

Workgroup: JMAP
Internet-Draft: draft-ietf-jmap-sieve-07
Published: 2 May 2022
Intended Status: Standards Track
Expires: 3 November 2022
Authors: K. Murchison
Fastmail

JMAP for Sieve Scripts

Abstract

This document specifies a data model for managing Sieve scripts on a server using the JSON Meta Application Protocol (JMAP).

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 <https://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 3 November 2022.

Copyright Notice

Copyright (c) 2022 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 (<https://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 Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

- [1. Introduction](#)
 - [1.1. Notational Conventions](#)
 - [1.2. Terminology](#)
 - [1.3. Addition to the Capabilities Object](#)
 - [1.3.1. urn:ietf:params:jmap:sieve](#)
- [2. Sieve Scripts](#)
 - [2.1. SieveScript/get](#)
 - [2.1.1. Examples](#)
 - [2.2. SieveScript/set](#)
 - [2.2.1. Examples](#)
 - [2.3. SieveScript/query](#)
 - [2.4. SieveScript/validate](#)
 - [2.5. SieveScript/test](#)
 - [2.5.1. Example](#)
- [3. Compatibility with JMAP Vacation Response](#)
- [4. Security Considerations](#)
- [5. IANA Considerations](#)
 - [5.1. JMAP Capability Registration for "sieve"](#)
 - [5.2. JMAP Error Codes Registry](#)
 - [5.2.1. invalidScript](#)
 - [5.2.2. scriptIsActive](#)
- [6. Acknowledgments](#)
- [7. References](#)
 - [7.1. Normative References](#)
 - [7.2. Informative References](#)
- [Appendix A. Change History \(To be removed by RFC Editor before publication\)](#)
- [Author's Address](#)

1. Introduction

[JMAP](#) [[RFC8620](#)] (JSON Meta Application Protocol) is a generic protocol for synchronizing data, such as mail, calendars or contacts, between a client and a server. It is optimized for mobile and web environments, and aims to provide a consistent interface to different data types.

This specification defines a data model for managing [Sieve](#) [[RFC5228](#)] scripts on a server using JMAP. The data model is designed to allow a server to provide consistent access to the same scripts via [ManageSieve](#) [[RFC5804](#)] as well as JMAP, however the functionality offered over the two protocols may differ.

1.1. Notational Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and

"OPTIONAL" in this document are to be interpreted as described in BCP 14 [[RFC2119](#)] [[RFC8174](#)] when, and only when, they appear in all capitals, as shown here.

Type signatures, examples, and property descriptions in this document follow the conventions established in Section 1.1 of [[RFC8620](#)]. Data types defined in the core specification are also used in this document.

Servers MUST support all properties specified for the new data type defined in this document.

For compatibility with publishing requirements, line breaks have been inserted inside long JSON strings, with the following continuation lines indented. To form the valid JSON example, any line breaks inside a string must be replaced with a space and any other white space after the line break removed.

1.2. Terminology

The same terminology is used in this document as in the core JMAP specification, see [[RFC8620](#)], Section 1.6.

The term SieveScript (with this specific capitalization) is used to refer to the data type defined in this document and instances of those data types.

1.3. Addition to the Capabilities Object

The capabilities object is returned as part of the JMAP Session object; see [[RFC8620](#)], Section 2. This document defines one additional capability URI.

1.3.1. urn:ietf:params:jmap:sieve

This represents support for the SieveScript data type and associated API methods. The value of this property in the JMAP Session capabilities property is an empty object.

The value of this property in an account's accountCapabilities property is an object that MUST contain the following information on server capabilities:

***maxSizeScriptName:** UnsignedInt

The maximum length, in (UTF-8) octets, allowed for the name of a SieveScript. For compatibility with ManageSieve, this MUST be at least 512 (up to 128 Unicode characters).

***maxSizeScript:** UnsignedInt|null

The maximum size (in octets) of a Sieve script the server is willing to store for the user, or null for no limit.

***maxNumberScripts:** UnsignedInt|null

The maximum number of Sieve scripts the server is willing to store for the user, or null for no limit.

***maxNumberRedirects:** UnsignedInt|null

The maximum number of Sieve "redirect" actions a script can perform during a single evaluation or null for no limit. Note that this is different from the total number of "redirect" actions a script can contain.

