.17487/RFC1845, September 1995, <https://www.rfc-editor.org/info/rfc1845>.

[RFC2034] Freed, N., "SMTP Service Extension for Returning Enhanced Error Codes", RFC 2034, DOI 10.17487/RFC2034, October 1996, <https://www.rfc-editor.org/info/rfc2034>.

[RFC2442] Freed, N., Newman, D., Belissent, J., and M. Hoy, "The Batch SMTP Media Type", RFC 2442, DOI 10.17487/RFC2442, November 1998, <https://www.rfc-editor.org/info/rfc2442>.

[RFC2920] Freed, N., "SMTP Service Extension for Command Pipelining", STD 60, RFC 2920, DOI 10.17487/RFC2920, September 2000, <https://www.rfc-editor.org/info/rfc2920>.

[RFC3207] Hoffman, P., "SMTP Service Extension for Secure SMTP over Transport Layer Security", RFC 3207, DOI 10.17487/RFC3207, February 2002, <https://www.rfc-editor.org/info/rfc3207>.

[RFC3865] Malamud, C., "A No Soliciting Simple Mail Transfer Protocol (SMTP) Service Extension", RFC 3865, DOI 10.17487/RFC3865, September 2004, <https://www.rfc-editor.org/info/rfc3865>.

[RFC4141] Toyoda, K. and D. Crocker, "SMTP and MIME Extensions for Content Conversion", RFC 4141, DOI 10.17487/RFC4141, November 2005, <https://www.rfc-editor.org/info/rfc4141>.

[RFC4405] Allman, E. and H. Katz, "SMTP Service Extension for Indicating the Responsible Submitter of an E-Mail Message", RFC 4405, DOI 10.17487/RFC4405, April 2006, <https://www.rfc-editor.org/info/rfc4405>.

[RFC4468] Newman, C., "Message Submission BURL Extension", RFC 4468, DOI 10.17487/RFC4468, May 2006, <https://www.rfc-editor.org/info/rfc4468>.

[RFC4954] Siemborski, R., Ed. and A. Melnikov, Ed., "SMTP Service Extension for Authentication", RFC 4954, DOI 10.17487/RFC4954, July 2007, <https://www.rfc-editor.org/info/rfc4954>.

[RFC7293] Mills, W. and M. Kucherawy, "The Require-Recipient-Valid-Since Header Field and SMTP Service Extension", RFC 7293, DOI 10.17487/RFC7293, July 2014, <https://www.rfc-editor.org/info/rfc7293>.

Appendix A. Acknowledgements

Thank you to Steve Kille for suggestions, comments and corrections on this document. Additional thank you goes to Barry Leiba, Sean Turner and Dave Crocker for reviews and comments on this document.

Some text was borrowed from draft-riechmann-multicast-mail-00, thus work of authors of that document is greatefully acknowledged.

Authors' Addresses

David Wilson
Isode Ltd
14 Castle Mews
Hampton, Middlesex TW12 2NP
UK

EMail: David.Wilson@isode.com

Alexey Melnikov (editor)
Isode Ltd
14 Castle Mews
Hampton, Middlesex TW12 2NP
UK

EMail: Alexey.Melnikov@isode.com