



# OpenShift Container Platform 4.20

## Image APIs

Reference guide for image APIs



# OpenShift Container Platform 4.20 Image APIs

---

Reference guide for image APIs

## Legal Notice

Copyright © 2025 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux<sup>®</sup> is the registered trademark of Linus Torvalds in the United States and other countries.

Java<sup>®</sup> is a registered trademark of Oracle and/or its affiliates.

XFS<sup>®</sup> is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL<sup>®</sup> is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js<sup>®</sup> is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack<sup>®</sup> Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

## Abstract

This document describes the OpenShift Container Platform image API objects and their detailed specifications.

## Table of Contents

|                                                           |           |
|-----------------------------------------------------------|-----------|
| <b>CHAPTER 1. IMAGE APIS</b>                              | <b>7</b>  |
| 1.1. IMAGE [IMAGE.OPENSIFT.IO/V1]                         | 7         |
| 1.2. IMAGESIGNATURE [IMAGE.OPENSIFT.IO/V1]                | 7         |
| 1.3. IMAGESTREAMIMAGE [IMAGE.OPENSIFT.IO/V1]              | 7         |
| 1.4. IMAGESTREAMIMPORT [IMAGE.OPENSIFT.IO/V1]             | 8         |
| 1.5. IMAGESTREAMLAYERS [IMAGE.OPENSIFT.IO/V1]             | 8         |
| 1.6. IMAGESTREAMMAPPING [IMAGE.OPENSIFT.IO/V1]            | 8         |
| 1.7. IMAGESTREAM [IMAGE.OPENSIFT.IO/V1]                   | 8         |
| 1.8. IMAGESTREAMTAG [IMAGE.OPENSIFT.IO/V1]                | 9         |
| 1.9. IMAGETAG [IMAGE.OPENSIFT.IO/V1]                      | 9         |
| 1.10. SECRETLIST [IMAGE.OPENSIFT.IO/V1]                   | 9         |
| <b>CHAPTER 2. IMAGE [IMAGE.OPENSIFT.IO/V1]</b>            | <b>11</b> |
| 2.1. SPECIFICATION                                        | 11        |
| 2.1.1. .dockerImageLayers                                 | 13        |
| 2.1.2. .dockerImageLayers[]                               | 13        |
| 2.1.3. .dockerImageManifests                              | 14        |
| 2.1.4. .dockerImageManifests[]                            | 14        |
| 2.1.5. .signatures                                        | 15        |
| 2.1.6. .signatures[]                                      | 15        |
| 2.1.7. .signatures[].conditions                           | 17        |
| 2.1.8. .signatures[].conditions[]                         | 17        |
| 2.1.9. .signatures[].issuedBy                             | 18        |
| 2.1.10. .signatures[].issuedTo                            | 18        |
| 2.2. API ENDPOINTS                                        | 19        |
| 2.2.1. /apis/image.openshift.io/v1/images                 | 19        |
| 2.2.2. /apis/image.openshift.io/v1/watch/images           | 21        |
| 2.2.3. /apis/image.openshift.io/v1/images/{name}          | 22        |
| 2.2.4. /apis/image.openshift.io/v1/watch/images/{name}    | 25        |
| <b>CHAPTER 3. IMAGESIGNATURE [IMAGE.OPENSIFT.IO/V1]</b>   | <b>26</b> |
| 3.1. SPECIFICATION                                        | 26        |
| 3.1.1. .conditions                                        | 27        |
| 3.1.2. .conditions[]                                      | 28        |
| 3.1.3. .issuedBy                                          | 28        |
| 3.1.4. .issuedTo                                          | 29        |
| 3.2. API ENDPOINTS                                        | 29        |
| 3.2.1. /apis/image.openshift.io/v1/imagesignatures        | 29        |
| 3.2.2. /apis/image.openshift.io/v1/imagesignatures/{name} | 31        |
| <b>CHAPTER 4. IMAGESTREAMIMAGE [IMAGE.OPENSIFT.IO/V1]</b> | <b>32</b> |
| 4.1. SPECIFICATION                                        | 32        |
| 4.1.1. .image                                             | 33        |
| 4.1.2. .image.dockerImageLayers                           | 36        |
| 4.1.3. .image.dockerImageLayers[]                         | 36        |
| 4.1.4. .image.dockerImageManifests                        | 37        |
| 4.1.5. .image.dockerImageManifests[]                      | 37        |
| 4.1.6. .image.signatures                                  | 38        |
| 4.1.7. .image.signatures[]                                | 38        |
| 4.1.8. .image.signatures[].conditions                     | 40        |
| 4.1.9. .image.signatures[].conditions[]                   | 40        |
| 4.1.10. .image.signatures[].issuedBy                      | 41        |

|                                                                                    |           |
|------------------------------------------------------------------------------------|-----------|
| 4.1.11. .image.signatures[].issuedTo                                               | 41        |
| 4.2. API ENDPOINTS                                                                 | 42        |
| 4.2.1. /apis/image.openshift.io/v1/namespaces/{namespace}/imagestreamimages/{name} | 42        |
| <b>CHAPTER 5. IMAGESTREAMIMPORT [IMAGE.OPENSIFT.IO/V1]</b>                         | <b>43</b> |
| 5.1. SPECIFICATION                                                                 | 43        |
| 5.1.1. .spec                                                                       | 44        |
| 5.1.2. .spec.images                                                                | 44        |
| 5.1.3. .spec.images[]                                                              | 45        |
| 5.1.4. .spec.images[].importPolicy                                                 | 45        |
| 5.1.5. .spec.images[].referencePolicy                                              | 46        |
| 5.1.6. .spec.repository                                                            | 47        |
| 5.1.7. .spec.repository.importPolicy                                               | 48        |
| 5.1.8. .spec.repository.referencePolicy                                            | 48        |
| 5.1.9. .status                                                                     | 49        |
| 5.1.10. .status.images                                                             | 50        |
| 5.1.11. .status.images[]                                                           | 51        |
| 5.1.12. .status.images[].image                                                     | 52        |
| 5.1.13. .status.images[].image.dockerImageLayers                                   | 55        |
| 5.1.14. .status.images[].image.dockerImageLayers[]                                 | 55        |
| 5.1.15. .status.images[].image.dockerImageManifests                                | 56        |
| 5.1.16. .status.images[].image.dockerImageManifests[]                              | 56        |
| 5.1.17. .status.images[].image.signatures                                          | 57        |
| 5.1.18. .status.images[].image.signatures[]                                        | 57        |
| 5.1.19. .status.images[].image.signatures[].conditions                             | 59        |
| 5.1.20. .status.images[].image.signatures[].conditions[]                           | 59        |
| 5.1.21. .status.images[].image.signatures[].issuedBy                               | 60        |
| 5.1.22. .status.images[].image.signatures[].issuedTo                               | 60        |
| 5.1.23. .status.images[].manifests                                                 | 61        |
| 5.1.24. .status.images[].manifests[]                                               | 61        |
| 5.1.25. .status.images[].manifests[].dockerImageLayers                             | 64        |
| 5.1.26. .status.images[].manifests[].dockerImageLayers[]                           | 64        |
| 5.1.27. .status.images[].manifests[].dockerImageManifests                          | 65        |
| 5.1.28. .status.images[].manifests[].dockerImageManifests[]                        | 65        |
| 5.1.29. .status.images[].manifests[].signatures                                    | 66        |
| 5.1.30. .status.images[].manifests[].signatures[]                                  | 66        |
| 5.1.31. .status.images[].manifests[].signatures[].conditions                       | 68        |
| 5.1.32. .status.images[].manifests[].signatures[].conditions[]                     | 68        |
| 5.1.33. .status.images[].manifests[].signatures[].issuedBy                         | 69        |
| 5.1.34. .status.images[].manifests[].signatures[].issuedTo                         | 69        |
| 5.1.35. .status.import                                                             | 70        |
| 5.1.36. .status.import.spec                                                        | 71        |
| 5.1.37. .status.import.spec.lookupPolicy                                           | 72        |
| 5.1.38. .status.import.spec.tags                                                   | 72        |
| 5.1.39. .status.import.spec.tags[]                                                 | 73        |
| 5.1.40. .status.import.spec.tags[].importPolicy                                    | 74        |
| 5.1.41. .status.import.spec.tags[].referencePolicy                                 | 75        |
| 5.1.42. .status.import.status                                                      | 76        |
| 5.1.43. .status.import.status.tags                                                 | 77        |
| 5.1.44. .status.import.status.tags[]                                               | 77        |
| 5.1.45. .status.import.status.tags[].conditions                                    | 78        |
| 5.1.46. .status.import.status.tags[].conditions[]                                  | 78        |
| 5.1.47. .status.import.status.tags[].items                                         | 79        |

|                                                                                      |            |
|--------------------------------------------------------------------------------------|------------|
| 5.1.48. .status.import.status.tags[].items[]                                         | 79         |
| 5.1.49. .status.repository                                                           | 80         |
| 5.1.50. .status.repository.images                                                    | 80         |
| 5.1.51. .status.repository.images[]                                                  | 81         |
| 5.1.52. .status.repository.images[].image                                            | 82         |
| 5.1.53. .status.repository.images[].image.dockerImageLayers                          | 85         |
| 5.1.54. .status.repository.images[].image.dockerImageLayers[]                        | 85         |
| 5.1.55. .status.repository.images[].image.dockerImageManifests                       | 86         |
| 5.1.56. .status.repository.images[].image.dockerImageManifests[]                     | 86         |
| 5.1.57. .status.repository.images[].image.signatures                                 | 87         |
| 5.1.58. .status.repository.images[].image.signatures[]                               | 87         |
| 5.1.59. .status.repository.images[].image.signatures[].conditions                    | 89         |
| 5.1.60. .status.repository.images[].image.signatures[].conditions[]                  | 89         |
| 5.1.61. .status.repository.images[].image.signatures[].issuedBy                      | 90         |
| 5.1.62. .status.repository.images[].image.signatures[].issuedTo                      | 90         |
| 5.1.63. .status.repository.images[].manifests                                        | 91         |
| 5.1.64. .status.repository.images[].manifests[]                                      | 91         |
| 5.1.65. .status.repository.images[].manifests[].dockerImageLayers                    | 94         |
| 5.1.66. .status.repository.images[].manifests[].dockerImageLayers[]                  | 94         |
| 5.1.67. .status.repository.images[].manifests[].dockerImageManifests                 | 95         |
| 5.1.68. .status.repository.images[].manifests[].dockerImageManifests[]               | 95         |
| 5.1.69. .status.repository.images[].manifests[].signatures                           | 96         |
| 5.1.70. .status.repository.images[].manifests[].signatures[]                         | 96         |
| 5.1.71. .status.repository.images[].manifests[].signatures[].conditions              | 98         |
| 5.1.72. .status.repository.images[].manifests[].signatures[].conditions[]            | 98         |
| 5.1.73. .status.repository.images[].manifests[].signatures[].issuedBy                | 99         |
| 5.1.74. .status.repository.images[].manifests[].signatures[].issuedTo                | 99         |
| 5.2. API ENDPOINTS                                                                   | 100        |
| 5.2.1. /apis/image.openshift.io/v1/namespaces/{namespace}/imagestreamimports         | 100        |
| <b>CHAPTER 6. IMAGESTREAMLAYERS [IMAGE.OPENSIFT.IO/V1]</b>                           | <b>102</b> |
| 6.1. SPECIFICATION                                                                   | 102        |
| 6.1.1. .blobs                                                                        | 103        |
| 6.1.2. .blobs{}                                                                      | 103        |
| 6.1.3. .images                                                                       | 104        |
| 6.1.4. .images{}                                                                     | 104        |
| 6.2. API ENDPOINTS                                                                   | 105        |
| 6.2.1. /apis/image.openshift.io/v1/namespaces/{namespace}/imagestreams/{name}/layers | 105        |
| <b>CHAPTER 7. IMAGESTREAMMAPPING [IMAGE.OPENSIFT.IO/V1]</b>                          | <b>106</b> |
| 7.1. SPECIFICATION                                                                   | 106        |
| 7.1.1. .image                                                                        | 108        |
| 7.1.2. .image.dockerImageLayers                                                      | 110        |
| 7.1.3. .image.dockerImageLayers[]                                                    | 110        |
| 7.1.4. .image.dockerImageManifests                                                   | 111        |
| 7.1.5. .image.dockerImageManifests[]                                                 | 111        |
| 7.1.6. .image.signatures                                                             | 112        |
| 7.1.7. .image.signatures[]                                                           | 112        |
| 7.1.8. .image.signatures[].conditions                                                | 114        |
| 7.1.9. .image.signatures[].conditions[]                                              | 114        |
| 7.1.10. .image.signatures[].issuedBy                                                 | 115        |
| 7.1.11. .image.signatures[].issuedTo                                                 | 115        |
| 7.2. API ENDPOINTS                                                                   | 116        |

|                                                                                      |            |
|--------------------------------------------------------------------------------------|------------|
| 7.2.1. /apis/image.openshift.io/v1/namespaces/{namespace}/imagestreammappings        | 116        |
| <b>CHAPTER 8. IMAGESTREAM [IMAGE.OPENSIFT.IO/V1]</b>                                 | <b>118</b> |
| 8.1. SPECIFICATION                                                                   | 118        |
| 8.1.1. .spec                                                                         | 119        |
| 8.1.2. .spec.lookupPolicy                                                            | 120        |
| 8.1.3. .spec.tags                                                                    | 120        |
| 8.1.4. .spec.tags[]                                                                  | 120        |
| 8.1.5. .spec.tags[].importPolicy                                                     | 122        |
| 8.1.6. .spec.tags[].referencePolicy                                                  | 122        |
| 8.1.7. .status                                                                       | 123        |
| 8.1.8. .status.tags                                                                  | 124        |
| 8.1.9. .status.tags[]                                                                | 124        |
| 8.1.10. .status.tags[].conditions                                                    | 125        |
| 8.1.11. .status.tags[].conditions[]                                                  | 125        |
| 8.1.12. .status.tags[].items                                                         | 126        |
| 8.1.13. .status.tags[].items[]                                                       | 126        |
| 8.2. API ENDPOINTS                                                                   | 127        |
| 8.2.1. /apis/image.openshift.io/v1/imagestreams                                      | 128        |
| 8.2.2. /apis/image.openshift.io/v1/watch/imagestreams                                | 128        |
| 8.2.3. /apis/image.openshift.io/v1/namespaces/{namespace}/imagestreams               | 128        |
| 8.2.4. /apis/image.openshift.io/v1/watch/namespaces/{namespace}/imagestreams         | 130        |
| 8.2.5. /apis/image.openshift.io/v1/namespaces/{namespace}/imagestreams/{name}        | 131        |
| 8.2.6. /apis/image.openshift.io/v1/watch/namespaces/{namespace}/imagestreams/{name}  | 134        |
| 8.2.7. /apis/image.openshift.io/v1/namespaces/{namespace}/imagestreams/{name}/status | 134        |
| <b>CHAPTER 9. IMAGESTREAMTAG [IMAGE.OPENSIFT.IO/V1]</b>                              | <b>138</b> |
| 9.1. SPECIFICATION                                                                   | 138        |
| 9.1.1. .conditions                                                                   | 140        |
| 9.1.2. .conditions[]                                                                 | 140        |
| 9.1.3. .image                                                                        | 141        |
| 9.1.4. .image.dockerImageLayers                                                      | 144        |
| 9.1.5. .image.dockerImageLayers[]                                                    | 144        |
| 9.1.6. .image.dockerImageManifests                                                   | 145        |
| 9.1.7. .image.dockerImageManifests[]                                                 | 145        |
| 9.1.8. .image.signatures                                                             | 146        |
| 9.1.9. .image.signatures[]                                                           | 146        |
| 9.1.10. .image.signatures[].conditions                                               | 148        |
| 9.1.11. .image.signatures[].conditions[]                                             | 148        |
| 9.1.12. .image.signatures[].issuedBy                                                 | 149        |
| 9.1.13. .image.signatures[].issuedTo                                                 | 149        |
| 9.1.14. .lookupPolicy                                                                | 150        |
| 9.1.15. .tag                                                                         | 150        |
| 9.1.16. .tag.importPolicy                                                            | 152        |
| 9.1.17. .tag.referencePolicy                                                         | 152        |
| 9.2. API ENDPOINTS                                                                   | 153        |
| 9.2.1. /apis/image.openshift.io/v1/imagestreamtags                                   | 154        |
| 9.2.2. /apis/image.openshift.io/v1/namespaces/{namespace}/imagestreamtags            | 154        |
| 9.2.3. /apis/image.openshift.io/v1/namespaces/{namespace}/imagestreamtags/{name}     | 155        |
| <b>CHAPTER 10. IMAGETAG [IMAGE.OPENSIFT.IO/V1]</b>                                   | <b>159</b> |
| 10.1. SPECIFICATION                                                                  | 159        |
| 10.1.1. .image                                                                       | 161        |
| 10.1.2. .image.dockerImageLayers                                                     | 163        |

|                                                                                        |            |
|----------------------------------------------------------------------------------------|------------|
| 10.1.3. .image.dockerImageLayers[]                                                     | 164        |
| 10.1.4. .image.dockerImageManifests                                                    | 164        |
| 10.1.5. .image.dockerImageManifests[]                                                  | 164        |
| 10.1.6. .image.signatures                                                              | 165        |
| 10.1.7. .image.signatures[]                                                            | 165        |
| 10.1.8. .image.signatures[].conditions                                                 | 167        |
| 10.1.9. .image.signatures[].conditions[]                                               | 167        |
| 10.1.10. .image.signatures[].issuedBy                                                  | 168        |
| 10.1.11. .image.signatures[].issuedTo                                                  | 168        |
| 10.1.12. .spec                                                                         | 169        |
| 10.1.13. .spec.importPolicy                                                            | 170        |
| 10.1.14. .spec.referencePolicy                                                         | 171        |
| 10.1.15. .status                                                                       | 172        |
| 10.1.16. .status.conditions                                                            | 173        |
| 10.1.17. .status.conditions[]                                                          | 173        |
| 10.1.18. .status.items                                                                 | 174        |
| 10.1.19. .status.items[]                                                               | 174        |
| 10.2. API ENDPOINTS                                                                    | 175        |
| 10.2.1. /apis/image.openshift.io/v1/imagetags                                          | 175        |
| 10.2.2. /apis/image.openshift.io/v1/namespaces/{namespace}/imagetags                   | 176        |
| 10.2.3. /apis/image.openshift.io/v1/namespaces/{namespace}/imagetags/{name}            | 177        |
| <b>CHAPTER 11. SECRETLIST [IMAGE.OPENSIFT.IO/V1]</b>                                   | <b>181</b> |
| 11.1. SPECIFICATION                                                                    | 181        |
| 11.2. API ENDPOINTS                                                                    | 181        |
| 11.2.1. /apis/image.openshift.io/v1/namespaces/{namespace}/imagestreams/{name}/secrets | 182        |



## CHAPTER 1. IMAGE APIS

### 1.1. IMAGE [IMAGE.OPENSIFT.IO/V1]

#### Description

Image is an immutable representation of a container image and metadata at a point in time. Images are named by taking a hash of their contents (metadata and content) and any change in format, content, or metadata results in a new name. The images resource is primarily for use by cluster administrators and integrations like the cluster image registry - end users instead access images via the `imagestreamtags` or `imagestreamimages` resources. While image metadata is stored in the API, any integration that implements the container image registry API must provide its own storage for the raw manifest data, image config, and layer contents.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

#### Type

**object**

### 1.2. IMAGESIGNATURE [IMAGE.OPENSIFT.IO/V1]

#### Description

ImageSignature holds a signature of an image. It allows to verify image identity and possibly other claims as long as the signature is trusted. Based on this information it is possible to restrict runnable images to those matching cluster-wide policy. Mandatory fields should be parsed by clients doing image verification. The others are parsed from signature's content by the server. They serve just an informative purpose.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

#### Type

**object**

### 1.3. IMAGESTREAMIMAGE [IMAGE.OPENSIFT.IO/V1]

#### Description

ImageStreamImage represents an Image that is retrieved by image name from an ImageStream. User interfaces and regular users can use this resource to access the metadata details of a tagged image in the image stream history for viewing, since Image resources are not directly accessible to end users. A not found error will be returned if no such image is referenced by a tag within the ImageStream. Images are created when spec tags are set on an image stream that represent an image in an external registry, when pushing to the integrated registry, or when tagging an existing image from one image stream to another. The name of an image stream image is in the form "`<STREAM>@<DIGEST>`", where the digest is the content addressable identifier for the image (`sha256:xxxxx...`). You can use `ImageStreamImages` as the `from.kind` of an image stream spec tag to reference an image exactly. The only operations supported on the `imagestreamimage` endpoint are retrieving the image.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

#### Type

**object**

## 1.4. IMAGESTREAMIMPORT [IMAGE.OPENSIFT.IO/V1]

### Description

The image stream import resource provides an easy way for a user to find and import container images from other container image registries into the server. Individual images or an entire image repository may be imported, and users may choose to see the results of the import prior to tagging the resulting images into the specified image stream.

This API is intended for end-user tools that need to see the metadata of the image prior to import (for instance, to generate an application from it). Clients that know the desired image can continue to create spec.tags directly into their image streams.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

### Type

**object**

## 1.5. IMAGESTREAMLAYERS [IMAGE.OPENSIFT.IO/V1]

### Description

ImageStreamLayers describes information about the layers referenced by images in this image stream.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

### Type

**object**

## 1.6. IMAGESTREAMMAPPING [IMAGE.OPENSIFT.IO/V1]

### Description

ImageStreamMapping represents a mapping from a single image stream tag to a container image as well as the reference to the container image stream the image came from. This resource is used by privileged integrators to create an image resource and to associate it with an image stream in the status tags field. Creating an ImageStreamMapping will allow any user who can view the image stream to tag or pull that image, so only create mappings where the user has proven they have access to the image contents directly. The only operation supported for this resource is create and the metadata name and namespace should be set to the image stream containing the tag that should be updated.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

### Type

**object**

## 1.7. IMAGESTREAM [IMAGE.OPENSIFT.IO/V1]

### Description

An ImageStream stores a mapping of tags to images, metadata overrides that are applied when images are tagged in a stream, and an optional reference to a container image repository on a registry. Users typically update the `spec.tags` field to point to external images which are imported from container registries using credentials in your namespace with the pull secret type, or to existing image stream tags and images which are immediately accessible for tagging or pulling. The history of images applied to a tag is visible in the `status.tags` field and any user who can view an image stream is allowed to tag that image into their own image streams. Access to pull images from the integrated registry is granted by having the "get imagestreams/layers" permission on a given image stream. Users may remove a tag by deleting the `imagestreamtag` resource, which causes both `spec` and `status` for that tag to be removed. Image stream history is retained until an administrator runs the prune operation, which removes references that are no longer in use. To preserve a historical image, ensure there is a tag in `spec` pointing to that image by its digest.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

## Type

**object**

## 1.8. IMAGESTREAMTAG [IMAGE.OPENSIFT.IO/V1]

### Description

ImageStreamTag represents an Image that is retrieved by tag name from an ImageStream. Use this resource to interact with the tags and images in an image stream by tag, or to see the image details for a particular tag. The image associated with this resource is the most recently successfully tagged, imported, or pushed image (as described in the image stream `status.tags.items` list for this tag). If an import is in progress or has failed the previous image will be shown. Deleting an image stream tag clears both the `status` and `spec` fields of an image stream. If no image can be retrieved for a given tag, a not found error will be returned.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

## Type

**object**

## 1.9. IMAGETAG [IMAGE.OPENSIFT.IO/V1]

### Description

ImageTag represents a single tag within an image stream and includes the `spec`, the status history, and the currently referenced image (if any) of the provided tag. This type replaces the `ImageStreamTag` by providing a full view of the tag. ImageTags are returned for every `spec` or `status` tag present on the image stream. If no tag exists in either form a not found error will be returned by the API. A create operation will succeed if no `spec` tag has already been defined and the `spec` field is set. Delete will remove both `spec` and `status` elements from the image stream.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

## Type

**object**

## 1.10. SECRETLIST [IMAGE.OPENSIFT.IO/V1]

### Description

SecretList is a list of Secret.