***sieveExtensions:** String[]

A list of case-sensitive Sieve capability strings (as listed in Sieve "require" action; see [[RFC5228](#)], Section 3.2) indicating the extensions supported by the Sieve engine.

***notificationMethods:** String[]|null

A list of [URI schema parts](#) [[RFC3986](#)] for notification methods supported by the Sieve ["enotify"](#) [[RFC5435](#)] extension, or null if the extension is not supported by the Sieve engine.

***externalLists:** String[]|null

A list of [URI schema parts](#) [[RFC3986](#)] for externally stored list types supported by the Sieve ["extlists"](#) [[RFC6134](#)] extension, or null if the extension is not supported by the Sieve engine.

***supportsTest:** Boolean

If true, the server supports the [SieveScript/test](#) ([Section 2.5](#)) method.

2. Sieve Scripts

A **SieveScript** object represents a single [Sieve](#) [[RFC5228](#)] script for filtering email messages at time of final delivery.

A **SieveScript** object has the following properties:

***id:** Id (immutable; server-set)

The id of the script.

***name:** String|null (optional; default is server-dependent)

User-visible name for the SieveScript. If non-null, this MUST be a [Net-Unicode](#) [RFC5198] string of at least 1 character in length, subject to the maximum size given in the capability object. For compatibility with ManageSieve, servers MUST reject names that contain control characters. Servers MAY reject names that violate server policy (e.g., names containing slash (/)). The name MUST be unique among all SieveScripts within an account.

***blobId:** Id

The id of the blob containing the raw octets of the script.

The script MUST be [UTF-8](#) [RFC3629] content of at least 1 character in length, subject to the syntax of [Sieve](#) [RFC5228]. The script MUST NOT contain any "require" statement(s) mentioning Sieve capability strings not present in the [capability](#) ([Section 1.3.1](#)) object. Note that if the Sieve ["ihave"](#) [RFC5463] capability string is present in the capability object, the script MAY mention unrecognized/unsupported extensions in the "ihave" test.

***isActive:** Boolean (server-set; default: false)

A user may have multiple SieveScripts on the server, yet only one script may be used for filtering of incoming messages. This is the active script. Users may have zero or one active script. The [SieveScript/set](#) ([Section 2.2](#)) method is used for changing the active script or disabling Sieve processing.

2.1. SieveScript/get

This is a standard "/get" method as described in [RFC8620], Section 5.1. The *ids* argument may be null to fetch all at once.

This method provides similar functionality to the GETSCRIPT and LISTSCRIPTS commands in [RFC5804].

2.1.1. Examples

Request (and response) to list all scripts:

```

{
  "using": [ "urn:ietf:params:jmap:core",
             "urn:ietf:params:jmap:sieve" ],
  "methodCalls": [
    ["SieveScript/get", {
      "accountId": "ken",
    }, "0"]
  ]
}

{
  "methodResponses": [
    [
      "SieveScript/get",
      {
        "state": "1634915373.240633104-120",
        "list": [
          {
            "id": "2d647053-dded-418d-917a-63eda3ac8f7b",
            "name": "test1",
            "isActive": true,
            "blobId": "s123"
          }
        ],
        "notFound": [],
        "accountId": "ken"
      },
      "0"
    ]
  ]
}

```

Request (and response) to download the script (assuming that the JMAP Download URL has been advertised in the JMAP Session object as having a path of `/jmap/download/{accountId}/{blobId}/{name}?accept={type}`). Note that the request-line has been wrapped for presentation purposes only.

```
GET
/jmap/download/ken/S123/test1.siv?accept=application/sieve
HTTP/1.1
Host: jmap.example.com
Authorization: Basic a2VuOnBhc3N3b3Jk
```

```
HTTP/1.1 200 OK
Date: Fri, 22 Oct 2021 15:27:38 GMT
Content-Type: application/sieve; charset=utf-8
Content-Disposition: attachment; filename="test1.siv"
Content-Length: 49
```

```
require ["fileinto"];
fileinto "INBOX.target";
```

