JMAP Sieve

draft-ietf-jmap-sieve-02
Ken Murchison
Changes since IETF 108

- accountCapabilities:
  - Sieve capability strings are now case-insensitive.
  - Added a `maxSizeScriptName` capability.
- SieveScript object:
  - The `name` property is now optional in the `/set` method. If `null`, the server assigns a unique name.
  - The `isActive` property is now server-set (via the `onSuccessActivateScript` argument to the `/set` method).
  - Do we need/want any additional properties? E.g, `includedBy`?
Changes since IETF 108

• SieveScript/set:
  ➢ Added an \textit{onSuccessActivateScript} argument – used to atomically set/change/unset the active script.
  ➢ Any changes made to SieveScript object(s) by \textit{onSuccessActivateScript} MUST be reported in the /set response as created and/or updated as appropriate.
  ➢ Can be used with or without any create, update, or destroy arguments.
Changes since IETF 108

• SieveScript/query:
  ➢ Can filter and sort on the name and/or the isActive properties.
  ➢ Do we need/want any additional filter or sort criteria? E.g., isIncluded, inUse?
Changes since IETF 108

• SieveScript/test Request:
  - `accountId` (Id): The id of the account to use.
  - `scriptContent` (String): Raw octets of the script to test.
  - `scriptId` (Id): The id of an existing SieveScript to test.
  - `emailBlobIds` (Id[]): The ids representing the raw octets of the RFC 5322 messages to test against.
  - `envelope` (Envelope|null): Information to assume was present in the SMTP transaction that delivered the message.
  - `lastVacationResponse` (UTCDate|null): The date-time at which the interpreter should assume that it last auto-replied to the sender of the message.
  - Do we need any additional arguments? E.g., for “environment” or “duplicate”?
Changes since IETF 108

- SieveScript/test Reply:
  - **accountId (Id)**: The id of the account used for the call.
  - **completed (Id[Action[]]|null)**: A map of the blob id to a set of `Action` types for each message that was successfully processed by the script. The `Action` type is a tuple, represented as a JSON array containing two elements:
    1) A String *name* of the Sieve action.
    2) A String[*] object containing named *arguments* for that 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.
Example Action Type

[
  "vacation",
  {
    "fcc": "INBOX.Sent",
    "flags": [ "\answered" ],
    "subject": "Auto: test email",
    "from": "ken@example.com",
    "reason": "Gone fishing."
  }
]
Possible Additions/Changes

• Should “:fcc” and its associated arguments reported in the /test response be in their own “fcc” sub-object?
• Do we need any (rate) limits for /test?
• ???