### Type

**object**

## CHAPTER 2. IMAGE [IMAGE.OPENSIFT.IO/V1]

### Description

Image is an immutable representation of a container image and metadata at a point in time. Images are named by taking a hash of their contents (metadata and content) and any change in format, content, or metadata results in a new name. The images resource is primarily for use by cluster administrators and integrations like the cluster image registry – end users instead access images via the `imagestreamtags` or `imagestreamimages` resources. While image metadata is stored in the API, any integration that implements the container image registry API must provide its own storage for the raw manifest data, image config, and layer contents.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

### Type

**object**

## 2.1. SPECIFICATION

| Property                   | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                   |
|----------------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>apiVersion</b>          | <b>string</b> | APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources</a> |
| <b>dockerImageConfig</b>   | <b>string</b> | DockerImageConfig is a JSON blob that the runtime uses to set up the container. This is a part of manifest schema v2. Will not be set when the image represents a manifest list.                                                                                                                                                                                                                              |
| <b>dockerImageLayers</b>   | <b>array</b>  | DockerImageLayers represents the layers in the image. May not be set if the image does not define that data or if the image represents a manifest list.                                                                                                                                                                                                                                                       |
| <b>dockerImageLayers[]</b> | <b>object</b> | ImageLayer represents a single layer of the image. Some images may have multiple layers. Some may have none.                                                                                                                                                                                                                                                                                                  |
| <b>dockerImageManifest</b> | <b>string</b> | DockerImageManifest is the raw JSON of the manifest                                                                                                                                                                                                                                                                                                                                                           |

| Property                            | Type                  | Description                                                                                                                                                                                                                                                                                                                                                                                                      |
|-------------------------------------|-----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dockerImageManifestMediaType</b> | <b>string</b>         | DockerImageManifestMediaType specifies the mediaType of manifest. This is a part of manifest schema v2.                                                                                                                                                                                                                                                                                                          |
| <b>dockerImageManifests</b>         | <b>array</b>          | DockerImageManifests holds information about sub-manifests when the image represents a manifest list. When this field is present, no DockerImageLayers should be specified.                                                                                                                                                                                                                                      |
| <b>dockerImageManifests[]</b>       | <b>object</b>         | ImageManifest represents sub-manifests of a manifest list. The Digest field points to a regular Image object.                                                                                                                                                                                                                                                                                                    |
| <b>dockerImageMetadata</b>          | <b>RawExtension</b>   | DockerImageMetadata contains metadata about this image                                                                                                                                                                                                                                                                                                                                                           |
| <b>dockerImageMetadataVersion</b>   | <b>string</b>         | DockerImageMetadataVersion conveys the version of the object, which if empty defaults to "1.0"                                                                                                                                                                                                                                                                                                                   |
| <b>dockerImageReference</b>         | <b>string</b>         | DockerImageReference is the string that can be used to pull this image.                                                                                                                                                                                                                                                                                                                                          |
| <b>dockerImageSignatures</b>        | <b>array (string)</b> | DockerImageSignatures provides the signatures as opaque blobs. This is a part of manifest schema v1.                                                                                                                                                                                                                                                                                                             |
| <b>kind</b>                         | <b>string</b>         | Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds</a> |

| Property            | Type                 | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|---------------------|----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>metadata</b>     | <b>ObjectMeta_v2</b> | metadata is the standard object's metadata. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata</a>                                                                                                                                                                                                                                                                                                                    |
| <b>signatures</b>   | <b>array</b>         | Signatures holds all signatures of the image.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| <b>signatures[]</b> | <b>object</b>        | <p>ImageSignature holds a signature of an image. It allows to verify image identity and possibly other claims as long as the signature is trusted. Based on this information it is possible to restrict runnable images to those matching cluster-wide policy. Mandatory fields should be parsed by clients doing image verification. The others are parsed from signature's content by the server. They serve just an informative purpose.</p> <p>Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).</p> |

### 2.1.1. .dockerImageLayers

#### Description

DockerImageLayers represents the layers in the image. May not be set if the image does not define that data or if the image represents a manifest list.

#### Type

**array**

### 2.1.2. .dockerImageLayers[]

#### Description

ImageLayer represents a single layer of the image. Some images may have multiple layers. Some may have none.

#### Type

**object**

#### Required

- **name**

- **size**
- **mediaType**

| Property         | Type           | Description                                                    |
|------------------|----------------|----------------------------------------------------------------|
| <b>mediaType</b> | <b>string</b>  | MediaType of the referenced object.                            |
| <b>name</b>      | <b>string</b>  | Name of the layer as defined by the underlying store.          |
| <b>size</b>      | <b>integer</b> | Size of the layer in bytes as defined by the underlying store. |

### 2.1.3. .dockerImageManifests

#### Description

DockerImageManifests holds information about sub-manifests when the image represents a manifest list. When this field is present, no DockerImageLayers should be specified.

#### Type

**array**

### 2.1.4. .dockerImageManifests[]

#### Description

ImageManifest represents sub-manifests of a manifest list. The Digest field points to a regular Image object.

#### Type

**object**

#### Required

- **digest**
- **mediaType**
- **manifestSize**
- **architecture**
- **os**

| Property            | Type          | Description                                                                                         |
|---------------------|---------------|-----------------------------------------------------------------------------------------------------|
| <b>architecture</b> | <b>string</b> | Architecture specifies the supported CPU architecture, for example <b>amd64</b> or <b>ppc64le</b> . |

| Property            | Type           | Description                                                                                                                                                                                                               |
|---------------------|----------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>digest</b>       | <b>string</b>  | Digest is the unique identifier for the manifest. It refers to an Image object.                                                                                                                                           |
| <b>manifestSize</b> | <b>integer</b> | ManifestSize represents the size of the raw object contents, in bytes.                                                                                                                                                    |
| <b>mediaType</b>    | <b>string</b>  | MediaType defines the type of the manifest, possible values are application/vnd.oci.image.manifest.v1+json, application/vnd.docker.distribution.manifest.v2+json or application/vnd.docker.distribution.manifest.v1+json. |
| <b>os</b>           | <b>string</b>  | OS specifies the operating system, for example <b>linux</b> .                                                                                                                                                             |
| <b>variant</b>      | <b>string</b>  | Variant is an optional field representing a variant of the CPU, for example v6 to specify a particular CPU variant of the ARM CPU.                                                                                        |

### 2.1.5. .signatures

#### Description

Signatures holds all signatures of the image.

#### Type

**array**

### 2.1.6. .signatures[]

#### Description

ImageSignature holds a signature of an image. It allows to verify image identity and possibly other claims as long as the signature is trusted. Based on this information it is possible to restrict runnable images to those matching cluster-wide policy. Mandatory fields should be parsed by clients doing image verification. The others are parsed from signature's content by the server. They serve just an informative purpose.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

#### Type

**object**

#### Required

- **type**
- **content**

| Property             | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                   |
|----------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>apiVersion</b>    | <b>string</b> | APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources</a> |
| <b>conditions</b>    | <b>array</b>  | Conditions represent the latest available observations of a signature's current state.                                                                                                                                                                                                                                                                                                                        |
| <b>conditions[]</b>  | <b>object</b> | SignatureCondition describes an image signature condition of particular kind at particular probe time.                                                                                                                                                                                                                                                                                                        |
| <b>content</b>       | <b>string</b> | Required: An opaque binary string which is an image's signature.                                                                                                                                                                                                                                                                                                                                              |
| <b>created</b>       | <b>Time</b>   | If specified, it is the time of signature's creation.                                                                                                                                                                                                                                                                                                                                                         |
| <b>imageIdentity</b> | <b>string</b> | A human readable string representing image's identity. It could be a product name and version, or an image pull spec (e.g. "registry.access.redhat.com/rhel7/rhel:7.2").                                                                                                                                                                                                                                      |
| <b>issuedBy</b>      | <b>object</b> | SignatureIssuer holds information about an issuer of signing certificate or key.                                                                                                                                                                                                                                                                                                                              |
| <b>issuedTo</b>      | <b>object</b> | SignatureSubject holds information about a person or entity who created the signature.                                                                                                                                                                                                                                                                                                                        |

| Property            | Type                   | Description                                                                                                                                                                                                                                                                                                                                                                                                      |
|---------------------|------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>kind</b>         | <b>string</b>          | Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds</a> |
| <b>metadata</b>     | <b>ObjectMeta_v2</b>   | metadata is the standard object's metadata. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata</a>                                                                                                                                                   |
| <b>signedClaims</b> | <b>object (string)</b> | Contains claims from the signature.                                                                                                                                                                                                                                                                                                                                                                              |
| <b>type</b>         | <b>string</b>          | Required: Describes a type of stored blob.                                                                                                                                                                                                                                                                                                                                                                       |

### 2.1.7. .signatures[].conditions

#### Description

Conditions represent the latest available observations of a signature's current state.

#### Type

**array**

### 2.1.8. .signatures[].conditions[]

#### Description

SignatureCondition describes an image signature condition of particular kind at particular probe time.

#### Type

**object**

#### Required

- **type**
- **status**

| Property                  | Type          | Description                                                      |
|---------------------------|---------------|------------------------------------------------------------------|
| <b>lastProbeTime</b>      | <b>Time</b>   | Last time the condition was checked.                             |
| <b>lastTransitionTime</b> | <b>Time</b>   | Last time the condition transit from one status to another.      |
| <b>message</b>            | <b>string</b> | Human readable message indicating details about last transition. |
| <b>reason</b>             | <b>string</b> | (brief) reason for the condition's last transition.              |
| <b>status</b>             | <b>string</b> | Status of the condition, one of True, False, Unknown.            |
| <b>type</b>               | <b>string</b> | Type of signature condition, Complete or Failed.                 |

### 2.1.9. .signatures[].issuedBy

#### Description

SignatureIssuer holds information about an issuer of signing certificate or key.

#### Type

**object**

| Property            | Type          | Description                                   |
|---------------------|---------------|-----------------------------------------------|
| <b>commonName</b>   | <b>string</b> | Common name (e.g. openshift-signing-service). |
| <b>organization</b> | <b>string</b> | Organization name.                            |

### 2.1.10. .signatures[].issuedTo

#### Description

SignatureSubject holds information about a person or entity who created the signature.

#### Type

**object**

#### Required

- **publicKeyID**

| Property            | Type          | Description                                                                                                                                                                                                       |
|---------------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>commonName</b>   | <b>string</b> | Common name (e.g. openshift-signing-service).                                                                                                                                                                     |
| <b>organization</b> | <b>string</b> | Organization name.                                                                                                                                                                                                |
| <b>publicKeyID</b>  | <b>string</b> | If present, it is a human readable key id of public key belonging to the subject used to verify image signature. It should contain at least 64 lowest bits of public key's fingerprint (e.g. 0x685ebe62bf278440). |

## 2.2. API ENDPOINTS

The following API endpoints are available:

- **/apis/image.openshift.io/v1/images**
  - **DELETE**: delete collection of Image
  - **GET**: list or watch objects of kind Image
  - **POST**: create an Image
- **/apis/image.openshift.io/v1/watch/images**
  - **GET**: watch individual changes to a list of Image. deprecated: use the 'watch' parameter with a list operation instead.
- **/apis/image.openshift.io/v1/images/{name}**
  - **DELETE**: delete an Image
  - **GET**: read the specified Image
  - **PATCH**: partially update the specified Image
  - **PUT**: replace the specified Image
- **/apis/image.openshift.io/v1/watch/images/{name}**
  - **GET**: watch changes to an object of kind Image. deprecated: use the 'watch' parameter with a list operation instead, filtered to a single item with the 'fieldSelector' parameter.

### 2.2.1. /apis/image.openshift.io/v1/images

HTTP method

**DELETE**

Description

delete collection of Image

**Table 2.1. Query parameters**

| Parameter     | Type          | Description                                                                                                                                                                                                                                              |
|---------------|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dryRun</b> | <b>string</b> | When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed |

**Table 2.2. HTTP responses**

| HTTP code          | Reponse body                     |
|--------------------|----------------------------------|
| 200 - OK           | <a href="#">Status_v5</a> schema |
| 401 - Unauthorized | Empty                            |

## HTTP method

### GET

## Description

list or watch objects of kind Image

**Table 2.3. HTTP responses**

| HTTP code          | Reponse body                     |
|--------------------|----------------------------------|
| 200 - OK           | <a href="#">ImageList</a> schema |
| 401 - Unauthorized | Empty                            |

## HTTP method

### POST

## Description

create an Image

**Table 2.4. Query parameters**

| Parameter     | Type          | Description                                                                                                                                                                                                                                              |
|---------------|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dryRun</b> | <b>string</b> | When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed |

| Parameter              | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|------------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>fieldValidation</b> | <b>string</b> | fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: <ul style="list-style-type: none"> <li>- Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23.</li> <li>- Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+</li> <li>- Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.</li> </ul> |

Table 2.5. Body parameters

| Parameter   | Type                | Description |
|-------------|---------------------|-------------|
| <b>body</b> | <b>Image</b> schema |             |

Table 2.6. HTTP responses

| HTTP code          | Response body       |
|--------------------|---------------------|
| 200 - OK           | <b>Image</b> schema |
| 201 - Created      | <b>Image</b> schema |
| 202 - Accepted     | <b>Image</b> schema |
| 401 - Unauthorized | Empty               |

### 2.2.2. /apis/image.openshift.io/v1/watch/images

HTTP method

**GET**

Description

watch individual changes to a list of Image. deprecated: use the 'watch' parameter with a list operation instead.

Table 2.7. HTTP responses

| HTTP code          | Reponse body                      |
|--------------------|-----------------------------------|
| 200 - OK           | <a href="#">WatchEvent</a> schema |
| 401 - Unauthorized | Empty                             |

### 2.2.3. /apis/image.openshift.io/v1/images/{name}

Table 2.8. Global path parameters

| Parameter   | Type          | Description       |
|-------------|---------------|-------------------|
| <b>name</b> | <b>string</b> | name of the Image |

#### HTTP method

#### DELETE

#### Description

delete an Image

Table 2.9. Query parameters

| Parameter     | Type          | Description                                                                                                                                                                                                                                              |
|---------------|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dryRun</b> | <b>string</b> | When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed |

Table 2.10. HTTP responses

| HTTP code          | Reponse body                     |
|--------------------|----------------------------------|
| 200 - OK           | <a href="#">Status_v5</a> schema |
| 202 - Accepted     | <a href="#">Status_v5</a> schema |
| 401 - Unauthorized | Empty                            |

#### HTTP method

#### GET

#### Description

read the specified Image

**Table 2.11. HTTP responses**

| HTTP code          | Response body       |
|--------------------|---------------------|
| 200 - OK           | <b>Image</b> schema |
| 401 - Unauthorized | Empty               |

## HTTP method

### PATCH

## Description

partially update the specified Image

**Table 2.12. Query parameters**

| Parameter              | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|------------------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dryRun</b>          | <b>string</b> | When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <b>fieldValidation</b> | <b>string</b> | fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered. |

**Table 2.13. HTTP responses**

| HTTP code | Response body       |
|-----------|---------------------|
| 200 - OK  | <b>Image</b> schema |

| HTTP code          | Response body                |
|--------------------|------------------------------|
| 201 - Created      | <a href="#">Image</a> schema |
| 401 - Unauthorized | Empty                        |

**HTTP method****PUT****Description**

replace the specified Image

**Table 2.14. Query parameters**

| Parameter              | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|------------------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dryRun</b>          | <b>string</b> | When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <b>fieldValidation</b> | <b>string</b> | fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered. |

**Table 2.15. Body parameters**

| Parameter   | Type                         | Description |
|-------------|------------------------------|-------------|
| <b>body</b> | <a href="#">Image</a> schema |             |

**Table 2.16. HTTP responses**

| HTTP code          | Reponse body                 |
|--------------------|------------------------------|
| 200 - OK           | <a href="#">Image</a> schema |
| 201 - Created      | <a href="#">Image</a> schema |
| 401 - Unauthorized | Empty                        |

#### 2.2.4. /apis/image.openshift.io/v1/watch/images/{name}

Table 2.17. Global path parameters

| Parameter   | Type          | Description       |
|-------------|---------------|-------------------|
| <b>name</b> | <b>string</b> | name of the Image |

#### HTTP method

##### GET

#### Description

watch changes to an object of kind Image. deprecated: use the 'watch' parameter with a list operation instead, filtered to a single item with the 'fieldSelector' parameter.

Table 2.18. HTTP responses

| HTTP code          | Reponse body                      |
|--------------------|-----------------------------------|
| 200 - OK           | <a href="#">WatchEvent</a> schema |
| 401 - Unauthorized | Empty                             |

## CHAPTER 3. IMAGESIGNATURE [IMAGE.OPENSIFT.IO/V1]

### Description

ImageSignature holds a signature of an image. It allows to verify image identity and possibly other claims as long as the signature is trusted. Based on this information it is possible to restrict runnable images to those matching cluster-wide policy. Mandatory fields should be parsed by clients doing image verification. The others are parsed from signature's content by the server. They serve just an informative purpose.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

### Type

**object**

### Required

- **type**
- **content**

## 3.1. SPECIFICATION

| Property            | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                   |
|---------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>apiVersion</b>   | <b>string</b> | APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources</a> |
| <b>conditions</b>   | <b>array</b>  | Conditions represent the latest available observations of a signature's current state.                                                                                                                                                                                                                                                                                                                        |
| <b>conditions[]</b> | <b>object</b> | SignatureCondition describes an image signature condition of particular kind at particular probe time.                                                                                                                                                                                                                                                                                                        |
| <b>content</b>      | <b>string</b> | Required: An opaque binary string which is an image's signature.                                                                                                                                                                                                                                                                                                                                              |
| <b>created</b>      | <b>Time</b>   | If specified, it is the time of signature's creation.                                                                                                                                                                                                                                                                                                                                                         |

| Property             | Type                   | Description                                                                                                                                                                                                                                                                                                                                                                                                         |
|----------------------|------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>imageIdentity</b> | <b>string</b>          | A human readable string representing image's identity. It could be a product name and version, or an image pull spec (e.g. "registry.access.redhat.com/rhel7/rhel:7.2").                                                                                                                                                                                                                                            |
| <b>issuedBy</b>      | <b>object</b>          | SignatureIssuer holds information about an issuer of signing certificate or key.                                                                                                                                                                                                                                                                                                                                    |
| <b>issuedTo</b>      | <b>object</b>          | SignatureSubject holds information about a person or entity who created the signature.                                                                                                                                                                                                                                                                                                                              |
| <b>kind</b>          | <b>string</b>          | Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info:<br><a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds</a> |
| <b>metadata</b>      | <b>ObjectMeta_v2</b>   | metadata is the standard object's metadata. More info:<br><a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata</a>                                                                                                                                                   |
| <b>signedClaims</b>  | <b>object (string)</b> | Contains claims from the signature.                                                                                                                                                                                                                                                                                                                                                                                 |
| <b>type</b>          | <b>string</b>          | Required: Describes a type of stored blob.                                                                                                                                                                                                                                                                                                                                                                          |

### 3.1.1. .conditions

#### Description

Conditions represent the latest available observations of a signature's current state.

#### Type

**array**

### 3.1.2. .conditions[]

#### Description

SignatureCondition describes an image signature condition of particular kind at particular probe time.

#### Type

**object**

#### Required

- **type**
- **status**

| Property                  | Type          | Description                                                      |
|---------------------------|---------------|------------------------------------------------------------------|
| <b>lastProbeTime</b>      | <b>Time</b>   | Last time the condition was checked.                             |
| <b>lastTransitionTime</b> | <b>Time</b>   | Last time the condition transit from one status to another.      |
| <b>message</b>            | <b>string</b> | Human readable message indicating details about last transition. |
| <b>reason</b>             | <b>string</b> | (brief) reason for the condition's last transition.              |
| <b>status</b>             | <b>string</b> | Status of the condition, one of True, False, Unknown.            |
| <b>type</b>               | <b>string</b> | Type of signature condition, Complete or Failed.                 |

### 3.1.3. .issuedBy

#### Description

SignatureIssuer holds information about an issuer of signing certificate or key.

#### Type

**object**

| Property            | Type          | Description                                   |
|---------------------|---------------|-----------------------------------------------|
| <b>commonName</b>   | <b>string</b> | Common name (e.g. openshift-signing-service). |
| <b>organization</b> | <b>string</b> | Organization name.                            |

### 3.1.4. .issuedTo

#### Description

SignatureSubject holds information about a person or entity who created the signature.

#### Type

**object**

#### Required

- **publicKeyID**

| Property            | Type          | Description                                                                                                                                                                                                       |
|---------------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>commonName</b>   | <b>string</b> | Common name (e.g. openshift-signing-service).                                                                                                                                                                     |
| <b>organization</b> | <b>string</b> | Organization name.                                                                                                                                                                                                |
| <b>publicKeyID</b>  | <b>string</b> | If present, it is a human readable key id of public key belonging to the subject used to verify image signature. It should contain at least 64 lowest bits of public key's fingerprint (e.g. 0x685ebe62bf278440). |

## 3.2. API ENDPOINTS

The following API endpoints are available:

- **/apis/image.openshift.io/v1/imagesignatures**
  - **POST**: create an ImageSignature
- **/apis/image.openshift.io/v1/imagesignatures/{name}**
  - **DELETE**: delete an ImageSignature

### 3.2.1. /apis/image.openshift.io/v1/imagesignatures

Table 3.1. Global query parameters

| Parameter     | Type          | Description                                                                                                                                                                                                                                              |
|---------------|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dryRun</b> | <b>string</b> | When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed |

| Parameter              | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|------------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>fieldValidation</b> | <b>string</b> | fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: <ul style="list-style-type: none"> <li>- Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23.</li> <li>- Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+</li> <li>- Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.</li> </ul> |

**HTTP method****POST****Description**

create an ImageSignature

**Table 3.2. Body parameters**

| Parameter   | Type                            | Description |
|-------------|---------------------------------|-------------|
| <b>body</b> | <b>ImageSignature</b><br>schema |             |

**Table 3.3. HTTP responses**

| HTTP code          | Response body                |
|--------------------|------------------------------|
| 200 - OK           | <b>ImageSignature</b> schema |
| 201 - Created      | <b>ImageSignature</b> schema |
| 202 - Accepted     | <b>ImageSignature</b> schema |
| 401 - Unauthorized | Empty                        |

### 3.2.2. /apis/image.openshift.io/v1/imagesignatures/{name}

Table 3.4. Global path parameters

| Parameter   | Type          | Description                |
|-------------|---------------|----------------------------|
| <b>name</b> | <b>string</b> | name of the ImageSignature |

Table 3.5. Global query parameters

| Parameter     | Type          | Description                                                                                                                                                                                                                                              |
|---------------|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dryRun</b> | <b>string</b> | When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed |

#### HTTP method

#### DELETE

#### Description

delete an ImageSignature

Table 3.6. HTTP responses

| HTTP code          | Reponse body                     |
|--------------------|----------------------------------|
| 200 - OK           | <a href="#">Status_v5</a> schema |
| 202 - Accepted     | <a href="#">Status_v5</a> schema |
| 401 - Unauthorized | Empty                            |

## CHAPTER 4. IMAGESTREAMIMAGE

### [IMAGE.OPENSIFT.IO/V1]

#### Description

ImageStreamImage represents an Image that is retrieved by image name from an ImageStream. User interfaces and regular users can use this resource to access the metadata details of a tagged image in the image stream history for viewing, since Image resources are not directly accessible to end users. A not found error will be returned if no such image is referenced by a tag within the ImageStream. Images are created when spec tags are set on an image stream that represent an image in an external registry, when pushing to the integrated registry, or when tagging an existing image from one image stream to another. The name of an image stream image is in the form "<STREAM>@<DIGEST>", where the digest is the content addressable identifier for the image (sha256:xxxxx...). You can use ImageStreamImages as the from.kind of an image stream spec tag to reference an image exactly. The only operations supported on the imagestreamimage endpoint are retrieving the image.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

#### Type

**object**

#### Required

- **image**

### 4.1. SPECIFICATION

| Property          | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                   |
|-------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>apiVersion</b> | <b>string</b> | APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources</a> |

| Property        | Type                 | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|-----------------|----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>image</b>    | <b>object</b>        | <p>Image is an immutable representation of a container image and metadata at a point in time. Images are named by taking a hash of their contents (metadata and content) and any change in format, content, or metadata results in a new name. The images resource is primarily for use by cluster administrators and integrations like the cluster image registry - end users instead access images via the imagestreamtags or imagestreamimages resources. While image metadata is stored in the API, any integration that implements the container image registry API must provide its own storage for the raw manifest data, image config, and layer contents.</p> <p>Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).</p> |
| <b>kind</b>     | <b>string</b>        | <p>Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info:</p> <p><a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds</a></p>                                                                                                                                                                                                                                                                                                                                                                           |
| <b>metadata</b> | <b>ObjectMeta_v2</b> | <p>metadata is the standard object's metadata. More info:</p> <p><a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata</a></p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |

### 4.1.1. .image

#### Description

Image is an immutable representation of a container image and metadata at a point in time. Images

are named by taking a hash of their contents (metadata and content) and any change in format, content, or metadata results in a new name. The images resource is primarily for use by cluster administrators and integrations like the cluster image registry – end users instead access images via the `imagestreamtags` or `imagestreamimages` resources. While image metadata is stored in the API, any integration that implements the container image registry API must provide its own storage for the raw manifest data, image config, and layer contents.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

## Type

### object

| Property                            | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                   |
|-------------------------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>apiVersion</b>                   | <b>string</b> | APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources</a> |
| <b>dockerImageConfig</b>            | <b>string</b> | DockerImageConfig is a JSON blob that the runtime uses to set up the container. This is a part of manifest schema v2. Will not be set when the image represents a manifest list.                                                                                                                                                                                                                              |
| <b>dockerImageLayers</b>            | <b>array</b>  | DockerImageLayers represents the layers in the image. May not be set if the image does not define that data or if the image represents a manifest list.                                                                                                                                                                                                                                                       |
| <b>dockerImageLayers[]</b>          | <b>object</b> | ImageLayer represents a single layer of the image. Some images may have multiple layers. Some may have none.                                                                                                                                                                                                                                                                                                  |
| <b>dockerImageManifest</b>          | <b>string</b> | DockerImageManifest is the raw JSON of the manifest                                                                                                                                                                                                                                                                                                                                                           |
| <b>dockerImageManifestMediaType</b> | <b>string</b> | DockerImageManifestMediaType specifies the mediaType of manifest. This is a part of manifest schema v2.                                                                                                                                                                                                                                                                                                       |

| Property                          | Type                  | Description                                                                                                                                                                                                                                                                                                                                                                                                      |
|-----------------------------------|-----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dockerImageManifests</b>       | <b>array</b>          | DockerImageManifests holds information about sub-manifests when the image represents a manifest list. When this field is present, no DockerImageLayers should be specified.                                                                                                                                                                                                                                      |
| <b>dockerImageManifests[]</b>     | <b>object</b>         | ImageManifest represents sub-manifests of a manifest list. The Digest field points to a regular Image object.                                                                                                                                                                                                                                                                                                    |
| <b>dockerImageMetadata</b>        | <b>RawExtension</b>   | DockerImageMetadata contains metadata about this image                                                                                                                                                                                                                                                                                                                                                           |
| <b>dockerImageMetadataVersion</b> | <b>string</b>         | DockerImageMetadataVersion conveys the version of the object, which if empty defaults to "1.0"                                                                                                                                                                                                                                                                                                                   |
| <b>dockerImageReference</b>       | <b>string</b>         | DockerImageReference is the string that can be used to pull this image.                                                                                                                                                                                                                                                                                                                                          |
| <b>dockerImageSignatures</b>      | <b>array (string)</b> | DockerImageSignatures provides the signatures as opaque blobs. This is a part of manifest schema v1.                                                                                                                                                                                                                                                                                                             |
| <b>kind</b>                       | <b>string</b>         | Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds</a> |
| <b>metadata</b>                   | <b>ObjectMeta_v2</b>  | metadata is the standard object's metadata. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata</a>                                                                                                                                                   |
| <b>signatures</b>                 | <b>array</b>          | Signatures holds all signatures of the image.                                                                                                                                                                                                                                                                                                                                                                    |

| Property            | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|---------------------|---------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>signatures[]</b> | <b>object</b> | <p>ImageSignature holds a signature of an image. It allows to verify image identity and possibly other claims as long as the signature is trusted. Based on this information it is possible to restrict runnable images to those matching cluster-wide policy. Mandatory fields should be parsed by clients doing image verification. The others are parsed from signature's content by the server. They serve just an informative purpose.</p> <p>Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).</p> |

#### 4.1.2. .image.dockerImageLayers

##### Description

DockerImageLayers represents the layers in the image. May not be set if the image does not define that data or if the image represents a manifest list.