Request (and response) to fetch the content of a single script:

```

{
  "using": [ "urn:ietf:params:jmap:core",
            "urn:ietf:params:jmap:blob",
            "urn:ietf:params:jmap:sieve" ],
  "methodCalls": [
    ["SieveScript/get", {
      "accountId": "ken",
      "ids": [ "2d647053-dded-418d-917a-63eda3ac8f7b" ],
    }, "0"],
    ["Blob/get", {
      "accountId": "ken",
      "#ids": {
        "resultOf": "0",
        "name": "SieveScript/get",
        "path": "/list/*/blobId"
      }
    }, "1"]
  ]
}

{
  "methodResponses": [
    [
      "SieveScript/get",
      {
        "state": "1634915373.240633104-120",
        "list": [
          {
            "id": "2d647053-dded-418d-917a-63eda3ac8f7b",
            "name": "test1",
            "isActive": true,
            "blobId": "S123"
          }
        ],
        "notFound": [],
        "accountId": "ken"
      },
      "0"
    ],
    [
      "Blob/get",
      {
        "list": [
          {
            "id": "S123",
            "data:asText":
"require [\\"fileinto\\"];\\r\\nfileinto \\"INBOX.target\\";\\r\\n",
            "size": 49
          }
        ]
      }
    ]
  ]
}

```



```
    ],  
    "notFound": [],  
    "accountId": "ken"  
  },  
  "1"  
]  
}
```

2.2. SieveScript/set

This is a standard `"/set"` method as described in [\[RFC8620\]](#), Section 5.3 but with the following additional request argument, which may be omitted:

***onSuccessActivateScript:** Id|null (optional)

If null, the currently active SieveScript (if any) will be deactivated if and only if all of the creations, modifications, and destructions (if any) succeed. Otherwise, the id of the SieveScript to activate if and only if all of the creations, modifications, and destructions (if any) succeed. (For references to SieveScript creations, this is equivalent to a creation-reference, so the id will be the creation id prefixed with a `"#"`.) If this argument is not present in the request, the currently active SieveScript (if any) will remain as such.

The id of any activated SieveScript MUST be reported in either the `"created"` or `"updated"` argument in the response as appropriate. The id of any deactivated SieveScript MUST be reported in the `"updated"` argument in the response.

This method provides similar functionality to the `PUTSCRIPT`, `DELETESCRIPT`, `RENAMESCRIPT`, and `SETACTIVE` commands in [\[RFC5804\]](#).

Script content must first be uploaded as a blob using either the standard upload mechanism (see [\[RFC8620\]](#) Section 6.1) or the JMAP Blob management extension (see [\[I-D.ietf-jmap-blob\]](#) Section 3.1).

If the SieveScript can not be created or updated because it would result in two SieveScripts with the same name, the server MUST reject the request with an `"alreadyExists"` SetError. An `"existingId"` property of type `"Id"` MUST be included on the SetError object with the id of the existing SieveScript.

If the SieveScript can not be created or updated because its size exceeds the `"maxSizeScript"` limit, the server MUST reject the request with a `"tooLarge"` SetError.

If the Sieve Script can not be created because it would exceed the `"maxNumberScripts"` limit, the server MUST reject the request with an `"overQuota"` SetError.

The active SieveScript MUST NOT be destroyed unless it is first deactivated in a separate SieveScript/set method call.

The following extra SetError types are defined:

For "create" and "update":

***invalidScript:**

The SieveScript content violates the [Sieve](#) [RFC5228] grammar and/or one or more extensions mentioned in the script's "require" statement(s) are not supported by the Sieve interpreter. The *description* property on the SetError object SHOULD contain a specific error message giving at least the line number of the first error.

For "destroy":

***scriptIsActive:**

The SieveScript is active.

2.2.1. Examples

Request (and response) to upload a script requiring the Imap4Flags [RFC5232] Extension (assuming that the JMAP Upload URL has been advertised in the JMAP Session object as having a path of "/jmap/upload/{accountId}/"):

```
POST /jmap/upload/ken/ HTTP/1.1
Host: jmap.example.com
Authorization: Basic a2VuOnBhc3N3b3Jk
Content-Type: application/sieve
Content-Length: 98
```

```
require "imapflags";

if address :is ["To", "Cc"] "jmap@ietf.org" {
    setflag "\\Flagged";
}
```

```
HTTP/1.1 201 Created
Date: Thu, 10 Dec 2020 17:14:31 GMT
Content-Type: application/json; charset=utf-8
Content-Length: 171
```

```
{
  "accountId": "ken",
  "blobId": "Gabcc83e44a6e19991c4568d0b94e1767c83dd123",
  "type": "application/sieve"
  "size": 98
}
```

Request (and response) to create and activate a script using the uploaded blob:

```
{
  "using": [ "urn:ietf:params:jmap:core",
            "urn:ietf:params:jmap:sieve" ],
  "methodCalls": [
    [ "SieveScript/set", {
      "accountId": "ken",
      "create": { "A": {
        "name": null,
        "blobId": "Gabcc83e44a6e19991c4568d0b94e1767c83dd123"
      } },
      "onSuccessActivateScript": "#A"
    }, "0" ]
  ]
}

{
  "methodResponses": [
    [
      "SieveScript/set",
      {
        "oldState": "1603741717.50737918-4096",
        "newState": "1603741751.227268529-4096",
        "created": {
          "A": {
            "id": "dd1b164f-8cdc-448c-9f54",
            "name": "ken-20201210T171432-0",
            "blobId": "Sdd1b164f-8cdc-448c-9f54",
            "isActive": true
          }
        },
        "updated": null,
        "destroyed": null,
        "notCreated": null,
        "notUpdated": null,
        "notDestroyed": null,
        "accountId": "ken"
      },
      "0"
    ]
  ]
}
```

Request (and response) to update script content using the [JMAP Blob management extension](#) [[I-D.ietf-jmap-blob](#)]:

```

{
  "using": [ "urn:ietf:params:jmap:core",
            "urn:ietf:params:jmap:blob",
            "urn:ietf:params:jmap:sieve" ],
  "methodCalls": [
    ["Blob/set", {
      "accountId": "ken",
      "create": { "B": {
        "data:asText": "redirect \"ken@example.com\"\\r\\n;",
        "type": "application/sieve"
      }
    }, "1"],
    ["SieveScript/set", {
      "accountId": "ken",
      "update": { "dd1b164f-8cdc-448c-9f54": {
        "blobId": "#B"
      }
    }, "2"]
  ]
}

{
  "methodResponses": [
    [
      "Blob/set",
      {
        "oldState": null,
        "newState": "1603741700.309607123-0128",
        "created": {
          "B": {
            "id": "G969c83e44a6e10871c4568d0b94e1767c83ddeae",
            "blobId": "G969c83e44a6e10871c4568d0b94e1767c83ddeae",
            "type": "application/sieve",
            "size": 29
          }
        },
        "updated": null,
        "destroyed": null,
        "notCreated": null,
        "notUpdated": null,
        "notDestroyed": null,
        "accountId": "ken"
      },
      "1"
    ],
    [
      "SieveScript/set",
      {

```

```
    "oldState": "1603741751.227268529-4096",
    "newState": "1603742603.309607868-4096",
    "created": null,
    "updated": {
      "dd1b164f-8cdc-448c-9f54": null
    },
    "destroyed": null,
    "notCreated": null,
    "notUpdated": null,
    "notDestroyed": null,
    "accountId": "ken"
  },
  "2"
]
}
```

Request (and response) to update script name and deactivate:

```
{
  "using": [ "urn:ietf:params:jmap:core",
             "urn:ietf:params:jmap:sieve" ],
  "methodCalls": [
    ["SieveScript/set", {
      "accountId": "ken",
      "update": { "dd1b164f-8cdc-448c-9f54": {
        "name": "myscript"
      }
    },
    "onSuccessActivateScript": null
  ], "3"]
  ]
}

{
  "methodResponses": [
    [
      "SieveScript/set",
      {
        "oldState": "1603742603.309607868-4096",
        "newState": "1603742967.852315428-4096",
        "created": null,
        "updated": {
          "dd1b164f-8cdc-448c-9f54": {
            "isActive": false
          }
        },
        "destroyed": null,
        "notCreated": null,
        "notUpdated": null,
        "notDestroyed": null,
        "accountId": "ken"
      },
      "3"
    ]
  ]
}
```

Request (and response) to activate a script:

```

{
  "using": [ "urn:ietf:params:jmap:core",
             "urn:ietf:params:jmap:sieve" ],
  "methodCalls": [
    ["SieveScript/set", {
      "accountId": "ken",
      "onSuccessActivateScript": "dd1b164f-8cdc-448c-9f54"
    }, "4"]
  ]
}

{
  "methodResponses": [
    [
      "SieveScript/set",
      {
        "oldState": "1603742967.852315428-4096",
        "newState": "1603744460.316617118-4096",
        "created": null,
        "updated": {
          "dd1b164f-8cdc-448c-9f54": {
            "isActive": true
          }
        },
        "destroyed": null,
        "notCreated": null,
        "notUpdated": null,
        "notDestroyed": null,
        "accountId": "ken"
      },
      "4"
    ]
  ]
}