##### Type

**array**

#### 4.1.3. .image.dockerImageLayers[]

##### Description

ImageLayer represents a single layer of the image. Some images may have multiple layers. Some may have none.

##### Type

**object**

##### Required

- **name**
- **size**
- **mediaType**

| Property         | Type          | Description                         |
|------------------|---------------|-------------------------------------|
| <b>mediaType</b> | <b>string</b> | MediaType of the referenced object. |

| Property    | Type           | Description                                                    |
|-------------|----------------|----------------------------------------------------------------|
| <b>name</b> | <b>string</b>  | Name of the layer as defined by the underlying store.          |
| <b>size</b> | <b>integer</b> | Size of the layer in bytes as defined by the underlying store. |

#### 4.1.4. .image.dockerImageManifests

##### Description

DockerImageManifests holds information about sub-manifests when the image represents a manifest list. When this field is present, no DockerImageLayers should be specified.

##### Type

**array**

#### 4.1.5. .image.dockerImageManifests[]

##### Description

ImageManifest represents sub-manifests of a manifest list. The Digest field points to a regular Image object.

##### Type

**object**

##### Required

- **digest**
- **mediaType**
- **manifestSize**
- **architecture**
- **os**

| Property            | Type          | Description                                                                                         |
|---------------------|---------------|-----------------------------------------------------------------------------------------------------|
| <b>architecture</b> | <b>string</b> | Architecture specifies the supported CPU architecture, for example <b>amd64</b> or <b>ppc64le</b> . |
| <b>digest</b>       | <b>string</b> | Digest is the unique identifier for the manifest. It refers to an Image object.                     |

| Property            | Type           | Description                                                                                                                                                                                                               |
|---------------------|----------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>manifestSize</b> | <b>integer</b> | ManifestSize represents the size of the raw object contents, in bytes.                                                                                                                                                    |
| <b>mediaType</b>    | <b>string</b>  | MediaType defines the type of the manifest, possible values are application/vnd.oci.image.manifest.v1+json, application/vnd.docker.distribution.manifest.v2+json or application/vnd.docker.distribution.manifest.v1+json. |
| <b>os</b>           | <b>string</b>  | OS specifies the operating system, for example <b>linux</b> .                                                                                                                                                             |
| <b>variant</b>      | <b>string</b>  | Variant is an optional field representing a variant of the CPU, for example v6 to specify a particular CPU variant of the ARM CPU.                                                                                        |

#### 4.1.6. .image.signatures

##### Description

Signatures holds all signatures of the image.

##### Type

**array**

#### 4.1.7. .image.signatures[]

##### Description

ImageSignature holds a signature of an image. It allows to verify image identity and possibly other claims as long as the signature is trusted. Based on this information it is possible to restrict runnable images to those matching cluster-wide policy. Mandatory fields should be parsed by clients doing image verification. The others are parsed from signature's content by the server. They serve just an informative purpose.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

##### Type

**object**

##### Required

- **type**
- **content**

| Property             | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                   |
|----------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>apiVersion</b>    | <b>string</b> | APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources</a> |
| <b>conditions</b>    | <b>array</b>  | Conditions represent the latest available observations of a signature's current state.                                                                                                                                                                                                                                                                                                                        |
| <b>conditions[]</b>  | <b>object</b> | SignatureCondition describes an image signature condition of particular kind at particular probe time.                                                                                                                                                                                                                                                                                                        |
| <b>content</b>       | <b>string</b> | Required: An opaque binary string which is an image's signature.                                                                                                                                                                                                                                                                                                                                              |
| <b>created</b>       | <b>Time</b>   | If specified, it is the time of signature's creation.                                                                                                                                                                                                                                                                                                                                                         |
| <b>imageIdentity</b> | <b>string</b> | A human readable string representing image's identity. It could be a product name and version, or an image pull spec (e.g. "registry.access.redhat.com/rhel7/rhel:7.2").                                                                                                                                                                                                                                      |
| <b>issuedBy</b>      | <b>object</b> | SignatureIssuer holds information about an issuer of signing certificate or key.                                                                                                                                                                                                                                                                                                                              |
| <b>issuedTo</b>      | <b>object</b> | SignatureSubject holds information about a person or entity who created the signature.                                                                                                                                                                                                                                                                                                                        |

| Property            | Type                                 | Description                                                                                                                                                                                                                                                                                                                                                                                                      |
|---------------------|--------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>kind</b>         | <b>string</b>                        | Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds</a> |
| <b>metadata</b>     | <b><a href="#">ObjectMeta_v2</a></b> | metadata is the standard object's metadata. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata</a>                                                                                                                                                   |
| <b>signedClaims</b> | <b>object (string)</b>               | Contains claims from the signature.                                                                                                                                                                                                                                                                                                                                                                              |
| <b>type</b>         | <b>string</b>                        | Required: Describes a type of stored blob.                                                                                                                                                                                                                                                                                                                                                                       |

#### 4.1.8. .image.signatures[].conditions

##### Description

Conditions represent the latest available observations of a signature's current state.

##### Type

**array**

#### 4.1.9. .image.signatures[].conditions[]

##### Description

SignatureCondition describes an image signature condition of particular kind at particular probe time.

##### Type

**object**

##### Required

- **type**
- **status**

| Property                  | Type          | Description                                                      |
|---------------------------|---------------|------------------------------------------------------------------|
| <b>lastProbeTime</b>      | <b>Time</b>   | Last time the condition was checked.                             |
| <b>lastTransitionTime</b> | <b>Time</b>   | Last time the condition transit from one status to another.      |
| <b>message</b>            | <b>string</b> | Human readable message indicating details about last transition. |
| <b>reason</b>             | <b>string</b> | (brief) reason for the condition's last transition.              |
| <b>status</b>             | <b>string</b> | Status of the condition, one of True, False, Unknown.            |
| <b>type</b>               | <b>string</b> | Type of signature condition, Complete or Failed.                 |

#### 4.1.10. .image.signatures[].issuedBy

##### Description

SignatureIssuer holds information about an issuer of signing certificate or key.

##### Type

**object**

| Property            | Type          | Description                                   |
|---------------------|---------------|-----------------------------------------------|
| <b>commonName</b>   | <b>string</b> | Common name (e.g. openshift-signing-service). |
| <b>organization</b> | <b>string</b> | Organization name.                            |

#### 4.1.11. .image.signatures[].issuedTo

##### Description

SignatureSubject holds information about a person or entity who created the signature.

##### Type

**object**

##### Required

- **publicKeyID**

| Property            | Type          | Description                                                                                                                                                                                                       |
|---------------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>commonName</b>   | <b>string</b> | Common name (e.g. openshift-signing-service).                                                                                                                                                                     |
| <b>organization</b> | <b>string</b> | Organization name.                                                                                                                                                                                                |
| <b>publicKeyID</b>  | <b>string</b> | If present, it is a human readable key id of public key belonging to the subject used to verify image signature. It should contain at least 64 lowest bits of public key's fingerprint (e.g. 0x685ebe62bf278440). |

## 4.2. API ENDPOINTS

The following API endpoints are available:

- **/apis/image.openshift.io/v1/namespaces/{namespace}/imagestreamimages/{name}**
  - **GET**: read the specified ImageStreamImage

### 4.2.1. /apis/image.openshift.io/v1/namespaces/{namespace}/imagestreamimages/{name}

Table 4.1. Global path parameters

| Parameter   | Type          | Description                  |
|-------------|---------------|------------------------------|
| <b>name</b> | <b>string</b> | name of the ImageStreamImage |

#### HTTP method

**GET**

#### Description

read the specified ImageStreamImage

Table 4.2. HTTP responses

| HTTP code          | Response body                           |
|--------------------|-----------------------------------------|
| 200 - OK           | <a href="#">ImageStreamImage</a> schema |
| 401 - Unauthorized | Empty                                   |

## CHAPTER 5. IMAGESTREAMIMPORT [IMAGE.OPENSIFT.IO/V1]

### Description

The image stream import resource provides an easy way for a user to find and import container images from other container image registries into the server. Individual images or an entire image repository may be imported, and users may choose to see the results of the import prior to tagging the resulting images into the specified image stream.

This API is intended for end-user tools that need to see the metadata of the image prior to import (for instance, to generate an application from it). Clients that know the desired image can continue to create spec.tags directly into their image streams.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

### Type

**object**

### Required

- **spec**
- **status**

## 5.1. SPECIFICATION

| Property          | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                      |
|-------------------|---------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>apiVersion</b> | <b>string</b> | APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources</a>    |
| <b>kind</b>       | <b>string</b> | Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds</a> |

| Property        | Type                          | Description                                                                                                                                                                                                                                                    |
|-----------------|-------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>metadata</b> | <a href="#">ObjectMeta_v2</a> | metadata is the standard object's metadata. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata</a> |
| <b>spec</b>     | <b>object</b>                 | ImageStreamImportSpec defines what images should be imported.                                                                                                                                                                                                  |
| <b>status</b>   | <b>object</b>                 | ImageStreamImportStatus contains information about the status of an image stream import.                                                                                                                                                                       |

### 5.1.1. .spec

#### Description

ImageStreamImportSpec defines what images should be imported.

#### Type

**object**

#### Required

- **import**

| Property          | Type           | Description                                                                                                                                            |
|-------------------|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>images</b>     | <b>array</b>   | Images are a list of individual images to import.                                                                                                      |
| <b>images[]</b>   | <b>object</b>  | ImageImportSpec describes a request to import a specific image.                                                                                        |
| <b>import</b>     | <b>boolean</b> | Import indicates whether to perform an import - if so, the specified tags are set on the spec and status of the image stream defined by the type meta. |
| <b>repository</b> | <b>object</b>  | RepositoryImportSpec describes a request to import images from a container image repository.                                                           |

### 5.1.2. .spec.images

**Description**

Images are a list of individual images to import.

**Type**

**array**

**5.1.3. .spec.images[]****Description**

ImageImportSpec describes a request to import a specific image.

**Type**

**object**

**Required**

- **from**

| Property               | Type                           | Description                                                                                                                                                                                                                                    |
|------------------------|--------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>from</b>            | <b>ObjectReference</b>         | From is the source of an image to import; only kind DockerImage is allowed                                                                                                                                                                     |
| <b>importPolicy</b>    | <b>object</b>                  | TagImportPolicy controls how images related to this tag will be imported.                                                                                                                                                                      |
| <b>includeManifest</b> | <b>boolean</b>                 | IncludeManifest determines if the manifest for each image is returned in the response                                                                                                                                                          |
| <b>referencePolicy</b> | <b>object</b>                  | TagReferencePolicy describes how pull-specs for images in this image stream tag are generated when image change triggers in deployment configs or builds are resolved. This allows the image stream author to control how images are accessed. |
| <b>to</b>              | <b>LocalObjectReference_v2</b> | To is a tag in the current image stream to assign the imported image to, if name is not specified the default tag from from.name will be used                                                                                                  |

**5.1.4. .spec.images[].importPolicy****Description**

TagImportPolicy controls how images related to this tag will be imported.

Type  
object

| Property          | Type           | Description                                                                                                             |
|-------------------|----------------|-------------------------------------------------------------------------------------------------------------------------|
| <b>importMode</b> | <b>string</b>  | ImportMode describes how to import an image manifest.                                                                   |
| <b>insecure</b>   | <b>boolean</b> | Insecure is true if the server may bypass certificate verification or connect directly over HTTP during image import.   |
| <b>scheduled</b>  | <b>boolean</b> | Scheduled indicates to the server that this tag should be periodically checked to ensure it is up to date, and imported |

### 5.1.5. .spec.images[].referencePolicy

#### Description

TagReferencePolicy describes how pull-specs for images in this image stream tag are generated when image change triggers in deployment configs or builds are resolved. This allows the image stream author to control how images are accessed.

Type  
object

#### Required

- **type**

| Property | Type | Description |
|----------|------|-------------|
|----------|------|-------------|

| Property    | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|-------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>type</b> | <b>string</b> | Type determines how the image pull spec should be transformed when the image stream tag is used in deployment config triggers or new builds. The default value is <b>Source</b> , indicating the original location of the image should be used (if imported). The user may also specify <b>Local</b> , indicating that the pull spec should point to the integrated container image registry and leverage the registry's ability to proxy the pull to an upstream registry. <b>Local</b> allows the credentials used to pull this image to be managed from the image stream's namespace, so others on the platform can access a remote image but have no access to the remote secret. It also allows the image layers to be mirrored into the local registry which the images can still be pulled even if the upstream registry is unavailable. |

### 5.1.6. .spec.repository

#### Description

RepositoryImportSpec describes a request to import images from a container image repository.

#### Type

**object**

#### Required

- **from**

| Property    | Type                   | Description                                                                                                                        |
|-------------|------------------------|------------------------------------------------------------------------------------------------------------------------------------|
| <b>from</b> | <b>ObjectReference</b> | From is the source for the image repository to import; only kind DockerImage and a name of a container image repository is allowed |

| Property               | Type           | Description                                                                                                                                                                                                                                    |
|------------------------|----------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>importPolicy</b>    | <b>object</b>  | TagImportPolicy controls how images related to this tag will be imported.                                                                                                                                                                      |
| <b>includeManifest</b> | <b>boolean</b> | IncludeManifest determines if the manifest for each image is returned in the response                                                                                                                                                          |
| <b>referencePolicy</b> | <b>object</b>  | TagReferencePolicy describes how pull-specs for images in this image stream tag are generated when image change triggers in deployment configs or builds are resolved. This allows the image stream author to control how images are accessed. |

### 5.1.7. .spec.repository.importPolicy

#### Description

TagImportPolicy controls how images related to this tag will be imported.

#### Type

**object**

| Property          | Type           | Description                                                                                                             |
|-------------------|----------------|-------------------------------------------------------------------------------------------------------------------------|
| <b>importMode</b> | <b>string</b>  | ImportMode describes how to import an image manifest.                                                                   |
| <b>insecure</b>   | <b>boolean</b> | Insecure is true if the server may bypass certificate verification or connect directly over HTTP during image import.   |
| <b>scheduled</b>  | <b>boolean</b> | Scheduled indicates to the server that this tag should be periodically checked to ensure it is up to date, and imported |

### 5.1.8. .spec.repository.referencePolicy

#### Description

TagReferencePolicy describes how pull-specs for images in this image stream tag are generated when image change triggers in deployment configs or builds are resolved. This allows the image stream author to control how images are accessed.

#### Type

**object****Required**

- **type**

| Property    | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|-------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>type</b> | <b>string</b> | Type determines how the image pull spec should be transformed when the image stream tag is used in deployment config triggers or new builds. The default value is <b>Source</b> , indicating the original location of the image should be used (if imported). The user may also specify <b>Local</b> , indicating that the pull spec should point to the integrated container image registry and leverage the registry's ability to proxy the pull to an upstream registry. <b>Local</b> allows the credentials used to pull this image to be managed from the image stream's namespace, so others on the platform can access a remote image but have no access to the remote secret. It also allows the image layers to be mirrored into the local registry which the images can still be pulled even if the upstream registry is unavailable. |

### 5.1.9. .status

**Description**

ImageStreamImportStatus contains information about the status of an image stream import.

**Type****object**

| Property        | Type          | Description                                                |
|-----------------|---------------|------------------------------------------------------------|
| <b>images</b>   | <b>array</b>  | Images is set with the result of importing spec.images     |
| <b>images[]</b> | <b>object</b> | ImageImportStatus describes the result of an image import. |

| Property          | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|-------------------|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>import</b>     | <b>object</b> | <p>An ImageStream stores a mapping of tags to images, metadata overrides that are applied when images are tagged in a stream, and an optional reference to a container image repository on a registry. Users typically update the spec.tags field to point to external images which are imported from container registries using credentials in your namespace with the pull secret type, or to existing image stream tags and images which are immediately accessible for tagging or pulling. The history of images applied to a tag is visible in the status.tags field and any user who can view an image stream is allowed to tag that image into their own image streams. Access to pull images from the integrated registry is granted by having the "get imagestreams/layers" permission on a given image stream. Users may remove a tag by deleting the imagestreamtag resource, which causes both spec and status for that tag to be removed. Image stream history is retained until an administrator runs the prune operation, which removes references that are no longer in use. To preserve a historical image, ensure there is a tag in spec pointing to that image by its digest.</p> <p>Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).</p> |
| <b>repository</b> | <b>object</b> | RepositoryImportStatus describes the result of an image repository import                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |

### 5.1.10. .status.images

**Description**

Images is set with the result of importing spec.images

**Type**

**array**

**5.1.11. .status.images[]****Description**

ImageImportStatus describes the result of an image import.

**Type**

**object**

**Required**

- **status**

| Property         | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|------------------|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>image</b>     | <b>object</b> | <p>Image is an immutable representation of a container image and metadata at a point in time. Images are named by taking a hash of their contents (metadata and content) and any change in format, content, or metadata results in a new name. The images resource is primarily for use by cluster administrators and integrations like the cluster image registry - end users instead access images via the imagestreamtags or imagestreamimages resources. While image metadata is stored in the API, any integration that implements the container image registry API must provide its own storage for the raw manifest data, image config, and layer contents.</p> <p>Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).</p> |
| <b>manifests</b> | <b>array</b>  | Manifests holds sub-manifests metadata when importing a manifest list                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |

| Property           | Type             | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|--------------------|------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>manifests[]</b> | <b>object</b>    | <p>Image is an immutable representation of a container image and metadata at a point in time. Images are named by taking a hash of their contents (metadata and content) and any change in format, content, or metadata results in a new name. The images resource is primarily for use by cluster administrators and integrations like the cluster image registry - end users instead access images via the imagestreamtags or imagestreamimages resources. While image metadata is stored in the API, any integration that implements the container image registry API must provide its own storage for the raw manifest data, image config, and layer contents.</p> <p>Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).</p> |
| <b>status</b>      | <b>Status_v5</b> | Status is the status of the image import, including errors encountered while retrieving the image                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| <b>tag</b>         | <b>string</b>    | Tag is the tag this image was located under, if any                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |

### 5.1.12. .status.images[].image

#### Description

Image is an immutable representation of a container image and metadata at a point in time. Images are named by taking a hash of their contents (metadata and content) and any change in format, content, or metadata results in a new name. The images resource is primarily for use by cluster administrators and integrations like the cluster image registry - end users instead access images via the imagestreamtags or imagestreamimages resources. While image metadata is stored in the API, any integration that implements the container image registry API must provide its own storage for the raw manifest data, image config, and layer contents.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

#### Type

**object**

| Property                            | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                   |
|-------------------------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>apiVersion</b>                   | <b>string</b> | APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources</a> |
| <b>dockerImageConfig</b>            | <b>string</b> | DockerImageConfig is a JSON blob that the runtime uses to set up the container. This is a part of manifest schema v2. Will not be set when the image represents a manifest list.                                                                                                                                                                                                                              |
| <b>dockerImageLayers</b>            | <b>array</b>  | DockerImageLayers represents the layers in the image. May not be set if the image does not define that data or if the image represents a manifest list.                                                                                                                                                                                                                                                       |
| <b>dockerImageLayers[]</b>          | <b>object</b> | ImageLayer represents a single layer of the image. Some images may have multiple layers. Some may have none.                                                                                                                                                                                                                                                                                                  |
| <b>dockerImageManifest</b>          | <b>string</b> | DockerImageManifest is the raw JSON of the manifest                                                                                                                                                                                                                                                                                                                                                           |
| <b>dockerImageManifestMediaType</b> | <b>string</b> | DockerImageManifestMediaType specifies the mediaType of manifest. This is a part of manifest schema v2.                                                                                                                                                                                                                                                                                                       |
| <b>dockerImageManifests</b>         | <b>array</b>  | DockerImageManifests holds information about sub-manifests when the image represents a manifest list. When this field is present, no DockerImageLayers should be specified.                                                                                                                                                                                                                                   |
| <b>dockerImageManifests[]</b>       | <b>object</b> | ImageManifest represents sub-manifests of a manifest list. The Digest field points to a regular Image object.                                                                                                                                                                                                                                                                                                 |

| Property                          | Type                          | Description                                                                                                                                                                                                                                                                                                                                                                                                      |
|-----------------------------------|-------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dockerImageMetadata</b>        | <a href="#">RawExtension</a>  | DockerImageMetadata contains metadata about this image                                                                                                                                                                                                                                                                                                                                                           |
| <b>dockerImageMetadataVersion</b> | <b>string</b>                 | DockerImageMetadataVersion conveys the version of the object, which if empty defaults to "1.0"                                                                                                                                                                                                                                                                                                                   |
| <b>dockerImageReference</b>       | <b>string</b>                 | DockerImageReference is the string that can be used to pull this image.                                                                                                                                                                                                                                                                                                                                          |
| <b>dockerImageSignatures</b>      | <b>array (string)</b>         | DockerImageSignatures provides the signatures as opaque blobs. This is a part of manifest schema v1.                                                                                                                                                                                                                                                                                                             |
| <b>kind</b>                       | <b>string</b>                 | Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds</a> |
| <b>metadata</b>                   | <a href="#">ObjectMeta_v2</a> | metadata is the standard object's metadata. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata</a>                                                                                                                                                   |
| <b>signatures</b>                 | <b>array</b>                  | Signatures holds all signatures of the image.                                                                                                                                                                                                                                                                                                                                                                    |

| Property            | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|---------------------|---------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>signatures[]</b> | <b>object</b> | <p>ImageSignature holds a signature of an image. It allows to verify image identity and possibly other claims as long as the signature is trusted. Based on this information it is possible to restrict runnable images to those matching cluster-wide policy. Mandatory fields should be parsed by clients doing image verification. The others are parsed from signature's content by the server. They serve just an informative purpose.</p> <p>Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).</p> |

### 5.1.13. .status.images[].image.dockerImageLayers

#### Description

DockerImageLayers represents the layers in the image. May not be set if the image does not define that data or if the image represents a manifest list.