```

Requests (and responses) to deactivate and destroy the active script:


```

{
  "using": [ "urn:ietf:params:jmap:core",
             "urn:ietf:params:jmap:sieve" ],
  "methodCalls": [
    ["SieveScript/set", {
      "accountId": "ken",
      "onSuccessActivateScript": null
    }, "5"],
    ["SieveScript/set", {
      "accountId": "ken",
      "destroy": [ "dd1b164f-8cdc-448c-9f54" ]
    }, "6"]
  ]
}

{
  "methodResponses": [
    [
      "SieveScript/set",
      {
        "oldState": "1603744460.316617118-4096",
        "newState": "1603744637.575375572-4096",
        "created": null,
        "updated": null,
        "updated": {
          "dd1b164f-8cdc-448c-9f54": {
            "isActive": false
          }
        },
        "destroyed": null,
        "notCreated": null,
        "notUpdated": null,
        "notDestroyed": null,
        "accountId": "ken"
      },
      "5"
    ],
    [
      "SieveScript/set",
      {
        "oldState": "1603744637.575375572-4096",
        "newState": "1603744637.854390875-4096",
        "created": null,
        "updated": null,
        "destroyed": [
          "dd1b164f-8cdc-448c-9f54"
        ],
        "notCreated": null,
        "notUpdated": null,

```

```
        "notDestroyed": null,  
        "accountId": "ken"  
    },  
    "6"  
]  
}
```

2.3. SieveScript/query

This is a standard "/query" method as described in [[RFC8620](#)], Section 5.5. A *FilterCondition* object has the following properties, either of which may be omitted:

***name:** String

The SieveScript "name" property contains the given string.

***isActive:** Boolean

The "isActive" property of the SieveScript must be identical to the value given to match the condition.

The following SieveScript properties MUST be supported for sorting:

***name**

***isActive**

2.4. SieveScript/validate

This method is used by the client to verify Sieve script validity without storing the script on the server, providing similar functionality to the CHECKSCRIPT command in [[RFC5804](#)].

The method takes the following arguments:

***accountId:** Id

The id of the account to use.

***blobId:** Id

The id of the blob containing the raw octets of the script to validate, subject to the same requirements in [Section 2](#).

The response has the following arguments:

***accountId:** Id

The id of the account used for this call.

***error:** SetError|null

A "invalidScript" SetError object if the script content is invalid (see [Section 2.2](#)), or null if the script content is valid.

As with the [SieveScript/set](#) ([Section 2.2](#)) method, script content must first be uploaded as a blob using either the standard upload mechanism (see [[RFC8620](#)] [Section 6.1](#)) or the JMAP Blob management extension (see [[I-D.ietf-jmap-blob](#)] [Section 3.1](#)).

2.5. SieveScript/test

This method is used by the client to ask the Sieve interpreter to evaluate a Sieve script against a set of emails and report the actions that would be performed for each.

When calling this method the "using" property of the Request object MUST contain the capabilities "urn:ietf:params:jmap:sieve" and "urn:ietf:params:jmap:mail". The latter is required due to the use of blob ids which may reference Email objects and the use of the Envelope object, as described below.

The **SieveScript/test** method takes the following arguments:

***accountId:** Id

The id of the account to use.

***scriptBlobId:** String

The id of the blob containing the raw octets of the script to validate, subject to the same requirements in [Section 2](#).

***emailBlobIds:** Id[]

The ids representing the raw octets of the [[RFC5322](#)] messages to test against.

***envelope:** Envelope|null

Information that the Sieve interpreter should assume was present in the SMTP transaction that delivered the message when evaluating "envelope" tests. If null, all "envelope" tests MUST evaluate to false. See [Section 7](#) of [[RFC8621](#)] for the contents of the Envelope object.

***lastVacationResponse:** UTCDate|null

The UTC date-time at which the Sieve interpreter should assume that it last auto-replied to the sender of the message, or null if the Sieve interpreter should assume that it has not auto-replied to the sender.

The response has the following arguments:

***accountId:** Id

The id of the account used for this call.