#### Type

**array**

### 5.1.14. .status.images[].image.dockerImageLayers[]

#### Description

ImageLayer represents a single layer of the image. Some images may have multiple layers. Some may have none.

#### Type

**object**

#### Required

- **name**
- **size**
- **mediaType**

| Property         | Type          | Description                         |
|------------------|---------------|-------------------------------------|
| <b>mediaType</b> | <b>string</b> | MediaType of the referenced object. |

| Property    | Type           | Description                                                    |
|-------------|----------------|----------------------------------------------------------------|
| <b>name</b> | <b>string</b>  | Name of the layer as defined by the underlying store.          |
| <b>size</b> | <b>integer</b> | Size of the layer in bytes as defined by the underlying store. |

### 5.1.15. `.status.images[].image.dockerImageManifests`

#### Description

DockerImageManifests holds information about sub-manifests when the image represents a manifest list. When this field is present, no DockerImageLayers should be specified.

#### Type

**array**

### 5.1.16. `.status.images[].image.dockerImageManifests[]`

#### Description

ImageManifest represents sub-manifests of a manifest list. The Digest field points to a regular Image object.

#### Type

**object**

#### Required

- **digest**
- **mediaType**
- **manifestSize**
- **architecture**
- **os**

| Property            | Type          | Description                                                                                         |
|---------------------|---------------|-----------------------------------------------------------------------------------------------------|
| <b>architecture</b> | <b>string</b> | Architecture specifies the supported CPU architecture, for example <b>amd64</b> or <b>ppc64le</b> . |
| <b>digest</b>       | <b>string</b> | Digest is the unique identifier for the manifest. It refers to an Image object.                     |

| Property            | Type           | Description                                                                                                                                                                                                               |
|---------------------|----------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>manifestSize</b> | <b>integer</b> | ManifestSize represents the size of the raw object contents, in bytes.                                                                                                                                                    |
| <b>mediaType</b>    | <b>string</b>  | MediaType defines the type of the manifest, possible values are application/vnd.oci.image.manifest.v1+json, application/vnd.docker.distribution.manifest.v2+json or application/vnd.docker.distribution.manifest.v1+json. |
| <b>os</b>           | <b>string</b>  | OS specifies the operating system, for example <b>linux</b> .                                                                                                                                                             |
| <b>variant</b>      | <b>string</b>  | Variant is an optional field representing a variant of the CPU, for example v6 to specify a particular CPU variant of the ARM CPU.                                                                                        |

### 5.1.17. .status.images[].image.signatures

#### Description

Signatures holds all signatures of the image.

#### Type

**array**

### 5.1.18. .status.images[].image.signatures[]

#### Description

ImageSignature holds a signature of an image. It allows to verify image identity and possibly other claims as long as the signature is trusted. Based on this information it is possible to restrict runnable images to those matching cluster-wide policy. Mandatory fields should be parsed by clients doing image verification. The others are parsed from signature's content by the server. They serve just an informative purpose.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

#### Type

**object**

#### Required

- **type**
- **content**

| Property             | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                   |
|----------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>apiVersion</b>    | <b>string</b> | APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources</a> |
| <b>conditions</b>    | <b>array</b>  | Conditions represent the latest available observations of a signature's current state.                                                                                                                                                                                                                                                                                                                        |
| <b>conditions[]</b>  | <b>object</b> | SignatureCondition describes an image signature condition of particular kind at particular probe time.                                                                                                                                                                                                                                                                                                        |
| <b>content</b>       | <b>string</b> | Required: An opaque binary string which is an image's signature.                                                                                                                                                                                                                                                                                                                                              |
| <b>created</b>       | <b>Time</b>   | If specified, it is the time of signature's creation.                                                                                                                                                                                                                                                                                                                                                         |
| <b>imageIdentity</b> | <b>string</b> | A human readable string representing image's identity. It could be a product name and version, or an image pull spec (e.g. "registry.access.redhat.com/rhel7/rhel:7.2").                                                                                                                                                                                                                                      |
| <b>issuedBy</b>      | <b>object</b> | SignatureIssuer holds information about an issuer of signing certificate or key.                                                                                                                                                                                                                                                                                                                              |
| <b>issuedTo</b>      | <b>object</b> | SignatureSubject holds information about a person or entity who created the signature.                                                                                                                                                                                                                                                                                                                        |

| Property            | Type                   | Description                                                                                                                                                                                                                                                                                                                                                                                                      |
|---------------------|------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>kind</b>         | <b>string</b>          | Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds</a> |
| <b>metadata</b>     | <b>ObjectMeta_v2</b>   | metadata is the standard object's metadata. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata</a>                                                                                                                                                   |
| <b>signedClaims</b> | <b>object (string)</b> | Contains claims from the signature.                                                                                                                                                                                                                                                                                                                                                                              |
| <b>type</b>         | <b>string</b>          | Required: Describes a type of stored blob.                                                                                                                                                                                                                                                                                                                                                                       |

### 5.1.19. .status.images[].image.signatures[].conditions

#### Description

Conditions represent the latest available observations of a signature's current state.

#### Type

**array**

### 5.1.20. .status.images[].image.signatures[].conditions[]

#### Description

SignatureCondition describes an image signature condition of particular kind at particular probe time.

#### Type

**object**

#### Required

- **type**
- **status**

| Property                  | Type          | Description                                                      |
|---------------------------|---------------|------------------------------------------------------------------|
| <b>lastProbeTime</b>      | <b>Time</b>   | Last time the condition was checked.                             |
| <b>lastTransitionTime</b> | <b>Time</b>   | Last time the condition transit from one status to another.      |
| <b>message</b>            | <b>string</b> | Human readable message indicating details about last transition. |
| <b>reason</b>             | <b>string</b> | (brief) reason for the condition's last transition.              |
| <b>status</b>             | <b>string</b> | Status of the condition, one of True, False, Unknown.            |
| <b>type</b>               | <b>string</b> | Type of signature condition, Complete or Failed.                 |

### 5.1.21. `.status.images[].image.signatures[].issuedBy`

#### Description

SignatureIssuer holds information about an issuer of signing certificate or key.

#### Type

**object**

| Property            | Type          | Description                                   |
|---------------------|---------------|-----------------------------------------------|
| <b>commonName</b>   | <b>string</b> | Common name (e.g. openshift-signing-service). |
| <b>organization</b> | <b>string</b> | Organization name.                            |

### 5.1.22. `.status.images[].image.signatures[].issuedTo`

#### Description

SignatureSubject holds information about a person or entity who created the signature.

#### Type

**object**

#### Required

- **publicKeyID**

| Property            | Type          | Description                                                                                                                                                                                                       |
|---------------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>commonName</b>   | <b>string</b> | Common name (e.g. openshift-signing-service).                                                                                                                                                                     |
| <b>organization</b> | <b>string</b> | Organization name.                                                                                                                                                                                                |
| <b>publicKeyID</b>  | <b>string</b> | If present, it is a human readable key id of public key belonging to the subject used to verify image signature. It should contain at least 64 lowest bits of public key's fingerprint (e.g. 0x685ebe62bf278440). |

### 5.1.23. .status.images[].manifests

#### Description

Manifests holds sub-manifests metadata when importing a manifest list

#### Type

**array**

### 5.1.24. .status.images[].manifests[]

#### Description

Image is an immutable representation of a container image and metadata at a point in time. Images are named by taking a hash of their contents (metadata and content) and any change in format, content, or metadata results in a new name. The images resource is primarily for use by cluster administrators and integrations like the cluster image registry - end users instead access images via the imagestreamtags or imagestreamimages resources. While image metadata is stored in the API, any integration that implements the container image registry API must provide its own storage for the raw manifest data, image config, and layer contents.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

#### Type

**object**

| Property | Type | Description |
|----------|------|-------------|
|----------|------|-------------|

| Property                            | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                   |
|-------------------------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>apiVersion</b>                   | <b>string</b> | APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources</a> |
| <b>dockerImageConfig</b>            | <b>string</b> | DockerImageConfig is a JSON blob that the runtime uses to set up the container. This is a part of manifest schema v2. Will not be set when the image represents a manifest list.                                                                                                                                                                                                                              |
| <b>dockerImageLayers</b>            | <b>array</b>  | DockerImageLayers represents the layers in the image. May not be set if the image does not define that data or if the image represents a manifest list.                                                                                                                                                                                                                                                       |
| <b>dockerImageLayers[]</b>          | <b>object</b> | ImageLayer represents a single layer of the image. Some images may have multiple layers. Some may have none.                                                                                                                                                                                                                                                                                                  |
| <b>dockerImageManifest</b>          | <b>string</b> | DockerImageManifest is the raw JSON of the manifest                                                                                                                                                                                                                                                                                                                                                           |
| <b>dockerImageManifestMediaType</b> | <b>string</b> | DockerImageManifestMediaType specifies the mediaType of manifest. This is a part of manifest schema v2.                                                                                                                                                                                                                                                                                                       |
| <b>dockerImageManifests</b>         | <b>array</b>  | DockerImageManifests holds information about sub-manifests when the image represents a manifest list. When this field is present, no DockerImageLayers should be specified.                                                                                                                                                                                                                                   |
| <b>dockerImageManifests[]</b>       | <b>object</b> | ImageManifest represents sub-manifests of a manifest list. The Digest field points to a regular Image object.                                                                                                                                                                                                                                                                                                 |

| Property                          | Type                  | Description                                                                                                                                                                                                                                                                                                                                                                                                      |
|-----------------------------------|-----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dockerImageMetadata</b>        | <b>RawExtension</b>   | DockerImageMetadata contains metadata about this image                                                                                                                                                                                                                                                                                                                                                           |
| <b>dockerImageMetadataVersion</b> | <b>string</b>         | DockerImageMetadataVersion conveys the version of the object, which if empty defaults to "1.0"                                                                                                                                                                                                                                                                                                                   |
| <b>dockerImageReference</b>       | <b>string</b>         | DockerImageReference is the string that can be used to pull this image.                                                                                                                                                                                                                                                                                                                                          |
| <b>dockerImageSignatures</b>      | <b>array (string)</b> | DockerImageSignatures provides the signatures as opaque blobs. This is a part of manifest schema v1.                                                                                                                                                                                                                                                                                                             |
| <b>kind</b>                       | <b>string</b>         | Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds</a> |
| <b>metadata</b>                   | <b>ObjectMeta_v2</b>  | metadata is the standard object's metadata. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata</a>                                                                                                                                                   |
| <b>signatures</b>                 | <b>array</b>          | Signatures holds all signatures of the image.                                                                                                                                                                                                                                                                                                                                                                    |

| Property            | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|---------------------|---------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>signatures[]</b> | <b>object</b> | <p>ImageSignature holds a signature of an image. It allows to verify image identity and possibly other claims as long as the signature is trusted. Based on this information it is possible to restrict runnable images to those matching cluster-wide policy. Mandatory fields should be parsed by clients doing image verification. The others are parsed from signature's content by the server. They serve just an informative purpose.</p> <p>Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).</p> |

### 5.1.25. .status.images[].manifests[].dockerImageLayers

#### Description

DockerImageLayers represents the layers in the image. May not be set if the image does not define that data or if the image represents a manifest list.

#### Type

**array**

### 5.1.26. .status.images[].manifests[].dockerImageLayers[]

#### Description

ImageLayer represents a single layer of the image. Some images may have multiple layers. Some may have none.

#### Type

**object**

#### Required

- **name**
- **size**
- **mediaType**

| Property         | Type          | Description                         |
|------------------|---------------|-------------------------------------|
| <b>mediaType</b> | <b>string</b> | MediaType of the referenced object. |

| Property    | Type           | Description                                                    |
|-------------|----------------|----------------------------------------------------------------|
| <b>name</b> | <b>string</b>  | Name of the layer as defined by the underlying store.          |
| <b>size</b> | <b>integer</b> | Size of the layer in bytes as defined by the underlying store. |

### 5.1.27. `.status.images[].manifests[].dockerImageManifests`

#### Description

DockerImageManifests holds information about sub-manifests when the image represents a manifest list. When this field is present, no DockerImageLayers should be specified.

#### Type

**array**

### 5.1.28. `.status.images[].manifests[].dockerImageManifests[]`

#### Description

ImageManifest represents sub-manifests of a manifest list. The Digest field points to a regular Image object.

#### Type

**object**

#### Required

- **digest**
- **mediaType**
- **manifestSize**
- **architecture**
- **os**

| Property            | Type          | Description                                                                                         |
|---------------------|---------------|-----------------------------------------------------------------------------------------------------|
| <b>architecture</b> | <b>string</b> | Architecture specifies the supported CPU architecture, for example <b>amd64</b> or <b>ppc64le</b> . |
| <b>digest</b>       | <b>string</b> | Digest is the unique identifier for the manifest. It refers to an Image object.                     |

| Property            | Type           | Description                                                                                                                                                                                                               |
|---------------------|----------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>manifestSize</b> | <b>integer</b> | ManifestSize represents the size of the raw object contents, in bytes.                                                                                                                                                    |
| <b>mediaType</b>    | <b>string</b>  | MediaType defines the type of the manifest, possible values are application/vnd.oci.image.manifest.v1+json, application/vnd.docker.distribution.manifest.v2+json or application/vnd.docker.distribution.manifest.v1+json. |
| <b>os</b>           | <b>string</b>  | OS specifies the operating system, for example <b>linux</b> .                                                                                                                                                             |
| <b>variant</b>      | <b>string</b>  | Variant is an optional field representing a variant of the CPU, for example v6 to specify a particular CPU variant of the ARM CPU.                                                                                        |

### 5.1.29. .status.images[].manifests[].signatures

#### Description

Signatures holds all signatures of the image.

#### Type

**array**

### 5.1.30. .status.images[].manifests[].signatures[]

#### Description

ImageSignature holds a signature of an image. It allows to verify image identity and possibly other claims as long as the signature is trusted. Based on this information it is possible to restrict runnable images to those matching cluster-wide policy. Mandatory fields should be parsed by clients doing image verification. The others are parsed from signature's content by the server. They serve just an informative purpose.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

#### Type

**object**

#### Required

- **type**
- **content**

| Property             | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                   |
|----------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>apiVersion</b>    | <b>string</b> | APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources</a> |
| <b>conditions</b>    | <b>array</b>  | Conditions represent the latest available observations of a signature's current state.                                                                                                                                                                                                                                                                                                                        |
| <b>conditions[]</b>  | <b>object</b> | SignatureCondition describes an image signature condition of particular kind at particular probe time.                                                                                                                                                                                                                                                                                                        |
| <b>content</b>       | <b>string</b> | Required: An opaque binary string which is an image's signature.                                                                                                                                                                                                                                                                                                                                              |
| <b>created</b>       | <b>Time</b>   | If specified, it is the time of signature's creation.                                                                                                                                                                                                                                                                                                                                                         |
| <b>imageIdentity</b> | <b>string</b> | A human readable string representing image's identity. It could be a product name and version, or an image pull spec (e.g. "registry.access.redhat.com/rhel7/rhel:7.2").                                                                                                                                                                                                                                      |
| <b>issuedBy</b>      | <b>object</b> | SignatureIssuer holds information about an issuer of signing certificate or key.                                                                                                                                                                                                                                                                                                                              |
| <b>issuedTo</b>      | <b>object</b> | SignatureSubject holds information about a person or entity who created the signature.                                                                                                                                                                                                                                                                                                                        |

| Property            | Type                   | Description                                                                                                                                                                                                                                                                                                                                                                                                      |
|---------------------|------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>kind</b>         | <b>string</b>          | Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds</a> |
| <b>metadata</b>     | <b>ObjectMeta_v2</b>   | metadata is the standard object's metadata. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata</a>                                                                                                                                                   |
| <b>signedClaims</b> | <b>object (string)</b> | Contains claims from the signature.                                                                                                                                                                                                                                                                                                                                                                              |
| <b>type</b>         | <b>string</b>          | Required: Describes a type of stored blob.                                                                                                                                                                                                                                                                                                                                                                       |

### 5.1.31. .status.images[].manifests[].signatures[].conditions

#### Description

Conditions represent the latest available observations of a signature's current state.

#### Type

**array**

### 5.1.32. .status.images[].manifests[].signatures[].conditions[]

#### Description

SignatureCondition describes an image signature condition of particular kind at particular probe time.

#### Type

**object**

#### Required

- **type**
- **status**

| Property                  | Type          | Description                                                      |
|---------------------------|---------------|------------------------------------------------------------------|
| <b>lastProbeTime</b>      | <b>Time</b>   | Last time the condition was checked.                             |
| <b>lastTransitionTime</b> | <b>Time</b>   | Last time the condition transit from one status to another.      |
| <b>message</b>            | <b>string</b> | Human readable message indicating details about last transition. |
| <b>reason</b>             | <b>string</b> | (brief) reason for the condition's last transition.              |
| <b>status</b>             | <b>string</b> | Status of the condition, one of True, False, Unknown.            |
| <b>type</b>               | <b>string</b> | Type of signature condition, Complete or Failed.                 |

### 5.1.33. .status.images[].manifests[].signatures[].issuedBy

#### Description

SignatureIssuer holds information about an issuer of signing certificate or key.

#### Type

**object**

| Property            | Type          | Description                                   |
|---------------------|---------------|-----------------------------------------------|
| <b>commonName</b>   | <b>string</b> | Common name (e.g. openshift-signing-service). |
| <b>organization</b> | <b>string</b> | Organization name.                            |

### 5.1.34. .status.images[].manifests[].signatures[].issuedTo

#### Description

SignatureSubject holds information about a person or entity who created the signature.

#### Type

**object**

#### Required

- **publicKeyID**

| Property            | Type          | Description                                                                                                                                                                                                       |
|---------------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>commonName</b>   | <b>string</b> | Common name (e.g. openshift-signing-service).                                                                                                                                                                     |
| <b>organization</b> | <b>string</b> | Organization name.                                                                                                                                                                                                |
| <b>publicKeyID</b>  | <b>string</b> | If present, it is a human readable key id of public key belonging to the subject used to verify image signature. It should contain at least 64 lowest bits of public key's fingerprint (e.g. 0x685ebe62bf278440). |

### 5.1.35. .status.import

#### Description

An ImageStream stores a mapping of tags to images, metadata overrides that are applied when images are tagged in a stream, and an optional reference to a container image repository on a registry. Users typically update the spec.tags field to point to external images which are imported from container registries using credentials in your namespace with the pull secret type, or to existing image stream tags and images which are immediately accessible for tagging or pulling. The history of images applied to a tag is visible in the status.tags field and any user who can view an image stream is allowed to tag that image into their own image streams. Access to pull images from the integrated registry is granted by having the "get imagestreams/layers" permission on a given image stream. Users may remove a tag by deleting the imagestreamtag resource, which causes both spec and status for that tag to be removed. Image stream history is retained until an administrator runs the prune operation, which removes references that are no longer in use. To preserve a historical image, ensure there is a tag in spec pointing to that image by its digest.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

#### Type

**object**

| Property          | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                   |
|-------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>apiVersion</b> | <b>string</b> | APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources</a> |

| Property        | Type                 | Description                                                                                                                                                                                                                                                                                                                                                                                                      |
|-----------------|----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>kind</b>     | <b>string</b>        | Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds</a> |
| <b>metadata</b> | <b>ObjectMeta_v2</b> | metadata is the standard object's metadata. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata</a>                                                                                                                                                   |
| <b>spec</b>     | <b>object</b>        | ImageStreamSpec represents options for ImageStreams.                                                                                                                                                                                                                                                                                                                                                             |
| <b>status</b>   | <b>object</b>        | ImageStreamStatus contains information about the state of this image stream.                                                                                                                                                                                                                                                                                                                                     |

### 5.1.36. .status.import.spec

#### Description

ImageStreamSpec represents options for ImageStreams.

#### Type

**object**

| Property                     | Type          | Description                                                                                                                                                                                                                                                                                                  |
|------------------------------|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dockerImageRepository</b> | <b>string</b> | dockerImageRepository is optional, if specified this stream is backed by a container repository on this server Deprecated: This field is deprecated as of v3.7 and will be removed in a future release. Specify the source for the tags to be imported in each tag via the spec.tags.from reference instead. |

| Property            | Type          | Description                                                                                                                                                                   |
|---------------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>lookupPolicy</b> | <b>object</b> | ImageLookupPolicy describes how an image stream can be used to override the image references used by pods, builds, and other resources in a namespace.                        |
| <b>tags</b>         | <b>array</b>  | tags map arbitrary string values to specific image locators                                                                                                                   |
| <b>tags[]</b>       | <b>object</b> | TagReference specifies optional annotations for images using this tag and an optional reference to an ImageStreamTag, ImageStreamImage, or DockerImage this tag should track. |

### 5.1.37. .status.import.spec.lookupPolicy

#### Description

ImageLookupPolicy describes how an image stream can be used to override the image references used by pods, builds, and other resources in a namespace.

#### Type

**object**

#### Required

- **local**

| Property     | Type           | Description                                                                                                                                                                                                                                                                                                                                                                                    |
|--------------|----------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>local</b> | <b>boolean</b> | local will change the docker short image references (like "mysql" or "php:latest") on objects in this namespace to the image ID whenever they match this image stream, instead of reaching out to a remote registry. The name will be fully qualified to an image ID if found. The tag's referencePolicy is taken into account on the replaced value. Only works within the current namespace. |

### 5.1.38. .status.import.spec.tags

#### Description

tags map arbitrary string values to specific image locators

Type

**array**

### 5.1.39. .status.import.spec.tags[]

Description

TagReference specifies optional annotations for images using this tag and an optional reference to an ImageStreamTag, ImageStreamImage, or DockerImage this tag should track.

Type

**object**

Required

- **name**

| Property           | Type                   | Description                                                                                                                                                                                                                              |
|--------------------|------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>annotations</b> | <b>object (string)</b> | Optional; if specified, annotations that are applied to images retrieved via ImageStreamTags.                                                                                                                                            |
| <b>from</b>        | <b>ObjectReference</b> | Optional; if specified, a reference to another image that this tag should point to. Valid values are ImageStreamTag, ImageStreamImage, and DockerImage. ImageStreamTag references can only reference a tag within this same ImageStream. |

| Property               | Type           | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|------------------------|----------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>generation</b>      | <b>integer</b> | Generation is a counter that tracks mutations to the spec tag (user intent). When a tag reference is changed the generation is set to match the current stream generation (which is incremented every time spec is changed). Other processes in the system like the image importer observe that the generation of spec tag is newer than the generation recorded in the status and use that as a trigger to import the newest remote tag. To trigger a new import, clients may set this value to zero which will reset the generation to the latest stream generation. Legacy clients will send this value as nil which will be merged with the current tag generation. |
| <b>importPolicy</b>    | <b>object</b>  | TagImportPolicy controls how images related to this tag will be imported.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| <b>name</b>            | <b>string</b>  | Name of the tag                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| <b>reference</b>       | <b>boolean</b> | Reference states if the tag will be imported. Default value is false, which means the tag will be imported.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| <b>referencePolicy</b> | <b>object</b>  | TagReferencePolicy describes how pull-specs for images in this image stream tag are generated when image change triggers in deployment configs or builds are resolved. This allows the image stream author to control how images are accessed.                                                                                                                                                                                                                                                                                                                                                                                                                          |

#### 5.1.40. `.status.import.spec.tags[].importPolicy`

##### Description

TagImportPolicy controls how images related to this tag will be imported.

##### Type

**object**

| Property          | Type           | Description                                                                                                             |
|-------------------|----------------|-------------------------------------------------------------------------------------------------------------------------|
| <b>importMode</b> | <b>string</b>  | ImportMode describes how to import an image manifest.                                                                   |
| <b>insecure</b>   | <b>boolean</b> | Insecure is true if the server may bypass certificate verification or connect directly over HTTP during image import.   |
| <b>scheduled</b>  | <b>boolean</b> | Scheduled indicates to the server that this tag should be periodically checked to ensure it is up to date, and imported |

#### 5.1.41. `.status.import.spec.tags[].referencePolicy`

##### Description

TagReferencePolicy describes how pull-specs for images in this image stream tag are generated when image change triggers in deployment configs or builds are resolved. This allows the image stream author to control how images are accessed.

##### Type

**object**

##### Required

- **type**

| Property | Type | Description |
|----------|------|-------------|
|----------|------|-------------|

| Property    | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|-------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>type</b> | <b>string</b> | Type determines how the image pull spec should be transformed when the image stream tag is used in deployment config triggers or new builds. The default value is <b>Source</b> , indicating the original location of the image should be used (if imported). The user may also specify <b>Local</b> , indicating that the pull spec should point to the integrated container image registry and leverage the registry's ability to proxy the pull to an upstream registry. <b>Local</b> allows the credentials used to pull this image to be managed from the image stream's namespace, so others on the platform can access a remote image but have no access to the remote secret. It also allows the image layers to be mirrored into the local registry which the images can still be pulled even if the upstream registry is unavailable. |

### 5.1.42. .status.import.status

#### Description

ImageStreamStatus contains information about the state of this image stream.

#### Type

**object**

#### Required

- **dockerImageRepository**

| Property                     | Type          | Description                                                                                                                                                      |
|------------------------------|---------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dockerImageRepository</b> | <b>string</b> | DockerImageRepository represents the effective location this stream may be accessed at. May be empty until the server determines where the repository is located |

| Property                           | Type          | Description                                                                                                                                                                                                         |
|------------------------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>publicDockerImageRepository</b> | <b>string</b> | PublicDockerImageRepository represents the public location from where the image can be pulled outside the cluster. This field may be empty if the administrator has not exposed the integrated registry externally. |
| <b>tags</b>                        | <b>array</b>  | Tags are a historical record of images associated with each tag. The first entry in the TagEvent array is the currently tagged image.                                                                               |
| <b>tags[]</b>                      | <b>object</b> | NamedTagEventList relates a tag to its image history.                                                                                                                                                               |

### 5.1.43. .status.import.status.tags

#### Description

Tags are a historical record of images associated with each tag. The first entry in the TagEvent array is the currently tagged image.

#### Type

**array**

### 5.1.44. .status.import.status.tags[]

#### Description

NamedTagEventList relates a tag to its image history.

#### Type

**object**

#### Required

- **tag**
- **items**

| Property          | Type         | Description                                                            |
|-------------------|--------------|------------------------------------------------------------------------|
| <b>conditions</b> | <b>array</b> | Conditions is an array of conditions that apply to the tag event list. |

| Property            | Type          | Description                                                                                        |
|---------------------|---------------|----------------------------------------------------------------------------------------------------|
| <b>conditions[]</b> | <b>object</b> | TagEventCondition contains condition information for a tag event.                                  |
| <b>items</b>        | <b>array</b>  | Standard object's metadata.                                                                        |
| <b>items[]</b>      | <b>object</b> | TagEvent is used by ImageStreamStatus to keep a historical record of images associated with a tag. |
| <b>tag</b>          | <b>string</b> | Tag is the tag for which the history is recorded                                                   |

#### 5.1.45. `.status.import.status.tags[].conditions`

##### Description

Conditions is an array of conditions that apply to the tag event list.

##### Type

**array**

#### 5.1.46. `.status.import.status.tags[].conditions[]`

##### Description

TagEventCondition contains condition information for a tag event.

##### Type

**object**

##### Required

- **type**
- **status**
- **generation**

| Property          | Type           | Description                                                           |
|-------------------|----------------|-----------------------------------------------------------------------|
| <b>generation</b> | <b>integer</b> | Generation is the spec tag generation that this status corresponds to |

| Property                  | Type          | Description                                                                                         |
|---------------------------|---------------|-----------------------------------------------------------------------------------------------------|
| <b>lastTransitionTime</b> | <b>Time</b>   | LastTransitionTime is the time the condition transitioned from one status to another.               |
| <b>message</b>            | <b>string</b> | Message is a human readable description of the details about last transition, complementing reason. |
| <b>reason</b>             | <b>string</b> | Reason is a brief machine readable explanation for the condition's last transition.                 |
| <b>status</b>             | <b>string</b> | Status of the condition, one of True, False, Unknown.                                               |
| <b>type</b>               | <b>string</b> | Type of tag event condition, currently only ImportSuccess                                           |

#### 5.1.47. `.status.import.status.tags[].items`

##### Description

Standard object's metadata.

##### Type

**array**

#### 5.1.48. `.status.import.status.tags[].items[]`

##### Description

TagEvent is used by ImageStreamStatus to keep a historical record of images associated with a tag.

##### Type

**object**

##### Required

- **created**
- **dockerImageReference**
- **image**
- **generation**

| Property | Type | Description |
|----------|------|-------------|
|----------|------|-------------|

| Property                    | Type           | Description                                                                   |
|-----------------------------|----------------|-------------------------------------------------------------------------------|
| <b>created</b>              | <b>Time</b>    | Created holds the time the TagEvent was created                               |
| <b>dockerImageReference</b> | <b>string</b>  | DockerImageReference is the string that can be used to pull this image        |
| <b>generation</b>           | <b>integer</b> | Generation is the spec tag generation that resulted in this tag being updated |
| <b>image</b>                | <b>string</b>  | Image is the image                                                            |

### 5.1.49. .status.repository

#### Description

RepositoryImportStatus describes the result of an image repository import

#### Type

**object**

| Property              | Type                  | Description                                                                                                                          |
|-----------------------|-----------------------|--------------------------------------------------------------------------------------------------------------------------------------|
| <b>additionalTags</b> | <b>array (string)</b> | AdditionalTags are tags that exist in the repository but were not imported because a maximum limit of automatic imports was applied. |
| <b>images</b>         | <b>array</b>          | Images is a list of images successfully retrieved by the import of the repository.                                                   |
| <b>images[]</b>       | <b>object</b>         | ImageImportStatus describes the result of an image import.                                                                           |
| <b>status</b>         | <b>Status_v5</b>      | Status reflects whether any failure occurred during import                                                                           |

### 5.1.50. .status.repository.images

#### Description

Images is a list of images successfully retrieved by the import of the repository.

#### Type

**array**

### 5.1.51. .status.repository.images[]

#### Description

ImageImportStatus describes the result of an image import.

#### Type

**object**

#### Required

- **status**

| Property         | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|------------------|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>image</b>     | <b>object</b> | <p>Image is an immutable representation of a container image and metadata at a point in time. Images are named by taking a hash of their contents (metadata and content) and any change in format, content, or metadata results in a new name. The images resource is primarily for use by cluster administrators and integrations like the cluster image registry - end users instead access images via the imagestreamtags or imagestreamimages resources. While image metadata is stored in the API, any integration that implements the container image registry API must provide its own storage for the raw manifest data, image config, and layer contents.</p> <p>Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).</p> |
| <b>manifests</b> | <b>array</b>  | Manifests holds sub-manifests metadata when importing a manifest list                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |

| Property           | Type             | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|--------------------|------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>manifests[]</b> | <b>object</b>    | <p>Image is an immutable representation of a container image and metadata at a point in time. Images are named by taking a hash of their contents (metadata and content) and any change in format, content, or metadata results in a new name. The images resource is primarily for use by cluster administrators and integrations like the cluster image registry - end users instead access images via the imagestreamtags or imagestreamimages resources. While image metadata is stored in the API, any integration that implements the container image registry API must provide its own storage for the raw manifest data, image config, and layer contents.</p> <p>Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).</p> |
| <b>status</b>      | <b>Status_v5</b> | Status is the status of the image import, including errors encountered while retrieving the image                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| <b>tag</b>         | <b>string</b>    | Tag is the tag this image was located under, if any                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |

### 5.1.52. .status.repository.images[].image

#### Description

Image is an immutable representation of a container image and metadata at a point in time. Images are named by taking a hash of their contents (metadata and content) and any change in format, content, or metadata results in a new name. The images resource is primarily for use by cluster administrators and integrations like the cluster image registry - end users instead access images via the imagestreamtags or imagestreamimages resources. While image metadata is stored in the API, any integration that implements the container image registry API must provide its own storage for the raw manifest data, image config, and layer contents.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

#### Type

**object**

| Property                            | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                   |
|-------------------------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>apiVersion</b>                   | <b>string</b> | APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources</a> |
| <b>dockerImageConfig</b>            | <b>string</b> | DockerImageConfig is a JSON blob that the runtime uses to set up the container. This is a part of manifest schema v2. Will not be set when the image represents a manifest list.                                                                                                                                                                                                                              |
| <b>dockerImageLayers</b>            | <b>array</b>  | DockerImageLayers represents the layers in the image. May not be set if the image does not define that data or if the image represents a manifest list.                                                                                                                                                                                                                                                       |
| <b>dockerImageLayers[]</b>          | <b>object</b> | ImageLayer represents a single layer of the image. Some images may have multiple layers. Some may have none.                                                                                                                                                                                                                                                                                                  |
| <b>dockerImageManifest</b>          | <b>string</b> | DockerImageManifest is the raw JSON of the manifest                                                                                                                                                                                                                                                                                                                                                           |
| <b>dockerImageManifestMediaType</b> | <b>string</b> | DockerImageManifestMediaType specifies the mediaType of manifest. This is a part of manifest schema v2.                                                                                                                                                                                                                                                                                                       |
| <b>dockerImageManifests</b>         | <b>array</b>  | DockerImageManifests holds information about sub-manifests when the image represents a manifest list. When this field is present, no DockerImageLayers should be specified.                                                                                                                                                                                                                                   |
| <b>dockerImageManifests[]</b>       | <b>object</b> | ImageManifest represents sub-manifests of a manifest list. The Digest field points to a regular Image object.                                                                                                                                                                                                                                                                                                 |

| Property                          | Type                          | Description                                                                                                                                                                                                                                                                                                                                                                                                      |
|-----------------------------------|-------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dockerImageMetadata</b>        | <a href="#">RawExtension</a>  | DockerImageMetadata contains metadata about this image                                                                                                                                                                                                                                                                                                                                                           |
| <b>dockerImageMetadataVersion</b> | <b>string</b>                 | DockerImageMetadataVersion conveys the version of the object, which if empty defaults to "1.0"                                                                                                                                                                                                                                                                                                                   |
| <b>dockerImageReference</b>       | <b>string</b>                 | DockerImageReference is the string that can be used to pull this image.                                                                                                                                                                                                                                                                                                                                          |
| <b>dockerImageSignatures</b>      | <b>array (string)</b>         | DockerImageSignatures provides the signatures as opaque blobs. This is a part of manifest schema v1.                                                                                                                                                                                                                                                                                                             |
| <b>kind</b>                       | <b>string</b>                 | Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds</a> |
| <b>metadata</b>                   | <a href="#">ObjectMeta_v2</a> | metadata is the standard object's metadata. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata</a>                                                                                                                                                   |
| <b>signatures</b>                 | <b>array</b>                  | Signatures holds all signatures of the image.                                                                                                                                                                                                                                                                                                                                                                    |

| Property            | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|---------------------|---------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>signatures[]</b> | <b>object</b> | <p>ImageSignature holds a signature of an image. It allows to verify image identity and possibly other claims as long as the signature is trusted. Based on this information it is possible to restrict runnable images to those matching cluster-wide policy. Mandatory fields should be parsed by clients doing image verification. The others are parsed from signature's content by the server. They serve just an informative purpose.</p> <p>Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).</p> |

### 5.1.53. .status.repository.images[].image.dockerImageLayers

#### Description

DockerImageLayers represents the layers in the image. May not be set if the image does not define that data or if the image represents a manifest list.

#### Type

**array**

### 5.1.54. .status.repository.images[].image.dockerImageLayers[]

#### Description

ImageLayer represents a single layer of the image. Some images may have multiple layers. Some may have none.

#### Type

**object**

#### Required

- **name**
- **size**
- **mediaType**

| Property         | Type          | Description                         |
|------------------|---------------|-------------------------------------|
| <b>mediaType</b> | <b>string</b> | MediaType of the referenced object. |

| Property    | Type           | Description                                                    |
|-------------|----------------|----------------------------------------------------------------|
| <b>name</b> | <b>string</b>  | Name of the layer as defined by the underlying store.          |
| <b>size</b> | <b>integer</b> | Size of the layer in bytes as defined by the underlying store. |

### 5.1.55. `.status.repository.images[].image.dockerImageManifests`

#### Description

DockerImageManifests holds information about sub-manifests when the image represents a manifest list. When this field is present, no DockerImageLayers should be specified.

#### Type

**array**

### 5.1.56. `.status.repository.images[].image.dockerImageManifests[]`

#### Description

ImageManifest represents sub-manifests of a manifest list. The Digest field points to a regular Image object.

#### Type

**object**

#### Required

- **digest**
- **mediaType**
- **manifestSize**
- **architecture**
- **os**

| Property            | Type          | Description                                                                                         |
|---------------------|---------------|-----------------------------------------------------------------------------------------------------|
| <b>architecture</b> | <b>string</b> | Architecture specifies the supported CPU architecture, for example <b>amd64</b> or <b>ppc64le</b> . |
| <b>digest</b>       | <b>string</b> | Digest is the unique identifier for the manifest. It refers to an Image object.                     |

| Property            | Type           | Description                                                                                                                                                                                                               |
|---------------------|----------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>manifestSize</b> | <b>integer</b> | ManifestSize represents the size of the raw object contents, in bytes.                                                                                                                                                    |
| <b>mediaType</b>    | <b>string</b>  | MediaType defines the type of the manifest, possible values are application/vnd.oci.image.manifest.v1+json, application/vnd.docker.distribution.manifest.v2+json or application/vnd.docker.distribution.manifest.v1+json. |
| <b>os</b>           | <b>string</b>  | OS specifies the operating system, for example <b>linux</b> .                                                                                                                                                             |
| <b>variant</b>      | <b>string</b>  | Variant is an optional field representing a variant of the CPU, for example v6 to specify a particular CPU variant of the ARM CPU.                                                                                        |

### 5.1.57. .status.repository.images[].image.signatures

#### Description

Signatures holds all signatures of the image.

#### Type

**array**

### 5.1.58. .status.repository.images[].image.signatures[]

#### Description

ImageSignature holds a signature of an image. It allows to verify image identity and possibly other claims as long as the signature is trusted. Based on this information it is possible to restrict runnable images to those matching cluster-wide policy. Mandatory fields should be parsed by clients doing image verification. The others are parsed from signature's content by the server. They serve just an informative purpose.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

#### Type

**object**

#### Required

- **type**
- **content**

| Property             | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                   |
|----------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>apiVersion</b>    | <b>string</b> | APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources</a> |
| <b>conditions</b>    | <b>array</b>  | Conditions represent the latest available observations of a signature's current state.                                                                                                                                                                                                                                                                                                                        |
| <b>conditions[]</b>  | <b>object</b> | SignatureCondition describes an image signature condition of particular kind at particular probe time.                                                                                                                                                                                                                                                                                                        |
| <b>content</b>       | <b>string</b> | Required: An opaque binary string which is an image's signature.                                                                                                                                                                                                                                                                                                                                              |
| <b>created</b>       | <b>Time</b>   | If specified, it is the time of signature's creation.                                                                                                                                                                                                                                                                                                                                                         |
| <b>imageIdentity</b> | <b>string</b> | A human readable string representing image's identity. It could be a product name and version, or an image pull spec (e.g. "registry.access.redhat.com/rhel7/rhel:7.2").                                                                                                                                                                                                                                      |
| <b>issuedBy</b>      | <b>object</b> | SignatureIssuer holds information about an issuer of signing certificate or key.                                                                                                                                                                                                                                                                                                                              |
| <b>issuedTo</b>      | <b>object</b> | SignatureSubject holds information about a person or entity who created the signature.                                                                                                                                                                                                                                                                                                                        |

| Property            | Type                   | Description                                                                                                                                                                                                                                                                                                                                                                                                      |
|---------------------|------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>kind</b>         | <b>string</b>          | Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds</a> |
| <b>metadata</b>     | <b>ObjectMeta_v2</b>   | metadata is the standard object's metadata. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata</a>                                                                                                                                                   |
| <b>signedClaims</b> | <b>object (string)</b> | Contains claims from the signature.                                                                                                                                                                                                                                                                                                                                                                              |
| <b>type</b>         | <b>string</b>          | Required: Describes a type of stored blob.                                                                                                                                                                                                                                                                                                                                                                       |

### 5.1.59. .status.repository.images[].image.signatures[].conditions

#### Description

Conditions represent the latest available observations of a signature's current state.

#### Type

**array**

### 5.1.60. .status.repository.images[].image.signatures[].conditions[]

#### Description

SignatureCondition describes an image signature condition of particular kind at particular probe time.

#### Type

**object**

#### Required

- **type**
- **status**

| Property                  | Type          | Description                                                      |
|---------------------------|---------------|------------------------------------------------------------------|
| <b>lastProbeTime</b>      | <b>Time</b>   | Last time the condition was checked.                             |
| <b>lastTransitionTime</b> | <b>Time</b>   | Last time the condition transit from one status to another.      |
| <b>message</b>            | <b>string</b> | Human readable message indicating details about last transition. |
| <b>reason</b>             | <b>string</b> | (brief) reason for the condition's last transition.              |
| <b>status</b>             | <b>string</b> | Status of the condition, one of True, False, Unknown.            |
| <b>type</b>               | <b>string</b> | Type of signature condition, Complete or Failed.                 |

### 5.1.61. `.status.repository.images[].image.signatures[].issuedBy`

#### Description

SignatureIssuer holds information about an issuer of signing certificate or key.

#### Type

**object**

| Property            | Type          | Description                                   |
|---------------------|---------------|-----------------------------------------------|
| <b>commonName</b>   | <b>string</b> | Common name (e.g. openshift-signing-service). |
| <b>organization</b> | <b>string</b> | Organization name.                            |

### 5.1.62. `.status.repository.images[].image.signatures[].issuedTo`

#### Description

SignatureSubject holds information about a person or entity who created the signature.

#### Type

**object**

#### Required

- **publicKeyID**

| Property            | Type          | Description                                                                                                                                                                                                       |
|---------------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>commonName</b>   | <b>string</b> | Common name (e.g. openshift-signing-service).                                                                                                                                                                     |
| <b>organization</b> | <b>string</b> | Organization name.                                                                                                                                                                                                |
| <b>publicKeyID</b>  | <b>string</b> | If present, it is a human readable key id of public key belonging to the subject used to verify image signature. It should contain at least 64 lowest bits of public key's fingerprint (e.g. 0x685ebe62bf278440). |

### 5.1.63. .status.repository.images[].manifests

#### Description

Manifests holds sub-manifests metadata when importing a manifest list

#### Type

**array**

### 5.1.64. .status.repository.images[].manifests[]

#### Description

Image is an immutable representation of a container image and metadata at a point in time. Images are named by taking a hash of their contents (metadata and content) and any change in format, content, or metadata results in a new name. The images resource is primarily for use by cluster administrators and integrations like the cluster image registry - end users instead access images via the imagestreamtags or imagestreamimages resources. While image metadata is stored in the API, any integration that implements the container image registry API must provide its own storage for the raw manifest data, image config, and layer contents.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

#### Type

**object**

| Property | Type | Description |
|----------|------|-------------|
|----------|------|-------------|

| Property                            | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                   |
|-------------------------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>apiVersion</b>                   | <b>string</b> | APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources</a> |
| <b>dockerImageConfig</b>            | <b>string</b> | DockerImageConfig is a JSON blob that the runtime uses to set up the container. This is a part of manifest schema v2. Will not be set when the image represents a manifest list.                                                                                                                                                                                                                              |
| <b>dockerImageLayers</b>            | <b>array</b>  | DockerImageLayers represents the layers in the image. May not be set if the image does not define that data or if the image represents a manifest list.                                                                                                                                                                                                                                                       |
| <b>dockerImageLayers[]</b>          | <b>object</b> | ImageLayer represents a single layer of the image. Some images may have multiple layers. Some may have none.                                                                                                                                                                                                                                                                                                  |
| <b>dockerImageManifest</b>          | <b>string</b> | DockerImageManifest is the raw JSON of the manifest                                                                                                                                                                                                                                                                                                                                                           |
| <b>dockerImageManifestMediaType</b> | <b>string</b> | DockerImageManifestMediaType specifies the mediaType of manifest. This is a part of manifest schema v2.                                                                                                                                                                                                                                                                                                       |
| <b>dockerImageManifests</b>         | <b>array</b>  | DockerImageManifests holds information about sub-manifests when the image represents a manifest list. When this field is present, no DockerImageLayers should be specified.                                                                                                                                                                                                                                   |
| <b>dockerImageManifests[]</b>       | <b>object</b> | ImageManifest represents sub-manifests of a manifest list. The Digest field points to a regular Image object.                                                                                                                                                                                                                                                                                                 |

| Property                          | Type                  | Description                                                                                                                                                                                                                                                                                                                                                                                                      |
|-----------------------------------|-----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dockerImageMetadata</b>        | <b>RawExtension</b>   | DockerImageMetadata contains metadata about this image                                                                                                                                                                                                                                                                                                                                                           |
| <b>dockerImageMetadataVersion</b> | <b>string</b>         | DockerImageMetadataVersion conveys the version of the object, which if empty defaults to "1.0"                                                                                                                                                                                                                                                                                                                   |
| <b>dockerImageReference</b>       | <b>string</b>         | DockerImageReference is the string that can be used to pull this image.                                                                                                                                                                                                                                                                                                                                          |
| <b>dockerImageSignatures</b>      | <b>array (string)</b> | DockerImageSignatures provides the signatures as opaque blobs. This is a part of manifest schema v1.                                                                                                                                                                                                                                                                                                             |
| <b>kind</b>                       | <b>string</b>         | Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds</a> |
| <b>metadata</b>                   | <b>ObjectMeta_v2</b>  | metadata is the standard object's metadata. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata</a>                                                                                                                                                   |
| <b>signatures</b>                 | <b>array</b>          | Signatures holds all signatures of the image.                                                                                                                                                                                                                                                                                                                                                                    |

| Property            | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|---------------------|---------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>signatures[]</b> | <b>object</b> | <p>ImageSignature holds a signature of an image. It allows to verify image identity and possibly other claims as long as the signature is trusted. Based on this information it is possible to restrict runnable images to those matching cluster-wide policy. Mandatory fields should be parsed by clients doing image verification. The others are parsed from signature's content by the server. They serve just an informative purpose.</p> <p>Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).</p> |

### 5.1.65. `.status.repository.images[].manifests[].dockerImageLayers`

#### Description

DockerImageLayers represents the layers in the image. May not be set if the image does not define that data or if the image represents a manifest list.

#### Type

**array**

### 5.1.66. `.status.repository.images[].manifests[].dockerImageLayers[]`

#### Description

ImageLayer represents a single layer of the image. Some images may have multiple layers. Some may have none.

#### Type

**object**

#### Required

- **name**
- **size**
- **mediaType**

| Property         | Type          | Description                         |
|------------------|---------------|-------------------------------------|
| <b>mediaType</b> | <b>string</b> | MediaType of the referenced object. |

| Property    | Type           | Description                                                    |
|-------------|----------------|----------------------------------------------------------------|
| <b>name</b> | <b>string</b>  | Name of the layer as defined by the underlying store.          |
| <b>size</b> | <b>integer</b> | Size of the layer in bytes as defined by the underlying store. |

### 5.1.67. `.status.repository.images[].manifests[].dockerImageManifests`

#### Description

DockerImageManifests holds information about sub-manifests when the image represents a manifest list. When this field is present, no DockerImageLayers should be specified.

#### Type

**array**

### 5.1.68. `.status.repository.images[].manifests[].dockerImageManifests[]`

#### Description

ImageManifest represents sub-manifests of a manifest list. The Digest field points to a regular Image object.

#### Type

**object**

#### Required

- **digest**
- **mediaType**
- **manifestSize**
- **architecture**
- **os**

| Property            | Type          | Description                                                                                         |
|---------------------|---------------|-----------------------------------------------------------------------------------------------------|
| <b>architecture</b> | <b>string</b> | Architecture specifies the supported CPU architecture, for example <b>amd64</b> or <b>ppc64le</b> . |
| <b>digest</b>       | <b>string</b> | Digest is the unique identifier for the manifest. It refers to an Image object.                     |

| Property            | Type           | Description                                                                                                                                                                                                               |
|---------------------|----------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>manifestSize</b> | <b>integer</b> | ManifestSize represents the size of the raw object contents, in bytes.                                                                                                                                                    |
| <b>mediaType</b>    | <b>string</b>  | MediaType defines the type of the manifest, possible values are application/vnd.oci.image.manifest.v1+json, application/vnd.docker.distribution.manifest.v2+json or application/vnd.docker.distribution.manifest.v1+json. |
| <b>os</b>           | <b>string</b>  | OS specifies the operating system, for example <b>linux</b> .                                                                                                                                                             |
| <b>variant</b>      | <b>string</b>  | Variant is an optional field representing a variant of the CPU, for example v6 to specify a particular CPU variant of the ARM CPU.                                                                                        |

### 5.1.69. .status.repository.images[].manifests[].signatures

#### Description

Signatures holds all signatures of the image.