***completed:** Id[Action[]]|null

A map of the blob id to a set of *Action* objects for each message successfully processed by the script, or null if none. The *Action* object has the following properties:

-action: String

The name of the Sieve action (e.g., "keep").

-taggedArgs: String[*]

An object containing any named (tagged) arguments for the action. The name MUST be the tag for the argument as given in the specification of the action (e.g., ":flags"). This may be an empty object if the action does not have any tagged arguments, or none were specified in the Sieve script (e.g., [discard](#) [RFC5228] or [ereject](#) [RFC5429] action).

-positionalArgs: *[]

An array containing any positional arguments for the action in the order as given in the specification of the action. This may be an empty array if the action does not have any positional arguments (e.g., [discard](#) [RFC5228] or [keep](#) [RFC5228] action).

***notCompleted:** Id[SetError]|null

A map of the blob id to a SetError object for each message that was not successfully processed by the script, or null if none. A "serverFail" SetError (see Section 3.6.2 of [RFC8620](#)) MUST be used to indicate a Sieve interpreter run-time error.

The JSON data type to use for each argument value is a direct mapping from its Sieve data type, per the following table:

Sieve Type	JSON Type
Number	Number
String	String
String List	String[]
tag with no value	Boolean (true)

Table 1

Recommendations for constructing the list of arguments are as follows:

- *Optional arguments in which the value is supplied by the Sieve interpreter SHOULD be included (e.g., ":from" and ":subject" arguments to the ["vacation"](#) [RFC5230] action).
- *Optional arguments in which the value is implicitly supplied by a Sieve variable SHOULD be included (e.g., "keep" or "fileinto" actions without an explicit ":flags" argument, but ["imap4flags"](#) [RFC5232] have been set on the internal variable).
- *Optional arguments in which the value is the specified default MAY be omitted.
- *Tagged arguments that are only used to determine whether the action will be executed and have no impact on the result of the action MAY be omitted (e.g., ":days" and ":addresses" arguments to the vacation action).

The following additional errors may be returned instead of the "SieveScript/test" response:

- *"invalidScript": The script content is invalid (see [Section 2.2](#)).
- *"notFound": The script referenced by the id could not be found.
- *"rateLimit": The number of recent test method calls has reached a server-defined limit.
- *"requestTooLarge": The total number of emailBlobIds exceeds the maximum number the server is willing to process in a single test method call.
- *"serverFail": The script failed preparation to be executed for some other reason.

2.5.1. Example

Assume that the following script has been created and has blob id "S123".

```
require [ "imapflags", "editheader", "vacation", "fcc" ];
setflag "$SieveFiltered";
addheader :last "X-Sieve-Filtered" "yes";
vacation :days 3 :fcc "INBOX.Sent" :flags "\\Answered" text:
Gone fishing.
;
;
```

Assume that the following email has been uploaded and assigned blob id "B456".

From: "Some Example Sender" <example@example.net>
To: ken@example.com
Subject: test email
Date: Wed, 23 Sep 2020 12:11:11 -0500
Content-Type: text/plain; charset="UTF-8"
MIME-Version: 1.0

This is a test email.

The following request executes the script against the email and provides envelope information for use by the "vacation" action.

```
{
  "using": [
    "urn:ietf:params:jmap:core",
    "urn:ietf:params:jmap:sieve",
    "urn:ietf:params:jmap:mail"
  ],
  "methodCalls": [
    [
      "SieveScript/test",
      {
        "accountId": "ken",
        "scriptBlobId": "S123",
        "emailBlobIds": [
          "B456"
        ],
        "envelope": {
          "mailFrom": {
            "email": "example@example.net",
            "parameters": null
          },
          "rcptTo": [
            {
              "email": "ken@example.com",
              "parameters": null
            }
          ]
        },
        "lastVacationResponse": null
      },
      "R1"
    ]
  ]
}
```

The following response lists the actions that would be performed by the script.

```
{
  "methodResponses": [
    [
      "SieveScript/test",
      {
        "completed": {
          "B456": [
            {
              "action": "addheader",
              "taggedArgs": {
                ":last": true
              },
              "positionalArgs": [ "X-Sieve-Filtered", "yes" ]
            },
            {
              "action": "vacation",
              "taggedArgs": {
                ":fcc": "INBOX.Sent",
                ":flags": [
                  "\\answered"
                ],
                ":subject": "Auto: test email",
                ":from": "ken@example.com"
              },
              "positionalArgs": [ "Gone fishing." ]
            },
            {
              "action": "keep",
              "taggedArgs": {
                ":flags": [
                  "$SieveFiltered"
                ]
              },
              "positionalArgs": [ ]
            }
          ]
        },
        "notCompleted": null,
        "accountId": "ken",
      },
      "R1"
    ]
  ]
}
```