#### Type

**array**

### 5.1.70. .status.repository.images[].manifests[].signatures[]

#### Description

ImageSignature holds a signature of an image. It allows to verify image identity and possibly other claims as long as the signature is trusted. Based on this information it is possible to restrict runnable images to those matching cluster-wide policy. Mandatory fields should be parsed by clients doing image verification. The others are parsed from signature's content by the server. They serve just an informative purpose.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

#### Type

**object**

#### Required

- **type**
- **content**

| Property             | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                   |
|----------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>apiVersion</b>    | <b>string</b> | APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources</a> |
| <b>conditions</b>    | <b>array</b>  | Conditions represent the latest available observations of a signature's current state.                                                                                                                                                                                                                                                                                                                        |
| <b>conditions[]</b>  | <b>object</b> | SignatureCondition describes an image signature condition of particular kind at particular probe time.                                                                                                                                                                                                                                                                                                        |
| <b>content</b>       | <b>string</b> | Required: An opaque binary string which is an image's signature.                                                                                                                                                                                                                                                                                                                                              |
| <b>created</b>       | <b>Time</b>   | If specified, it is the time of signature's creation.                                                                                                                                                                                                                                                                                                                                                         |
| <b>imageIdentity</b> | <b>string</b> | A human readable string representing image's identity. It could be a product name and version, or an image pull spec (e.g. "registry.access.redhat.com/rhel7/rhel:7.2").                                                                                                                                                                                                                                      |
| <b>issuedBy</b>      | <b>object</b> | SignatureIssuer holds information about an issuer of signing certificate or key.                                                                                                                                                                                                                                                                                                                              |
| <b>issuedTo</b>      | <b>object</b> | SignatureSubject holds information about a person or entity who created the signature.                                                                                                                                                                                                                                                                                                                        |

| Property            | Type                   | Description                                                                                                                                                                                                                                                                                                                                                                                                      |
|---------------------|------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>kind</b>         | <b>string</b>          | Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds</a> |
| <b>metadata</b>     | <b>ObjectMeta_v2</b>   | metadata is the standard object's metadata. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata</a>                                                                                                                                                   |
| <b>signedClaims</b> | <b>object (string)</b> | Contains claims from the signature.                                                                                                                                                                                                                                                                                                                                                                              |
| <b>type</b>         | <b>string</b>          | Required: Describes a type of stored blob.                                                                                                                                                                                                                                                                                                                                                                       |

### 5.1.71. .status.repository.images[].manifests[].signatures[].conditions

#### Description

Conditions represent the latest available observations of a signature's current state.

#### Type

**array**

### 5.1.72. .status.repository.images[].manifests[].signatures[].conditions[]

#### Description

SignatureCondition describes an image signature condition of particular kind at particular probe time.

#### Type

**object**

#### Required

- **type**
- **status**

| Property                  | Type          | Description                                                      |
|---------------------------|---------------|------------------------------------------------------------------|
| <b>lastProbeTime</b>      | <b>Time</b>   | Last time the condition was checked.                             |
| <b>lastTransitionTime</b> | <b>Time</b>   | Last time the condition transit from one status to another.      |
| <b>message</b>            | <b>string</b> | Human readable message indicating details about last transition. |
| <b>reason</b>             | <b>string</b> | (brief) reason for the condition's last transition.              |
| <b>status</b>             | <b>string</b> | Status of the condition, one of True, False, Unknown.            |
| <b>type</b>               | <b>string</b> | Type of signature condition, Complete or Failed.                 |

### 5.1.73. `.status.repository.images[].manifests[].signatures[].issuedBy`

#### Description

SignatureIssuer holds information about an issuer of signing certificate or key.

#### Type

**object**

| Property            | Type          | Description                                   |
|---------------------|---------------|-----------------------------------------------|
| <b>commonName</b>   | <b>string</b> | Common name (e.g. openshift-signing-service). |
| <b>organization</b> | <b>string</b> | Organization name.                            |

### 5.1.74. `.status.repository.images[].manifests[].signatures[].issuedTo`

#### Description

SignatureSubject holds information about a person or entity who created the signature.

#### Type

**object**

#### Required

- **publicKeyID**

| Property            | Type          | Description                                                                                                                                                                                                       |
|---------------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>commonName</b>   | <b>string</b> | Common name (e.g. openshift-signing-service).                                                                                                                                                                     |
| <b>organization</b> | <b>string</b> | Organization name.                                                                                                                                                                                                |
| <b>publicKeyID</b>  | <b>string</b> | If present, it is a human readable key id of public key belonging to the subject used to verify image signature. It should contain at least 64 lowest bits of public key's fingerprint (e.g. 0x685ebe62bf278440). |

## 5.2. API ENDPOINTS

The following API endpoints are available:

- **/apis/image.openshift.io/v1/namespaces/{namespace}/imagestreamimports**
  - **POST**: create an ImageStreamImport

### 5.2.1. /apis/image.openshift.io/v1/namespaces/{namespace}/imagestreamimports

Table 5.1. Global query parameters

| Parameter     | Type          | Description                                                                                                                                                                                                                                              |
|---------------|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dryRun</b> | <b>string</b> | When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed |

| Parameter              | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|------------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>fieldValidation</b> | <b>string</b> | fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: <ul style="list-style-type: none"> <li>- Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23.</li> <li>- Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+</li> <li>- Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.</li> </ul> |

**HTTP method****POST****Description**

create an ImageStreamImport

**Table 5.2. Body parameters**

| Parameter   | Type                               | Description |
|-------------|------------------------------------|-------------|
| <b>body</b> | <b>ImageStreamImport</b><br>schema |             |

**Table 5.3. HTTP responses**

| HTTP code          | Response body                   |
|--------------------|---------------------------------|
| 200 - OK           | <b>ImageStreamImport</b> schema |
| 201 - Created      | <b>ImageStreamImport</b> schema |
| 202 - Accepted     | <b>ImageStreamImport</b> schema |
| 401 - Unauthorized | Empty                           |

## CHAPTER 6. IMAGESTREAMLAYERS

### [IMAGE.OPENSIFT.IO/V1]

#### Description

ImageStreamLayers describes information about the layers referenced by images in this image stream.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

#### Type

**object**

#### Required

- **blobs**
- **images**

## 6.1. SPECIFICATION

| Property          | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                   |
|-------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>apiVersion</b> | <b>string</b> | APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources</a> |
| <b>blobs</b>      | <b>object</b> | blobs is a map of blob name to metadata about the blob.                                                                                                                                                                                                                                                                                                                                                       |
| <b>blobs{}</b>    | <b>object</b> | ImageLayerData contains metadata about an image layer.                                                                                                                                                                                                                                                                                                                                                        |
| <b>images</b>     | <b>object</b> | images is a map between an image name and the names of the blobs and config that comprise the image.                                                                                                                                                                                                                                                                                                          |
| <b>images{}</b>   | <b>object</b> | ImageBlobReferences describes the blob references within an image.                                                                                                                                                                                                                                                                                                                                            |

| Property        | Type                 | Description                                                                                                                                                                                                                                                                                                                                                                                                      |
|-----------------|----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>kind</b>     | <b>string</b>        | Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds</a> |
| <b>metadata</b> | <b>ObjectMeta_v2</b> | metadata is the standard object's metadata. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata</a>                                                                                                                                                   |

### 6.1.1. .blobs

#### Description

`blobs` is a map of blob name to metadata about the blob.

#### Type

**object**

### 6.1.2. .blobs{}

#### Description

`ImageLayerData` contains metadata about an image layer.

#### Type

**object**

#### Required

- **size**
- **mediaType**

| Property         | Type          | Description                         |
|------------------|---------------|-------------------------------------|
| <b>mediaType</b> | <b>string</b> | MediaType of the referenced object. |

| Property    | Type           | Description                                                                                                                                     |
|-------------|----------------|-------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>size</b> | <b>integer</b> | Size of the layer in bytes as defined by the underlying store. This field is optional if the necessary information about size is not available. |

### 6.1.3. .images

#### Description

`images` is a map between an image name and the names of the blobs and config that comprise the image.

#### Type

**object**

### 6.1.4. .images{}

#### Description

`ImageBlobReferences` describes the blob references within an image.

#### Type

**object**

| Property            | Type           | Description                                                                                                                                                                                                                                                                                                                               |
|---------------------|----------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>config</b>       | <b>string</b>  | <code>config</code> , if set, is the blob that contains the image config. Some images do not have separate config blobs and this field will be set to nil if so.                                                                                                                                                                          |
| <b>imageMissing</b> | <b>boolean</b> | <code>imageMissing</code> is true if the image is referenced by the image stream but the image object has been deleted from the API by an administrator. When this field is set, layers and config fields may be empty and callers that depend on the image metadata should consider the image to be unavailable for download or viewing. |

| Property         | Type                  | Description                                                                                                                                                                                                                                         |
|------------------|-----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>layers</b>    | <b>array (string)</b> | layers is the list of blobs that compose this image, from base layer to top layer. All layers referenced by this array will be defined in the blobs map. Some images may have zero layers.                                                          |
| <b>manifests</b> | <b>array (string)</b> | manifests is the list of other image names that this image points to. For a single architecture image, it is empty. For a multi-arch image, it consists of the digests of single architecture images, such images shouldn't have layers nor config. |

## 6.2. API ENDPOINTS

The following API endpoints are available:

- **/apis/image.openshift.io/v1/namespaces/{namespace}/imagestreams/{name}/layers**
  - **GET**: read layers of the specified ImageStream

### 6.2.1. /apis/image.openshift.io/v1/namespaces/{namespace}/imagestreams/{name}/

Table 6.1. Global path parameters

| Parameter   | Type          | Description                   |
|-------------|---------------|-------------------------------|
| <b>name</b> | <b>string</b> | name of the ImageStreamLayers |

HTTP method

**GET**

Description

read layers of the specified ImageStream

Table 6.2. HTTP responses

| HTTP code          | Response body                   |
|--------------------|---------------------------------|
| 200 - OK           | <b>ImageStreamLayers</b> schema |
| 401 - Unauthorized | Empty                           |

## CHAPTER 7. IMAGESTREAMMAPPING

### [IMAGE.OPENSIFT.IO/V1]

#### Description

ImageStreamMapping represents a mapping from a single image stream tag to a container image as well as the reference to the container image stream the image came from. This resource is used by privileged integrators to create an image resource and to associate it with an image stream in the status tags field. Creating an ImageStreamMapping will allow any user who can view the image stream to tag or pull that image, so only create mappings where the user has proven they have access to the image contents directly. The only operation supported for this resource is create and the metadata name and namespace should be set to the image stream containing the tag that should be updated.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

#### Type

**object**

#### Required

- **image**
- **tag**

## 7.1. SPECIFICATION

| Property          | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                   |
|-------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>apiVersion</b> | <b>string</b> | APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources</a> |

| Property        | Type                 | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|-----------------|----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>image</b>    | <b>object</b>        | <p>Image is an immutable representation of a container image and metadata at a point in time. Images are named by taking a hash of their contents (metadata and content) and any change in format, content, or metadata results in a new name. The images resource is primarily for use by cluster administrators and integrations like the cluster image registry - end users instead access images via the imagestreamtags or imagestreamimages resources. While image metadata is stored in the API, any integration that implements the container image registry API must provide its own storage for the raw manifest data, image config, and layer contents.</p> <p>Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).</p> |
| <b>kind</b>     | <b>string</b>        | <p>Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info:</p> <p><a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds</a></p>                                                                                                                                                                                                                                                                                                                                                                           |
| <b>metadata</b> | <b>ObjectMeta_v2</b> | <p>metadata is the standard object's metadata. More info:</p> <p><a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata</a></p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| <b>tag</b>      | <b>string</b>        | <p>Tag is a string value this image can be located with inside the stream.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |

### 7.1.1. .image

#### Description

Image is an immutable representation of a container image and metadata at a point in time. Images are named by taking a hash of their contents (metadata and content) and any change in format, content, or metadata results in a new name. The images resource is primarily for use by cluster administrators and integrations like the cluster image registry – end users instead access images via the `imagestreamtags` or `imagestreamimages` resources. While image metadata is stored in the API, any integration that implements the container image registry API must provide its own storage for the raw manifest data, image config, and layer contents.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

#### Type

**object**

| Property                   | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                   |
|----------------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>apiVersion</b>          | <b>string</b> | APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources</a> |
| <b>dockerImageConfig</b>   | <b>string</b> | DockerImageConfig is a JSON blob that the runtime uses to set up the container. This is a part of manifest schema v2. Will not be set when the image represents a manifest list.                                                                                                                                                                                                                              |
| <b>dockerImageLayers</b>   | <b>array</b>  | DockerImageLayers represents the layers in the image. May not be set if the image does not define that data or if the image represents a manifest list.                                                                                                                                                                                                                                                       |
| <b>dockerImageLayers[]</b> | <b>object</b> | ImageLayer represents a single layer of the image. Some images may have multiple layers. Some may have none.                                                                                                                                                                                                                                                                                                  |
| <b>dockerImageManifest</b> | <b>string</b> | DockerImageManifest is the raw JSON of the manifest                                                                                                                                                                                                                                                                                                                                                           |

| Property                            | Type                  | Description                                                                                                                                                                                                                                                                                                                                                                                                         |
|-------------------------------------|-----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dockerImageManifestMediaType</b> | <b>string</b>         | DockerImageManifestMediaType specifies the mediaType of manifest. This is a part of manifest schema v2.                                                                                                                                                                                                                                                                                                             |
| <b>dockerImageManifests</b>         | <b>array</b>          | DockerImageManifests holds information about sub-manifests when the image represents a manifest list. When this field is present, no DockerImageLayers should be specified.                                                                                                                                                                                                                                         |
| <b>dockerImageManifests[]</b>       | <b>object</b>         | ImageManifest represents sub-manifests of a manifest list. The Digest field points to a regular Image object.                                                                                                                                                                                                                                                                                                       |
| <b>dockerImageMetadata</b>          | <b>RawExtension</b>   | DockerImageMetadata contains metadata about this image                                                                                                                                                                                                                                                                                                                                                              |
| <b>dockerImageMetadataVersion</b>   | <b>string</b>         | DockerImageMetadataVersion conveys the version of the object, which if empty defaults to "1.0"                                                                                                                                                                                                                                                                                                                      |
| <b>dockerImageReference</b>         | <b>string</b>         | DockerImageReference is the string that can be used to pull this image.                                                                                                                                                                                                                                                                                                                                             |
| <b>dockerImageSignatures</b>        | <b>array (string)</b> | DockerImageSignatures provides the signatures as opaque blobs. This is a part of manifest schema v1.                                                                                                                                                                                                                                                                                                                |
| <b>kind</b>                         | <b>string</b>         | Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info:<br><a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds</a> |

| Property            | Type                 | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|---------------------|----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>metadata</b>     | <b>ObjectMeta_v2</b> | metadata is the standard object's metadata. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata</a>                                                                                                                                                                                                                                                                                                                    |
| <b>signatures</b>   | <b>array</b>         | Signatures holds all signatures of the image.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| <b>signatures[]</b> | <b>object</b>        | <p>ImageSignature holds a signature of an image. It allows to verify image identity and possibly other claims as long as the signature is trusted. Based on this information it is possible to restrict runnable images to those matching cluster-wide policy. Mandatory fields should be parsed by clients doing image verification. The others are parsed from signature's content by the server. They serve just an informative purpose.</p> <p>Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).</p> |

### 7.1.2. .image.dockerImageLayers

#### Description

DockerImageLayers represents the layers in the image. May not be set if the image does not define that data or if the image represents a manifest list.

#### Type

**array**

### 7.1.3. .image.dockerImageLayers[]

#### Description

ImageLayer represents a single layer of the image. Some images may have multiple layers. Some may have none.

#### Type

**object**

#### Required

- **name**

- **size**
- **mediaType**

| Property         | Type           | Description                                                    |
|------------------|----------------|----------------------------------------------------------------|
| <b>mediaType</b> | <b>string</b>  | MediaType of the referenced object.                            |
| <b>name</b>      | <b>string</b>  | Name of the layer as defined by the underlying store.          |
| <b>size</b>      | <b>integer</b> | Size of the layer in bytes as defined by the underlying store. |

#### 7.1.4. .image.dockerImageManifests

##### Description

DockerImageManifests holds information about sub-manifests when the image represents a manifest list. When this field is present, no DockerImageLayers should be specified.

##### Type

**array**

#### 7.1.5. .image.dockerImageManifests[]

##### Description

ImageManifest represents sub-manifests of a manifest list. The Digest field points to a regular Image object.

##### Type

**object**

##### Required

- **digest**
- **mediaType**
- **manifestSize**
- **architecture**
- **os**

| Property            | Type          | Description                                                                                         |
|---------------------|---------------|-----------------------------------------------------------------------------------------------------|
| <b>architecture</b> | <b>string</b> | Architecture specifies the supported CPU architecture, for example <b>amd64</b> or <b>ppc64le</b> . |

| Property            | Type           | Description                                                                                                                                                                                                               |
|---------------------|----------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>digest</b>       | <b>string</b>  | Digest is the unique identifier for the manifest. It refers to an Image object.                                                                                                                                           |
| <b>manifestSize</b> | <b>integer</b> | ManifestSize represents the size of the raw object contents, in bytes.                                                                                                                                                    |
| <b>mediaType</b>    | <b>string</b>  | MediaType defines the type of the manifest, possible values are application/vnd.oci.image.manifest.v1+json, application/vnd.docker.distribution.manifest.v2+json or application/vnd.docker.distribution.manifest.v1+json. |
| <b>os</b>           | <b>string</b>  | OS specifies the operating system, for example <b>linux</b> .                                                                                                                                                             |
| <b>variant</b>      | <b>string</b>  | Variant is an optional field representing a variant of the CPU, for example v6 to specify a particular CPU variant of the ARM CPU.                                                                                        |

### 7.1.6. .image.signatures

#### Description

Signatures holds all signatures of the image.

#### Type

**array**

### 7.1.7. .image.signatures[]

#### Description

ImageSignature holds a signature of an image. It allows to verify image identity and possibly other claims as long as the signature is trusted. Based on this information it is possible to restrict runnable images to those matching cluster-wide policy. Mandatory fields should be parsed by clients doing image verification. The others are parsed from signature's content by the server. They serve just an informative purpose.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

#### Type

**object**

#### Required

- **type**
- **content**

| Property             | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                   |
|----------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>apiVersion</b>    | <b>string</b> | APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources</a> |
| <b>conditions</b>    | <b>array</b>  | Conditions represent the latest available observations of a signature's current state.                                                                                                                                                                                                                                                                                                                        |
| <b>conditions[]</b>  | <b>object</b> | SignatureCondition describes an image signature condition of particular kind at particular probe time.                                                                                                                                                                                                                                                                                                        |
| <b>content</b>       | <b>string</b> | Required: An opaque binary string which is an image's signature.                                                                                                                                                                                                                                                                                                                                              |
| <b>created</b>       | <b>Time</b>   | If specified, it is the time of signature's creation.                                                                                                                                                                                                                                                                                                                                                         |
| <b>imageIdentity</b> | <b>string</b> | A human readable string representing image's identity. It could be a product name and version, or an image pull spec (e.g. "registry.access.redhat.com/rhel7/rhel:7.2").                                                                                                                                                                                                                                      |
| <b>issuedBy</b>      | <b>object</b> | SignatureIssuer holds information about an issuer of signing certificate or key.                                                                                                                                                                                                                                                                                                                              |
| <b>issuedTo</b>      | <b>object</b> | SignatureSubject holds information about a person or entity who created the signature.                                                                                                                                                                                                                                                                                                                        |

| Property            | Type                   | Description                                                                                                                                                                                                                                                                                                                                                                                                      |
|---------------------|------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>kind</b>         | <b>string</b>          | Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds</a> |
| <b>metadata</b>     | <b>ObjectMeta_v2</b>   | metadata is the standard object's metadata. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata</a>                                                                                                                                                   |
| <b>signedClaims</b> | <b>object (string)</b> | Contains claims from the signature.                                                                                                                                                                                                                                                                                                                                                                              |
| <b>type</b>         | <b>string</b>          | Required: Describes a type of stored blob.                                                                                                                                                                                                                                                                                                                                                                       |

### 7.1.8. .image.signatures[].conditions

#### Description

Conditions represent the latest available observations of a signature's current state.

#### Type

**array**

### 7.1.9. .image.signatures[].conditions[]

#### Description

SignatureCondition describes an image signature condition of particular kind at particular probe time.

#### Type

**object**

#### Required

- **type**
- **status**

| Property                  | Type          | Description                                                      |
|---------------------------|---------------|------------------------------------------------------------------|
| <b>lastProbeTime</b>      | <b>Time</b>   | Last time the condition was checked.                             |
| <b>lastTransitionTime</b> | <b>Time</b>   | Last time the condition transit from one status to another.      |
| <b>message</b>            | <b>string</b> | Human readable message indicating details about last transition. |
| <b>reason</b>             | <b>string</b> | (brief) reason for the condition's last transition.              |
| <b>status</b>             | <b>string</b> | Status of the condition, one of True, False, Unknown.            |
| <b>type</b>               | <b>string</b> | Type of signature condition, Complete or Failed.                 |

### 7.1.10. .image.signatures[].issuedBy

#### Description

SignatureIssuer holds information about an issuer of signing certificate or key.

#### Type

**object**

| Property            | Type          | Description                                   |
|---------------------|---------------|-----------------------------------------------|
| <b>commonName</b>   | <b>string</b> | Common name (e.g. openshift-signing-service). |
| <b>organization</b> | <b>string</b> | Organization name.                            |

### 7.1.11. .image.signatures[].issuedTo

#### Description

SignatureSubject holds information about a person or entity who created the signature.

#### Type

**object**

#### Required

- **publicKeyID**

| Property            | Type          | Description                                                                                                                                                                                                       |
|---------------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>commonName</b>   | <b>string</b> | Common name (e.g. openshift-signing-service).                                                                                                                                                                     |
| <b>organization</b> | <b>string</b> | Organization name.                                                                                                                                                                                                |
| <b>publicKeyID</b>  | <b>string</b> | If present, it is a human readable key id of public key belonging to the subject used to verify image signature. It should contain at least 64 lowest bits of public key's fingerprint (e.g. 0x685ebe62bf278440). |

## 7.2. API ENDPOINTS

The following API endpoints are available:

- **/apis/image.openshift.io/v1/namespaces/{namespace}/imagestreammappings**
  - **POST**: create an ImageStreamMapping

### 7.2.1. /apis/image.openshift.io/v1/namespaces/{namespace}/imagestreammappings

Table 7.1. Global query parameters

| Parameter     | Type          | Description                                                                                                                                                                                                                                              |
|---------------|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dryRun</b> | <b>string</b> | When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed |

| Parameter              | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|------------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>fieldValidation</b> | <b>string</b> | fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: <ul style="list-style-type: none"> <li>- Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23.</li> <li>- Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+</li> <li>- Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.</li> </ul> |

**HTTP method****POST****Description**

create an ImageStreamMapping

**Table 7.2. Body parameters**

| Parameter   | Type                             | Description |
|-------------|----------------------------------|-------------|
| <b>body</b> | <b>ImageStreamMapping</b> schema |             |

**Table 7.3. HTTP responses**

| HTTP code          | Response body                    |
|--------------------|----------------------------------|
| 200 - OK           | <b>ImageStreamMapping</b> schema |
| 201 - Created      | <b>ImageStreamMapping</b> schema |
| 202 - Accepted     | <b>ImageStreamMapping</b> schema |
| 401 - Unauthorized | Empty                            |

## CHAPTER 8. IMAGESTREAM [IMAGE.OPENSIFT.IO/V1]

### Description

An ImageStream stores a mapping of tags to images, metadata overrides that are applied when images are tagged in a stream, and an optional reference to a container image repository on a registry. Users typically update the `spec.tags` field to point to external images which are imported from container registries using credentials in your namespace with the pull secret type, or to existing image stream tags and images which are immediately accessible for tagging or pulling. The history of images applied to a tag is visible in the `status.tags` field and any user who can view an image stream is allowed to tag that image into their own image streams. Access to pull images from the integrated registry is granted by having the "get imagestreams/layers" permission on a given image stream. Users may remove a tag by deleting the `imagestreamtag` resource, which causes both `spec` and `status` for that tag to be removed. Image stream history is retained until an administrator runs the prune operation, which removes references that are no longer in use. To preserve a historical image, ensure there is a tag in `spec` pointing to that image by its digest.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

### Type

**object**

## 8.1. SPECIFICATION

| Property          | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                      |
|-------------------|---------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>apiVersion</b> | <b>string</b> | APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources</a>    |
| <b>kind</b>       | <b>string</b> | Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds</a> |

| Property        | Type                 | Description                                                                                                                                                                                                                                                    |
|-----------------|----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>metadata</b> | <b>ObjectMeta_v2</b> | metadata is the standard object's metadata. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata</a> |
| <b>spec</b>     | <b>object</b>        | ImageStreamSpec represents options for ImageStreams.                                                                                                                                                                                                           |
| <b>status</b>   | <b>object</b>        | ImageStreamStatus contains information about the state of this image stream.                                                                                                                                                                                   |

### 8.1.1. .spec

#### Description

ImageStreamSpec represents options for ImageStreams.

#### Type

**object**

| Property                     | Type          | Description                                                                                                                                                                                                                                                                                                  |
|------------------------------|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dockerImageRepository</b> | <b>string</b> | dockerImageRepository is optional, if specified this stream is backed by a container repository on this server Deprecated: This field is deprecated as of v3.7 and will be removed in a future release. Specify the source for the tags to be imported in each tag via the spec.tags.from reference instead. |
| <b>lookupPolicy</b>          | <b>object</b> | ImageLookupPolicy describes how an image stream can be used to override the image references used by pods, builds, and other resources in a namespace.                                                                                                                                                       |
| <b>tags</b>                  | <b>array</b>  | tags map arbitrary string values to specific image locators                                                                                                                                                                                                                                                  |

| Property      | Type          | Description                                                                                                                                                                   |
|---------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>tags[]</b> | <b>object</b> | TagReference specifies optional annotations for images using this tag and an optional reference to an ImageStreamTag, ImageStreamImage, or DockerImage this tag should track. |

### 8.1.2. .spec.lookupPolicy

#### Description

ImageLookupPolicy describes how an image stream can be used to override the image references used by pods, builds, and other resources in a namespace.

#### Type

**object**

#### Required

- **local**

| Property     | Type           | Description                                                                                                                                                                                                                                                                                                                                                                                    |
|--------------|----------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>local</b> | <b>boolean</b> | local will change the docker short image references (like "mysql" or "php:latest") on objects in this namespace to the image ID whenever they match this image stream, instead of reaching out to a remote registry. The name will be fully qualified to an image ID if found. The tag's referencePolicy is taken into account on the replaced value. Only works within the current namespace. |

### 8.1.3. .spec.tags

#### Description

tags map arbitrary string values to specific image locators

#### Type

**array**

### 8.1.4. .spec.tags[]

#### Description

TagReference specifies optional annotations for images using this tag and an optional reference to an ImageStreamTag, ImageStreamImage, or DockerImage this tag should track.

## Type

**object**

## Required

- **name**

| Property            | Type                   | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|---------------------|------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>annotations</b>  | <b>object (string)</b> | Optional; if specified, annotations that are applied to images retrieved via ImageStreamTags.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| <b>from</b>         | <b>ObjectReference</b> | Optional; if specified, a reference to another image that this tag should point to. Valid values are ImageStreamTag, ImageStreamImage, and DockerImage. ImageStreamTag references can only reference a tag within this same ImageStream.                                                                                                                                                                                                                                                                                                                                                                                                                                |
| <b>generation</b>   | <b>integer</b>         | Generation is a counter that tracks mutations to the spec tag (user intent). When a tag reference is changed the generation is set to match the current stream generation (which is incremented every time spec is changed). Other processes in the system like the image importer observe that the generation of spec tag is newer than the generation recorded in the status and use that as a trigger to import the newest remote tag. To trigger a new import, clients may set this value to zero which will reset the generation to the latest stream generation. Legacy clients will send this value as nil which will be merged with the current tag generation. |
| <b>importPolicy</b> | <b>object</b>          | TagImportPolicy controls how images related to this tag will be imported.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| <b>name</b>         | <b>string</b>          | Name of the tag                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |

| Property               | Type           | Description                                                                                                                                                                                                                                    |
|------------------------|----------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>reference</b>       | <b>boolean</b> | Reference states if the tag will be imported. Default value is false, which means the tag will be imported.                                                                                                                                    |
| <b>referencePolicy</b> | <b>object</b>  | TagReferencePolicy describes how pull-specs for images in this image stream tag are generated when image change triggers in deployment configs or builds are resolved. This allows the image stream author to control how images are accessed. |

### 8.1.5. .spec.tags[].importPolicy

#### Description

TagImportPolicy controls how images related to this tag will be imported.

#### Type

**object**

| Property          | Type           | Description                                                                                                             |
|-------------------|----------------|-------------------------------------------------------------------------------------------------------------------------|
| <b>importMode</b> | <b>string</b>  | ImportMode describes how to import an image manifest.                                                                   |
| <b>insecure</b>   | <b>boolean</b> | Insecure is true if the server may bypass certificate verification or connect directly over HTTP during image import.   |
| <b>scheduled</b>  | <b>boolean</b> | Scheduled indicates to the server that this tag should be periodically checked to ensure it is up to date, and imported |

### 8.1.6. .spec.tags[].referencePolicy

#### Description

TagReferencePolicy describes how pull-specs for images in this image stream tag are generated when image change triggers in deployment configs or builds are resolved. This allows the image stream author to control how images are accessed.

#### Type

**object**

**Required**

- **type**

| Property    | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|-------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>type</b> | <b>string</b> | Type determines how the image pull spec should be transformed when the image stream tag is used in deployment config triggers or new builds. The default value is <b>Source</b> , indicating the original location of the image should be used (if imported). The user may also specify <b>Local</b> , indicating that the pull spec should point to the integrated container image registry and leverage the registry's ability to proxy the pull to an upstream registry. <b>Local</b> allows the credentials used to pull this image to be managed from the image stream's namespace, so others on the platform can access a remote image but have no access to the remote secret. It also allows the image layers to be mirrored into the local registry which the images can still be pulled even if the upstream registry is unavailable. |

**8.1.7. .status****Description**

ImageStreamStatus contains information about the state of this image stream.

**Type**

**object**

**Required**

- **dockerImageRepository**

| Property | Type | Description |
|----------|------|-------------|
|----------|------|-------------|

| Property                           | Type          | Description                                                                                                                                                                                                         |
|------------------------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dockerImageRepository</b>       | <b>string</b> | DockerImageRepository represents the effective location this stream may be accessed at. May be empty until the server determines where the repository is located                                                    |
| <b>publicDockerImageRepository</b> | <b>string</b> | PublicDockerImageRepository represents the public location from where the image can be pulled outside the cluster. This field may be empty if the administrator has not exposed the integrated registry externally. |
| <b>tags</b>                        | <b>array</b>  | Tags are a historical record of images associated with each tag. The first entry in the TagEvent array is the currently tagged image.                                                                               |
| <b>tags[]</b>                      | <b>object</b> | NamedTagEventList relates a tag to its image history.                                                                                                                                                               |

### 8.1.8. .status.tags

#### Description

Tags are a historical record of images associated with each tag. The first entry in the TagEvent array is the currently tagged image.

#### Type

**array**

### 8.1.9. .status.tags[]

#### Description

NamedTagEventList relates a tag to its image history.

#### Type

**object**

#### Required

- **tag**
- **items**

| Property            | Type          | Description                                                                                        |
|---------------------|---------------|----------------------------------------------------------------------------------------------------|
| <b>conditions</b>   | <b>array</b>  | Conditions is an array of conditions that apply to the tag event list.                             |
| <b>conditions[]</b> | <b>object</b> | TagEventCondition contains condition information for a tag event.                                  |
| <b>items</b>        | <b>array</b>  | Standard object's metadata.                                                                        |
| <b>items[]</b>      | <b>object</b> | TagEvent is used by ImageStreamStatus to keep a historical record of images associated with a tag. |
| <b>tag</b>          | <b>string</b> | Tag is the tag for which the history is recorded                                                   |

#### 8.1.10. .status.tags[].conditions

##### Description

Conditions is an array of conditions that apply to the tag event list.

##### Type

**array**

#### 8.1.11. .status.tags[].conditions[]

##### Description

TagEventCondition contains condition information for a tag event.

##### Type

**object**

##### Required

- **type**
- **status**
- **generation**

| Property          | Type           | Description                                                           |
|-------------------|----------------|-----------------------------------------------------------------------|
| <b>generation</b> | <b>integer</b> | Generation is the spec tag generation that this status corresponds to |

| Property                  | Type          | Description                                                                                         |
|---------------------------|---------------|-----------------------------------------------------------------------------------------------------|
| <b>lastTransitionTime</b> | <b>Time</b>   | LastTransitionTime is the time the condition transitioned from one status to another.               |
| <b>message</b>            | <b>string</b> | Message is a human readable description of the details about last transition, complementing reason. |
| <b>reason</b>             | <b>string</b> | Reason is a brief machine readable explanation for the condition's last transition.                 |
| <b>status</b>             | <b>string</b> | Status of the condition, one of True, False, Unknown.                                               |
| <b>type</b>               | <b>string</b> | Type of tag event condition, currently only ImportSuccess                                           |

### 8.1.12. `.status.tags[].items`

#### Description

Standard object's metadata.

#### Type

**array**

### 8.1.13. `.status.tags[].items[]`

#### Description

TagEvent is used by ImageStreamStatus to keep a historical record of images associated with a tag.

#### Type

**object**

#### Required

- **created**
- **dockerImageReference**
- **image**
- **generation**

| Property                    | Type           | Description                                                                   |
|-----------------------------|----------------|-------------------------------------------------------------------------------|
| <b>created</b>              | <b>Time</b>    | Created holds the time the TagEvent was created                               |
| <b>dockerImageReference</b> | <b>string</b>  | DockerImageReference is the string that can be used to pull this image        |
| <b>generation</b>           | <b>integer</b> | Generation is the spec tag generation that resulted in this tag being updated |
| <b>image</b>                | <b>string</b>  | Image is the image                                                            |

## 8.2. API ENDPOINTS

The following API endpoints are available:

- **/apis/image.openshift.io/v1/imagestreams**
  - **GET**: list or watch objects of kind ImageStream
- **/apis/image.openshift.io/v1/watch/imagestreams**
  - **GET**: watch individual changes to a list of ImageStream. deprecated: use the 'watch' parameter with a list operation instead.
- **/apis/image.openshift.io/v1/namespaces/{namespace}/imagestreams**
  - **DELETE**: delete collection of ImageStream
  - **GET**: list or watch objects of kind ImageStream
  - **POST**: create an ImageStream
- **/apis/image.openshift.io/v1/watch/namespaces/{namespace}/imagestreams**
  - **GET**: watch individual changes to a list of ImageStream. deprecated: use the 'watch' parameter with a list operation instead.
- **/apis/image.openshift.io/v1/namespaces/{namespace}/imagestreams/{name}**
  - **DELETE**: delete an ImageStream
  - **GET**: read the specified ImageStream
  - **PATCH**: partially update the specified ImageStream
  - **PUT**: replace the specified ImageStream
- **/apis/image.openshift.io/v1/watch/namespaces/{namespace}/imagestreams/{name}**

- **GET**: watch changes to an object of kind ImageStream. deprecated: use the 'watch' parameter with a list operation instead, filtered to a single item with the 'fieldSelector' parameter.
- **/apis/image.openshift.io/v1/namespaces/{namespace}/imagestreams/{name}/status**
  - **GET**: read status of the specified ImageStream
  - **PATCH**: partially update status of the specified ImageStream
  - **PUT**: replace status of the specified ImageStream

### 8.2.1. /apis/image.openshift.io/v1/imagestreams

HTTP method

**GET**

Description

list or watch objects of kind ImageStream

Table 8.1. HTTP responses

| HTTP code          | Reponse body                           |
|--------------------|----------------------------------------|
| 200 - OK           | <a href="#">ImageStreamList</a> schema |
| 401 - Unauthorized | Empty                                  |

### 8.2.2. /apis/image.openshift.io/v1/watch/imagestreams

HTTP method

**GET**

Description

watch individual changes to a list of ImageStream. deprecated: use the 'watch' parameter with a list operation instead.

Table 8.2. HTTP responses

| HTTP code          | Reponse body                      |
|--------------------|-----------------------------------|
| 200 - OK           | <a href="#">WatchEvent</a> schema |
| 401 - Unauthorized | Empty                             |

### 8.2.3. /apis/image.openshift.io/v1/namespaces/{namespace}/imagestreams

HTTP method

**DELETE**

Description

delete collection of ImageStream

**Table 8.3. Query parameters**

| Parameter     | Type          | Description                                                                                                                                                                                                                                              |
|---------------|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dryRun</b> | <b>string</b> | When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed |

**Table 8.4. HTTP responses**

| HTTP code          | Reponse body            |
|--------------------|-------------------------|
| 200 - OK           | <b>Status_v5</b> schema |
| 401 - Unauthorized | Empty                   |

#### HTTP method

#### GET

#### Description

list or watch objects of kind ImageStream

**Table 8.5. HTTP responses**

| HTTP code          | Reponse body                  |
|--------------------|-------------------------------|
| 200 - OK           | <b>ImageStreamList</b> schema |
| 401 - Unauthorized | Empty                         |

#### HTTP method

#### POST

#### Description

create an ImageStream

**Table 8.6. Query parameters**

| Parameter     | Type          | Description                                                                                                                                                                                                                                              |
|---------------|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dryRun</b> | <b>string</b> | When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed |

| Parameter              | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|------------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>fieldValidation</b> | <b>string</b> | fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: <ul style="list-style-type: none"> <li>- Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23.</li> <li>- Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+</li> <li>- Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.</li> </ul> |

Table 8.7. Body parameters

| Parameter   | Type                      | Description |
|-------------|---------------------------|-------------|
| <b>body</b> | <b>ImageStream</b> schema |             |

Table 8.8. HTTP responses

| HTTP code          | Response body             |
|--------------------|---------------------------|
| 200 - OK           | <b>ImageStream</b> schema |
| 201 - Created      | <b>ImageStream</b> schema |
| 202 - Accepted     | <b>ImageStream</b> schema |
| 401 - Unauthorized | Empty                     |

### 8.2.4. /apis/image.openshift.io/v1/watch/namespaces/{namespace}/imagestreams

#### HTTP method

#### GET

#### Description

watch individual changes to a list of ImageStream. deprecated: use the 'watch' parameter with a list operation instead.

Table 8.9. HTTP responses

| HTTP code          | Reponse body                      |
|--------------------|-----------------------------------|
| 200 - OK           | <a href="#">WatchEvent</a> schema |
| 401 - Unauthorized | Empty                             |

### 8.2.5. /apis/image.openshift.io/v1/namespaces/{namespace}/imagestreams/{name}

Table 8.10. Global path parameters

| Parameter   | Type          | Description             |
|-------------|---------------|-------------------------|
| <b>name</b> | <b>string</b> | name of the ImageStream |

#### HTTP method

#### DELETE

#### Description

delete an ImageStream

Table 8.11. Query parameters

| Parameter     | Type          | Description                                                                                                                                                                                                                                              |
|---------------|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dryRun</b> | <b>string</b> | When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed |

Table 8.12. HTTP responses

| HTTP code          | Reponse body                     |
|--------------------|----------------------------------|
| 200 - OK           | <a href="#">Status_v5</a> schema |
| 202 - Accepted     | <a href="#">Status_v5</a> schema |
| 401 - Unauthorized | Empty                            |

#### HTTP method

#### GET

#### Description

read the specified ImageStream

Table 8.13. HTTP responses

| HTTP code          | Response body                      |
|--------------------|------------------------------------|
| 200 - OK           | <a href="#">ImageStream</a> schema |
| 401 - Unauthorized | Empty                              |

**HTTP method****PATCH****Description**

partially update the specified ImageStream

Table 8.14. Query parameters

| Parameter              | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|------------------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dryRun</b>          | <b>string</b> | When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <b>fieldValidation</b> | <b>string</b> | fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered. |

Table 8.15. HTTP responses

| HTTP code     | Response body                      |
|---------------|------------------------------------|
| 200 - OK      | <a href="#">ImageStream</a> schema |
| 201 - Created | <a href="#">ImageStream</a> schema |

| HTTP code          | Response body |
|--------------------|---------------|
| 401 - Unauthorized | Empty         |

**HTTP method****PUT****Description**

replace the specified ImageStream

**Table 8.16. Query parameters**

| Parameter              | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|------------------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dryRun</b>          | <b>string</b> | When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <b>fieldValidation</b> | <b>string</b> | fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered. |

**Table 8.17. Body parameters**

| Parameter   | Type                      | Description |
|-------------|---------------------------|-------------|
| <b>body</b> | <b>ImageStream</b> schema |             |

**Table 8.18. HTTP responses**

| HTTP code          | Reponse body                       |
|--------------------|------------------------------------|
| 200 - OK           | <a href="#">ImageStream</a> schema |
| 201 - Created      | <a href="#">ImageStream</a> schema |
| 401 - Unauthorized | Empty                              |

### 8.2.6. /apis/image.openshift.io/v1/watch/namespaces/{namespace}/imagestreams/{

Table 8.19. Global path parameters

| Parameter   | Type          | Description             |
|-------------|---------------|-------------------------|
| <b>name</b> | <b>string</b> | name of the ImageStream |

#### HTTP method

#### GET

#### Description

watch changes to an object of kind ImageStream. deprecated: use the 'watch' parameter with a list operation instead, filtered to a single item with the 'fieldSelector' parameter.

Table 8.20. HTTP responses

| HTTP code          | Reponse body                      |
|--------------------|-----------------------------------|
| 200 - OK           | <a href="#">WatchEvent</a> schema |
| 401 - Unauthorized | Empty                             |

### 8.2.7. /apis/image.openshift.io/v1/namespaces/{namespace}/imagestreams/{name}/

Table 8.21. Global path parameters

| Parameter   | Type          | Description             |
|-------------|---------------|-------------------------|
| <b>name</b> | <b>string</b> | name of the ImageStream |

#### HTTP method

#### GET

#### Description

read status of the specified ImageStream

Table 8.22. HTTP responses

| HTTP code          | Reponse body              |
|--------------------|---------------------------|
| 200 - OK           | <b>ImageStream</b> schema |
| 401 - Unauthorized | Empty                     |

**HTTP method****PATCH****Description**

partially update status of the specified ImageStream

**Table 8.23. Query parameters**

| Parameter              | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|------------------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dryRun</b>          | <b>string</b> | When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <b>fieldValidation</b> | <b>string</b> | fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered. |

**Table 8.24. HTTP responses**

| HTTP code     | Reponse body              |
|---------------|---------------------------|
| 200 - OK      | <b>ImageStream</b> schema |
| 201 - Created | <b>ImageStream</b> schema |

| HTTP code          | Response body |
|--------------------|---------------|
| 401 - Unauthorized | Empty         |

**HTTP method****PUT****Description**

replace status of the specified ImageStream

**Table 8.25. Query parameters**

| Parameter              | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|------------------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dryRun</b>          | <b>string</b> | When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <b>fieldValidation</b> | <b>string</b> | fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered. |

**Table 8.26. Body parameters**

| Parameter   | Type                      | Description |
|-------------|---------------------------|-------------|
| <b>body</b> | <b>ImageStream</b> schema |             |

**Table 8.27. HTTP responses**

| HTTP code          | Reponse body                       |
|--------------------|------------------------------------|
| 200 - OK           | <a href="#">ImageStream</a> schema |
| 201 - Created      | <a href="#">ImageStream</a> schema |
| 401 - Unauthorized | Empty                              |

## CHAPTER 9. IMAGESTREAMTAG [IMAGE.OPENSIFT.IO/V1]

### Description

ImageStreamTag represents an Image that is retrieved by tag name from an ImageStream. Use this resource to interact with the tags and images in an image stream by tag, or to see the image details for a particular tag. The image associated with this resource is the most recently successfully tagged, imported, or pushed image (as described in the image stream status.tags.items list for this tag). If an import is in progress or has failed the previous image will be shown. Deleting an image stream tag clears both the status and spec fields of an image stream. If no image can be retrieved for a given tag, a not found error will be returned.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

### Type

**object**

### Required

- **tag**
- **generation**
- **lookupPolicy**
- **image**

## 9.1. SPECIFICATION

| Property            | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                   |
|---------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>apiVersion</b>   | <b>string</b> | APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources</a> |
| <b>conditions</b>   | <b>array</b>  | conditions is an array of conditions that apply to the image stream tag.                                                                                                                                                                                                                                                                                                                                      |
| <b>conditions[]</b> | <b>object</b> | TagEventCondition contains condition information for a tag event.                                                                                                                                                                                                                                                                                                                                             |

| Property          | Type           | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|-------------------|----------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>generation</b> | <b>integer</b> | generation is the current generation of the tagged image - if tag is provided and this value is not equal to the tag generation, a user has requested an import that has not completed, or conditions will be filled out indicating any error.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| <b>image</b>      | <b>object</b>  | <p>Image is an immutable representation of a container image and metadata at a point in time. Images are named by taking a hash of their contents (metadata and content) and any change in format, content, or metadata results in a new name. The images resource is primarily for use by cluster administrators and integrations like the cluster image registry - end users instead access images via the imagestreamtags or imagestreamimages resources. While image metadata is stored in the API, any integration that implements the container image registry API must provide its own storage for the raw manifest data, image config, and layer contents.</p> <p>Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).</p> |
| <b>kind</b>       | <b>string</b>  | <p>Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info:</p> <p><a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds</a></p>                                                                                                                                                                                                                                                                                                                                                                           |

| Property            | Type                 | Description                                                                                                                                                                                                                                                    |
|---------------------|----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>lookupPolicy</b> | <b>object</b>        | ImageLookupPolicy describes how an image stream can be used to override the image references used by pods, builds, and other resources in a namespace.                                                                                                         |
| <b>metadata</b>     | <b>ObjectMeta_v2</b> | metadata is the standard object's metadata. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata</a> |
| <b>tag</b>          | <b>object</b>        | TagReference specifies optional annotations for images using this tag and an optional reference to an ImageStreamTag, ImageStreamImage, or DockerImage this tag should track.                                                                                  |

### 9.1.1. .conditions

#### Description

conditions is an array of conditions that apply to the image stream tag.

#### Type

**array**

### 9.1.2. .conditions[]

#### Description

TagEventCondition contains condition information for a tag event.

#### Type

**object**

#### Required

- **type**
- **status**
- **generation**

| Property                  | Type           | Description                                                                                         |
|---------------------------|----------------|-----------------------------------------------------------------------------------------------------|
| <b>generation</b>         | <b>integer</b> | Generation is the spec tag generation that this status corresponds to                               |
| <b>lastTransitionTime</b> | <b>Time</b>    | LastTransitionTime is the time the condition transitioned from one status to another.               |
| <b>message</b>            | <b>string</b>  | Message is a human readable description of the details about last transition, complementing reason. |
| <b>reason</b>             | <b>string</b>  | Reason is a brief machine readable explanation for the condition's last transition.                 |
| <b>status</b>             | <b>string</b>  | Status of the condition, one of True, False, Unknown.                                               |
| <b>type</b>               | <b>string</b>  | Type of tag event condition, currently only ImportSuccess                                           |

### 9.1.3. .image

#### Description

Image is an immutable representation of a container image and metadata at a point in time. Images are named by taking a hash of their contents (metadata and content) and any change in format, content, or metadata results in a new name. The images resource is primarily for use by cluster administrators and integrations like the cluster image registry – end users instead access images via the imagestreamtags or imagestreamimages resources. While image metadata is stored in the API, any integration that implements the container image registry API must provide its own storage for the raw manifest data, image config, and layer contents.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

#### Type

**object**

| Property | Type | Description |
|----------|------|-------------|
|----------|------|-------------|

| Property                            | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                   |
|-------------------------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>apiVersion</b>                   | <b>string</b> | APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources</a> |
| <b>dockerImageConfig</b>            | <b>string</b> | DockerImageConfig is a JSON blob that the runtime uses to set up the container. This is a part of manifest schema v2. Will not be set when the image represents a manifest list.                                                                                                                                                                                                                              |
| <b>dockerImageLayers</b>            | <b>array</b>  | DockerImageLayers represents the layers in the image. May not be set if the image does not define that data or if the image represents a manifest list.                                                                                                                                                                                                                                                       |
| <b>dockerImageLayers[]</b>          | <b>object</b> | ImageLayer represents a single layer of the image. Some images may have multiple layers. Some may have none.                                                                                                                                                                                                                                                                                                  |
| <b>dockerImageManifest</b>          | <b>string</b> | DockerImageManifest is the raw JSON of the manifest                                                                                                                                                                                                                                                                                                                                                           |
| <b>dockerImageManifestMediaType</b> | <b>string</b> | DockerImageManifestMediaType specifies the mediaType of manifest. This is a part of manifest schema v2.                                                                                                                                                                                                                                                                                                       |
| <b>dockerImageManifests</b>         | <b>array</b>  | DockerImageManifests holds information about sub-manifests when the image represents a manifest list. When this field is present, no DockerImageLayers should be specified.                                                                                                                                                                                                                                   |
| <b>dockerImageManifests[]</b>       | <b>object</b> | ImageManifest represents sub-manifests of a manifest list. The Digest field points to a regular Image object.                                                                                                                                                                                                                                                                                                 |

| Property                          | Type                  | Description                                                                                                                                                                                                                                                                                                                                                                                                      |
|-----------------------------------|-----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dockerImageMetadata</b>        | <b>RawExtension</b>   | DockerImageMetadata contains metadata about this image                                                                                                                                                                                                                                                                                                                                                           |
| <b>dockerImageMetadataVersion</b> | <b>string</b>         | DockerImageMetadataVersion conveys the version of the object, which if empty defaults to "1.0"                                                                                                                                                                                                                                                                                                                   |
| <b>dockerImageReference</b>       | <b>string</b>         | DockerImageReference is the string that can be used to pull this image.                                                                                                                                                                                                                                                                                                                                          |
| <b>dockerImageSignatures</b>      | <b>array (string)</b> | DockerImageSignatures provides the signatures as opaque blobs. This is a part of manifest schema v1.                                                                                                                                                                                                                                                                                                             |
| <b>kind</b>                       | <b>string</b>         | Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds</a> |
| <b>metadata</b>                   | <b>ObjectMeta_v2</b>  | metadata is the standard object's metadata. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata</a>                                                                                                                                                   |
| <b>signatures</b>                 | <b>array</b>          | Signatures holds all signatures of the image.                                                                                                                                                                                                                                                                                                                                                                    |

| Property            | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|---------------------|---------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>signatures[]</b> | <b>object</b> | <p>ImageSignature holds a signature of an image. It allows to verify image identity and possibly other claims as long as the signature is trusted. Based on this information it is possible to restrict runnable images to those matching cluster-wide policy. Mandatory fields should be parsed by clients doing image verification. The others are parsed from signature's content by the server. They serve just an informative purpose.</p> <p>Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).</p> |

#### 9.1.4. .image.dockerImageLayers

##### Description

DockerImageLayers represents the layers in the image. May not be set if the image does not define that data or if the image represents a manifest list.