3. Compatibility with JMAP Vacation Response

Section 8 of [[RFC8621](#)] defines a VacationResponse object to represent an autoresponder to incoming email messages. Servers that implement the VacationResponse as a Sieve script that resides amongst other user scripts are subject to the following requirements:

- *MUST allow the VacationResponse Sieve script to be fetched by the [SieveScript/get](#) ([Section 2.1](#)) method.

- *MUST allow the VacationResponse Sieve script to be [de]activated via the "onSuccessActivateScript" argument to the [SieveScript/set](#) ([Section 2.2](#)) method.

- *MUST NOT allow the VacationResponse Sieve script to be destroyed or have its content updated by the [SieveScript/set](#) ([Section 2.2](#)) method. Any such request MUST be rejected with a "forbidden" SetError. A "description" property MAY be present with an explanation that the script can only be modified by a VacationResponse/set method.

4. Security Considerations

All security considerations of [JMAP](#) [[RFC8620](#)] and [Sieve](#) [[RFC5228](#)] apply to this specification.

5. IANA Considerations

5.1. JMAP Capability Registration for "sieve"

IANA will register the "sieve" JMAP Capability as follows:

Capability Name: urn:ietf:params:jmap:sieve

Specification document: this document

Intended use: common

Change Controller: IETF

Security and privacy considerations: this document, [Section 4](#)

5.2. JMAP Error Codes Registry

The following sub-sections register two new error codes in the JMAP Error Codes registry, as defined in [[RFC8620](#)].

5.2.1. `invalidScript`

JMAP Error Code: `invalidScript`

Intended use: common

Change controller: IETF

Reference: This document, [Section 2.2](#)

Description: The SieveScript violates the [Sieve grammar](#) [RFC5228] and/or one or more extensions mentioned in the script's "require" statement(s) are not supported by the Sieve interpreter.

5.2.2. `scriptIsActive`

JMAP Error Code: `scriptIsActive`

Intended use: common

Change controller: IETF

Reference: This document, [Section 2.2](#)

Description: The client tried to destroy the active SieveScript.

6. Acknowledgments

The concepts in this document are based largely on those in [RFC5804]. The author would like to thank the authors of that document for providing both inspiration and some borrowed text for this document.

The author would also like to thank the following individuals for contributing their ideas and support for writing this specification: Bron Gondwana, Neil Jenkins, Alexey Melnikov, and Ricardo Signes.

7. References

7.1. Normative References

- [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>>.
- [RFC3629] Yergeau, F., "UTF-8, a transformation format of ISO 10646", STD 63, RFC 3629, DOI 10.17487/RFC3629, November 2003, <<https://www.rfc-editor.org/info/rfc3629>>.

[RFC3986]

Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, DOI 10.17487/RFC3986, January 2005, <<https://www.rfc-editor.org/info/rfc3986>>.

[RFC5198]

Klensin, J. and M. Padlipsky, "Unicode Format for Network Interchange", RFC 5198, DOI 10.17487/RFC5198, March 2008, <<https://www.rfc-editor.org/info/rfc5198>>.

[RFC5228]

Guenther, P., Ed. and T. Showalter, Ed., "Sieve: An Email Filtering Language", RFC 5228, DOI 10.17487/RFC5228, January 2008, <<https://www.rfc-editor.org/info/rfc5228>>.

[RFC5322]

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

[RFC5435]

Melnikov, A., Ed., Leiba, B., Ed., Segmuller, W., and T. Martin, "Sieve Email Filtering: Extension for Notifications", RFC 5435, DOI 10.17487/RFC5435, January 2009, <<https://www.rfc-editor.org/info/rfc5435>>.

[RFC6134]

Melnikov, A. and B. Leiba, "Sieve Extension: Externally Stored Lists", RFC 6134, DOI 10.17487/RFC6134, July 2011, <<https://www.rfc-editor.org/info/rfc6134>>.

[RFC8174]

Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.

[RFC8620]

Jenkins, N. and C. Newman, "The JSON Meta Application Protocol (JMAP)", RFC 8620, DOI 10.17487/RFC8620, July 2019, <<https://www.rfc-editor.org/info/rfc8620>>.

[RFC8621]

Jenkins, N. and C. Newman, "The JSON Meta Application Protocol (JMAP) for Mail", RFC 8621, DOI 10.17487/RFC8621, August 2019, <<https://www.rfc-editor.org/info/rfc8621>>.

7.2. Informative References

[I-D.ietf-jmap-blob] Gondwana, B., "JMAP Blob management extension",

Work in Progress, Internet-Draft, draft-ietf-jmap-blob-11, 27 March 2022, <<https://datatracker.ietf.org/doc/html/draft-ietf-jmap-blob-11>>.

[RFC5230]

Showalter, T. and N. Freed, Ed., "Sieve Email Filtering: Vacation Extension", RFC 5230, DOI 10.17487/RFC5230, January 2008, <<https://www.rfc-editor.org/info/rfc5230>>.

[RFC5232]

Melnikov, A., "Sieve Email Filtering: Imap4flags Extension", RFC 5232, DOI 10.17487/RFC5232, January 2008, <<https://www.rfc-editor.org/info/rfc5232>>.

[RFC5429]

Stone, A., Ed., "Sieve Email Filtering: Reject and Extended Reject Extensions", RFC 5429, DOI 10.17487/RFC5429, March 2009, <<https://www.rfc-editor.org/info/rfc5429>>.

[RFC5463]

Freed, N., "Sieve Email Filtering: Ihave Extension", RFC 5463, DOI 10.17487/RFC5463, March 2009, <<https://www.rfc-editor.org/info/rfc5463>>.

[RFC5804]

Melnikov, A., Ed. and T. Martin, "A Protocol for Remotely Managing Sieve Scripts", RFC 5804, DOI 10.17487/RFC5804, July 2010, <<https://www.rfc-editor.org/info/rfc5804>>.

Appendix A. Change History (To be removed by RFC Editor before publication)

Changes since ietf-06:

- *None (refreshed to avoid expiration).

Changes since ietf-05:

- *Converted source from xml2rfc v2 to v3.

- *Added examples for SieveScript/get.

- *Miscellaneous editorial changes.

Changes since ietf-04:

- *SieveScript/test: Switched from using a JSON array for each completed action and its args to a JSON object.

- *Switched to referencing draft-ietf-jmap-blob.

- *Miscellaneous editorial changes.

Changes since ietf-03:

- *SieveScript/test: Moved positional arguments into their own array (because the specifications don't use a consistent method for defining the action syntax or naming of positional arguments).

Changes since ietf-02:

- *Removed open issues.
- *Reverted back to using only blob ids for script content.
- *Added "rateLimit" and "requestTooLarge" to the list of possible error codes for /set method.
- *Added Compatibility with JMAP Vacation Response section.
- *Added RFC5228 to Security Considerations.
- *Miscellaneous editorial changes.

Changes since ietf-01:

- *Removed normative references to ManageSieve (RFC 5804).
- *Added the 'maxSizeScriptName' capability.
- *Made the 'name' property in the SieveScript object optional.
- *Added requirements for the 'name' property in the SieveScript object.
- *Removed the 'blobId' property from the SieveScript object.
- *Removed the 'replaceOnCreate' argument from the /set method.
- *Removed the 'blobId' argument from the /validate method.
- *Removed the 'scriptBlobId' argument from, and added the 'scriptContent' argument to, the /test method.
- *Editorial fixes from Neil Jenkins and Ricardo Signes.
- *Other miscellaneous text reorganization and editorial fixes.

Changes since ietf-00:

- *Specified that changes made by onSuccessActivateScript MUST be reported in the /set response as created and/or updated as appropriate.
- *Reworked and specified more of the /test response based on implementation experience.

Changes since murchison-01:

- *Explicitly stated that Sieve capability strings are case-sensitive.

- *errorDescription is now String|null.

- *Added /query method.

- *Added /test method.

Changes since murchison-00:

- *Added IANA registration for "scriptIsActive" JMAP error code.

- *Added open issue about /set{create} with an existing script name.

Author's Address

Kenneth Murchison
Fastmail US LLC
1429 Walnut Street - Suite 1201
Philadelphia, PA 19102
United States of America

Email: murch@fastmailteam.com