##### Type

**array**

#### 9.1.5. .image.dockerImageLayers[]

##### Description

ImageLayer represents a single layer of the image. Some images may have multiple layers. Some may have none.

##### Type

**object**

##### Required

- **name**
- **size**
- **mediaType**

| Property         | Type          | Description                         |
|------------------|---------------|-------------------------------------|
| <b>mediaType</b> | <b>string</b> | MediaType of the referenced object. |

| Property    | Type           | Description                                                    |
|-------------|----------------|----------------------------------------------------------------|
| <b>name</b> | <b>string</b>  | Name of the layer as defined by the underlying store.          |
| <b>size</b> | <b>integer</b> | Size of the layer in bytes as defined by the underlying store. |

### 9.1.6. .image.dockerImageManifests

#### Description

DockerImageManifests holds information about sub-manifests when the image represents a manifest list. When this field is present, no DockerImageLayers should be specified.

#### Type

**array**

### 9.1.7. .image.dockerImageManifests[]

#### Description

ImageManifest represents sub-manifests of a manifest list. The Digest field points to a regular Image object.

#### Type

**object**

#### Required

- **digest**
- **mediaType**
- **manifestSize**
- **architecture**
- **os**

| Property            | Type          | Description                                                                                         |
|---------------------|---------------|-----------------------------------------------------------------------------------------------------|
| <b>architecture</b> | <b>string</b> | Architecture specifies the supported CPU architecture, for example <b>amd64</b> or <b>ppc64le</b> . |
| <b>digest</b>       | <b>string</b> | Digest is the unique identifier for the manifest. It refers to an Image object.                     |

| Property            | Type           | Description                                                                                                                                                                                                               |
|---------------------|----------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>manifestSize</b> | <b>integer</b> | ManifestSize represents the size of the raw object contents, in bytes.                                                                                                                                                    |
| <b>mediaType</b>    | <b>string</b>  | MediaType defines the type of the manifest, possible values are application/vnd.oci.image.manifest.v1+json, application/vnd.docker.distribution.manifest.v2+json or application/vnd.docker.distribution.manifest.v1+json. |
| <b>os</b>           | <b>string</b>  | OS specifies the operating system, for example <b>linux</b> .                                                                                                                                                             |
| <b>variant</b>      | <b>string</b>  | Variant is an optional field representing a variant of the CPU, for example v6 to specify a particular CPU variant of the ARM CPU.                                                                                        |

### 9.1.8. .image.signatures

#### Description

Signatures holds all signatures of the image.

#### Type

**array**

### 9.1.9. .image.signatures[]

#### Description

ImageSignature holds a signature of an image. It allows to verify image identity and possibly other claims as long as the signature is trusted. Based on this information it is possible to restrict runnable images to those matching cluster-wide policy. Mandatory fields should be parsed by clients doing image verification. The others are parsed from signature's content by the server. They serve just an informative purpose.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

#### Type

**object**

#### Required

- **type**
- **content**

| Property             | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                   |
|----------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>apiVersion</b>    | <b>string</b> | APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources</a> |
| <b>conditions</b>    | <b>array</b>  | Conditions represent the latest available observations of a signature's current state.                                                                                                                                                                                                                                                                                                                        |
| <b>conditions[]</b>  | <b>object</b> | SignatureCondition describes an image signature condition of particular kind at particular probe time.                                                                                                                                                                                                                                                                                                        |
| <b>content</b>       | <b>string</b> | Required: An opaque binary string which is an image's signature.                                                                                                                                                                                                                                                                                                                                              |
| <b>created</b>       | <b>Time</b>   | If specified, it is the time of signature's creation.                                                                                                                                                                                                                                                                                                                                                         |
| <b>imageIdentity</b> | <b>string</b> | A human readable string representing image's identity. It could be a product name and version, or an image pull spec (e.g. "registry.access.redhat.com/rhel7/rhel:7.2").                                                                                                                                                                                                                                      |
| <b>issuedBy</b>      | <b>object</b> | SignatureIssuer holds information about an issuer of signing certificate or key.                                                                                                                                                                                                                                                                                                                              |
| <b>issuedTo</b>      | <b>object</b> | SignatureSubject holds information about a person or entity who created the signature.                                                                                                                                                                                                                                                                                                                        |

| Property            | Type                                 | Description                                                                                                                                                                                                                                                                                                                                                                                                      |
|---------------------|--------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>kind</b>         | <b>string</b>                        | Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds</a> |
| <b>metadata</b>     | <b><a href="#">ObjectMeta_v2</a></b> | metadata is the standard object's metadata. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata</a>                                                                                                                                                   |
| <b>signedClaims</b> | <b>object (string)</b>               | Contains claims from the signature.                                                                                                                                                                                                                                                                                                                                                                              |
| <b>type</b>         | <b>string</b>                        | Required: Describes a type of stored blob.                                                                                                                                                                                                                                                                                                                                                                       |

### 9.1.10. .image.signatures[].conditions

#### Description

Conditions represent the latest available observations of a signature's current state.

#### Type

**array**

### 9.1.11. .image.signatures[].conditions[]

#### Description

SignatureCondition describes an image signature condition of particular kind at particular probe time.

#### Type

**object**

#### Required

- **type**
- **status**

| Property                  | Type          | Description                                                      |
|---------------------------|---------------|------------------------------------------------------------------|
| <b>lastProbeTime</b>      | <b>Time</b>   | Last time the condition was checked.                             |
| <b>lastTransitionTime</b> | <b>Time</b>   | Last time the condition transit from one status to another.      |
| <b>message</b>            | <b>string</b> | Human readable message indicating details about last transition. |
| <b>reason</b>             | <b>string</b> | (brief) reason for the condition's last transition.              |
| <b>status</b>             | <b>string</b> | Status of the condition, one of True, False, Unknown.            |
| <b>type</b>               | <b>string</b> | Type of signature condition, Complete or Failed.                 |

### 9.1.12. .image.signatures[].issuedBy

#### Description

SignatureIssuer holds information about an issuer of signing certificate or key.

#### Type

**object**

| Property            | Type          | Description                                   |
|---------------------|---------------|-----------------------------------------------|
| <b>commonName</b>   | <b>string</b> | Common name (e.g. openshift-signing-service). |
| <b>organization</b> | <b>string</b> | Organization name.                            |

### 9.1.13. .image.signatures[].issuedTo

#### Description

SignatureSubject holds information about a person or entity who created the signature.

#### Type

**object**

#### Required

- **publicKeyID**

| Property            | Type          | Description                                                                                                                                                                                                       |
|---------------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>commonName</b>   | <b>string</b> | Common name (e.g. openshift-signing-service).                                                                                                                                                                     |
| <b>organization</b> | <b>string</b> | Organization name.                                                                                                                                                                                                |
| <b>publicKeyID</b>  | <b>string</b> | If present, it is a human readable key id of public key belonging to the subject used to verify image signature. It should contain at least 64 lowest bits of public key's fingerprint (e.g. 0x685ebe62bf278440). |

### 9.1.14. .lookupPolicy

#### Description

ImageLookupPolicy describes how an image stream can be used to override the image references used by pods, builds, and other resources in a namespace.

#### Type

**object**

#### Required

- **local**

| Property     | Type           | Description                                                                                                                                                                                                                                                                                                                                                                                    |
|--------------|----------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>local</b> | <b>boolean</b> | local will change the docker short image references (like "mysql" or "php:latest") on objects in this namespace to the image ID whenever they match this image stream, instead of reaching out to a remote registry. The name will be fully qualified to an image ID if found. The tag's referencePolicy is taken into account on the replaced value. Only works within the current namespace. |

### 9.1.15. .tag

#### Description

TagReference specifies optional annotations for images using this tag and an optional reference to an ImageStreamTag, ImageStreamImage, or DockerImage this tag should track.

#### Type

**object**

## Required

- **name**

| Property            | Type                   | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|---------------------|------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>annotations</b>  | <b>object (string)</b> | Optional; if specified, annotations that are applied to images retrieved via ImageStreamTags.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| <b>from</b>         | <b>ObjectReference</b> | Optional; if specified, a reference to another image that this tag should point to. Valid values are ImageStreamTag, ImageStreamImage, and DockerImage. ImageStreamTag references can only reference a tag within this same ImageStream.                                                                                                                                                                                                                                                                                                                                                                                                                                |
| <b>generation</b>   | <b>integer</b>         | Generation is a counter that tracks mutations to the spec tag (user intent). When a tag reference is changed the generation is set to match the current stream generation (which is incremented every time spec is changed). Other processes in the system like the image importer observe that the generation of spec tag is newer than the generation recorded in the status and use that as a trigger to import the newest remote tag. To trigger a new import, clients may set this value to zero which will reset the generation to the latest stream generation. Legacy clients will send this value as nil which will be merged with the current tag generation. |
| <b>importPolicy</b> | <b>object</b>          | TagImportPolicy controls how images related to this tag will be imported.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| <b>name</b>         | <b>string</b>          | Name of the tag                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| <b>reference</b>    | <b>boolean</b>         | Reference states if the tag will be imported. Default value is false, which means the tag will be imported.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |

| Property               | Type          | Description                                                                                                                                                                                                                                    |
|------------------------|---------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>referencePolicy</b> | <b>object</b> | TagReferencePolicy describes how pull-specs for images in this image stream tag are generated when image change triggers in deployment configs or builds are resolved. This allows the image stream author to control how images are accessed. |

### 9.1.16. .tag.importPolicy

#### Description

TagImportPolicy controls how images related to this tag will be imported.

#### Type

**object**

| Property          | Type           | Description                                                                                                             |
|-------------------|----------------|-------------------------------------------------------------------------------------------------------------------------|
| <b>importMode</b> | <b>string</b>  | ImportMode describes how to import an image manifest.                                                                   |
| <b>insecure</b>   | <b>boolean</b> | Insecure is true if the server may bypass certificate verification or connect directly over HTTP during image import.   |
| <b>scheduled</b>  | <b>boolean</b> | Scheduled indicates to the server that this tag should be periodically checked to ensure it is up to date, and imported |

### 9.1.17. .tag.referencePolicy

#### Description

TagReferencePolicy describes how pull-specs for images in this image stream tag are generated when image change triggers in deployment configs or builds are resolved. This allows the image stream author to control how images are accessed.

#### Type

**object**

#### Required

- **type**

| Property    | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|-------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>type</b> | <b>string</b> | Type determines how the image pull spec should be transformed when the image stream tag is used in deployment config triggers or new builds. The default value is <b>Source</b> , indicating the original location of the image should be used (if imported). The user may also specify <b>Local</b> , indicating that the pull spec should point to the integrated container image registry and leverage the registry's ability to proxy the pull to an upstream registry. <b>Local</b> allows the credentials used to pull this image to be managed from the image stream's namespace, so others on the platform can access a remote image but have no access to the remote secret. It also allows the image layers to be mirrored into the local registry which the images can still be pulled even if the upstream registry is unavailable. |

## 9.2. API ENDPOINTS

The following API endpoints are available:

- **/apis/image.openshift.io/v1/imagestreamtags**
  - **GET**: list objects of kind ImageStreamTag
- **/apis/image.openshift.io/v1/namespaces/{namespace}/imagestreamtags**
  - **GET**: list objects of kind ImageStreamTag
  - **POST**: create an ImageStreamTag
- **/apis/image.openshift.io/v1/namespaces/{namespace}/imagestreamtags/{name}**
  - **DELETE**: delete an ImageStreamTag
  - **GET**: read the specified ImageStreamTag
  - **PATCH**: partially update the specified ImageStreamTag
  - **PUT**: replace the specified ImageStreamTag

### 9.2.1. /apis/image.openshift.io/v1/imagestreamtags

HTTP method

**GET**

Description

list objects of kind ImageStreamTag

Table 9.1. HTTP responses

| HTTP code          | Reponse body                              |
|--------------------|-------------------------------------------|
| 200 - OK           | <a href="#">ImageStreamTagList</a> schema |
| 401 - Unauthorized | Empty                                     |

### 9.2.2. /apis/image.openshift.io/v1/namespaces/{namespace}/imagestreamtags

HTTP method

**GET**

Description

list objects of kind ImageStreamTag

Table 9.2. HTTP responses

| HTTP code          | Reponse body                              |
|--------------------|-------------------------------------------|
| 200 - OK           | <a href="#">ImageStreamTagList</a> schema |
| 401 - Unauthorized | Empty                                     |

HTTP method

**POST**

Description

create an ImageStreamTag

Table 9.3. Query parameters

| Parameter     | Type          | Description                                                                                                                                                                                                                                              |
|---------------|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dryRun</b> | <b>string</b> | When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed |

| Parameter              | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|------------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>fieldValidation</b> | <b>string</b> | fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: <ul style="list-style-type: none"> <li>- Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23.</li> <li>- Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+</li> <li>- Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.</li> </ul> |

Table 9.4. Body parameters

| Parameter   | Type                            | Description |
|-------------|---------------------------------|-------------|
| <b>body</b> | <b>ImageStreamTag</b><br>schema |             |

Table 9.5. HTTP responses

| HTTP code          | Response body                |
|--------------------|------------------------------|
| 200 - OK           | <b>ImageStreamTag</b> schema |
| 201 - Created      | <b>ImageStreamTag</b> schema |
| 202 - Accepted     | <b>ImageStreamTag</b> schema |
| 401 - Unauthorized | Empty                        |

### 9.2.3. /apis/image.openshift.io/v1/namespaces/{namespace}/imagestreamtags/{name}

Table 9.6. Global path parameters

| Parameter   | Type          | Description                |
|-------------|---------------|----------------------------|
| <b>name</b> | <b>string</b> | name of the ImageStreamTag |

**HTTP method****DELETE****Description**

delete an ImageStreamTag

**Table 9.7. Query parameters**

| Parameter     | Type          | Description                                                                                                                                                                                                                                              |
|---------------|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dryRun</b> | <b>string</b> | When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed |

**Table 9.8. HTTP responses**

| HTTP code          | Reponse body                     |
|--------------------|----------------------------------|
| 200 - OK           | <a href="#">Status_v5</a> schema |
| 202 - Accepted     | <a href="#">Status_v5</a> schema |
| 401 - Unauthorized | Empty                            |

**HTTP method****GET****Description**

read the specified ImageStreamTag

**Table 9.9. HTTP responses**

| HTTP code          | Reponse body                          |
|--------------------|---------------------------------------|
| 200 - OK           | <a href="#">ImageStreamTag</a> schema |
| 401 - Unauthorized | Empty                                 |

**HTTP method****PATCH****Description**

partially update the specified ImageStreamTag

**Table 9.10. Query parameters**

| Parameter              | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|------------------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dryRun</b>          | <b>string</b> | When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <b>fieldValidation</b> | <b>string</b> | fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered. |

Table 9.11. HTTP responses

| HTTP code          | Response body                         |
|--------------------|---------------------------------------|
| 200 - OK           | <a href="#">ImageStreamTag</a> schema |
| 201 - Created      | <a href="#">ImageStreamTag</a> schema |
| 401 - Unauthorized | Empty                                 |

**HTTP method****PUT****Description**

replace the specified ImageStreamTag

Table 9.12. Query parameters

| Parameter | Type | Description |
|-----------|------|-------------|
|-----------|------|-------------|

| Parameter              | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|------------------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dryRun</b>          | <b>string</b> | When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <b>fieldValidation</b> | <b>string</b> | fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered. |

Table 9.13. Body parameters

| Parameter   | Type                            | Description |
|-------------|---------------------------------|-------------|
| <b>body</b> | <b>ImageStreamTag</b><br>schema |             |

Table 9.14. HTTP responses

| HTTP code          | Reponse body                 |
|--------------------|------------------------------|
| 200 - OK           | <b>ImageStreamTag</b> schema |
| 201 - Created      | <b>ImageStreamTag</b> schema |
| 401 - Unauthorized | Empty                        |

## CHAPTER 10. IMAGETAG [IMAGE.OPENSIFT.IO/V1]

### Description

ImageTag represents a single tag within an image stream and includes the spec, the status history, and the currently referenced image (if any) of the provided tag. This type replaces the ImageStreamTag by providing a full view of the tag. ImageTags are returned for every spec or status tag present on the image stream. If no tag exists in either form a not found error will be returned by the API. A create operation will succeed if no spec tag has already been defined and the spec field is set. Delete will remove both spec and status elements from the image stream.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

### Type

**object**

### Required

- **spec**
- **status**
- **image**

## 10.1. SPECIFICATION

| Property          | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                   |
|-------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>apiVersion</b> | <b>string</b> | APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources</a> |

| Property        | Type                 | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|-----------------|----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>image</b>    | <b>object</b>        | <p>Image is an immutable representation of a container image and metadata at a point in time. Images are named by taking a hash of their contents (metadata and content) and any change in format, content, or metadata results in a new name. The images resource is primarily for use by cluster administrators and integrations like the cluster image registry - end users instead access images via the imagestreamtags or imagestreamimages resources. While image metadata is stored in the API, any integration that implements the container image registry API must provide its own storage for the raw manifest data, image config, and layer contents.</p> <p>Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).</p> |
| <b>kind</b>     | <b>string</b>        | <p>Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds</a></p>                                                                                                                                                                                                                                                                                                                                                                                  |
| <b>metadata</b> | <b>ObjectMeta_v2</b> | <p>metadata is the standard object's metadata. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata</a></p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |

| Property      | Type          | Description                                                                                                                                                                   |
|---------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>spec</b>   | <b>object</b> | TagReference specifies optional annotations for images using this tag and an optional reference to an ImageStreamTag, ImageStreamImage, or DockerImage this tag should track. |
| <b>status</b> | <b>object</b> | NamedTagEventList relates a tag to its image history.                                                                                                                         |

### 10.1.1. .image

#### Description

Image is an immutable representation of a container image and metadata at a point in time. Images are named by taking a hash of their contents (metadata and content) and any change in format, content, or metadata results in a new name. The images resource is primarily for use by cluster administrators and integrations like the cluster image registry – end users instead access images via the imagestreamtags or imagestreamimages resources. While image metadata is stored in the API, any integration that implements the container image registry API must provide its own storage for the raw manifest data, image config, and layer contents.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

#### Type

**object**

| Property                 | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                   |
|--------------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>apiVersion</b>        | <b>string</b> | APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources</a> |
| <b>dockerImageConfig</b> | <b>string</b> | DockerImageConfig is a JSON blob that the runtime uses to set up the container. This is a part of manifest schema v2. Will not be set when the image represents a manifest list.                                                                                                                                                                                                                              |

| Property                            | Type                  | Description                                                                                                                                                                 |
|-------------------------------------|-----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dockerImageLayers</b>            | <b>array</b>          | DockerImageLayers represents the layers in the image. May not be set if the image does not define that data or if the image represents a manifest list.                     |
| <b>dockerImageLayers[]</b>          | <b>object</b>         | ImageLayer represents a single layer of the image. Some images may have multiple layers. Some may have none.                                                                |
| <b>dockerImageManifest</b>          | <b>string</b>         | DockerImageManifest is the raw JSON of the manifest                                                                                                                         |
| <b>dockerImageManifestMediaType</b> | <b>string</b>         | DockerImageManifestMediaType specifies the mediaType of manifest. This is a part of manifest schema v2.                                                                     |
| <b>dockerImageManifests</b>         | <b>array</b>          | DockerImageManifests holds information about sub-manifests when the image represents a manifest list. When this field is present, no DockerImageLayers should be specified. |
| <b>dockerImageManifests[]</b>       | <b>object</b>         | ImageManifest represents sub-manifests of a manifest list. The Digest field points to a regular Image object.                                                               |
| <b>dockerImageMetadata</b>          | <b>RawExtension</b>   | DockerImageMetadata contains metadata about this image                                                                                                                      |
| <b>dockerImageMetadataVersion</b>   | <b>string</b>         | DockerImageMetadataVersion conveys the version of the object, which if empty defaults to "1.0"                                                                              |
| <b>dockerImageReference</b>         | <b>string</b>         | DockerImageReference is the string that can be used to pull this image.                                                                                                     |
| <b>dockerImageSignatures</b>        | <b>array (string)</b> | DockerImageSignatures provides the signatures as opaque blobs. This is a part of manifest schema v1.                                                                        |

| Property            | Type                 | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|---------------------|----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>kind</b>         | <b>string</b>        | Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds</a>                                                                                                                                                                  |
| <b>metadata</b>     | <b>ObjectMeta_v2</b> | metadata is the standard object's metadata. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata</a>                                                                                                                                                                                                                                                                                                                    |
| <b>signatures</b>   | <b>array</b>         | Signatures holds all signatures of the image.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| <b>signatures[]</b> | <b>object</b>        | <p>ImageSignature holds a signature of an image. It allows to verify image identity and possibly other claims as long as the signature is trusted. Based on this information it is possible to restrict runnable images to those matching cluster-wide policy. Mandatory fields should be parsed by clients doing image verification. The others are parsed from signature's content by the server. They serve just an informative purpose.</p> <p>Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).</p> |

### 10.1.2. .image.dockerImageLayers

#### Description

DockerImageLayers represents the layers in the image. May not be set if the image does not define that data or if the image represents a manifest list.

#### Type

**array**

### 10.1.3. .image.dockerImageLayers[]

#### Description

ImageLayer represents a single layer of the image. Some images may have multiple layers. Some may have none.

#### Type

**object**

#### Required

- **name**
- **size**
- **mediaType**

| Property         | Type           | Description                                                    |
|------------------|----------------|----------------------------------------------------------------|
| <b>mediaType</b> | <b>string</b>  | MediaType of the referenced object.                            |
| <b>name</b>      | <b>string</b>  | Name of the layer as defined by the underlying store.          |
| <b>size</b>      | <b>integer</b> | Size of the layer in bytes as defined by the underlying store. |

### 10.1.4. .image.dockerImageManifests

#### Description

DockerImageManifests holds information about sub-manifests when the image represents a manifest list. When this field is present, no DockerImageLayers should be specified.

#### Type

**array**

### 10.1.5. .image.dockerImageManifests[]

#### Description

ImageManifest represents sub-manifests of a manifest list. The Digest field points to a regular Image object.

#### Type

**object**

#### Required

- **digest**
- **mediaType**
- **manifestSize**

- **architecture**
- **os**

| Property            | Type           | Description                                                                                                                                                                                                               |
|---------------------|----------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>architecture</b> | <b>string</b>  | Architecture specifies the supported CPU architecture, for example <b>amd64</b> or <b>ppc64le</b> .                                                                                                                       |
| <b>digest</b>       | <b>string</b>  | Digest is the unique identifier for the manifest. It refers to an Image object.                                                                                                                                           |
| <b>manifestSize</b> | <b>integer</b> | ManifestSize represents the size of the raw object contents, in bytes.                                                                                                                                                    |
| <b>mediaType</b>    | <b>string</b>  | MediaType defines the type of the manifest, possible values are application/vnd.oci.image.manifest.v1+json, application/vnd.docker.distribution.manifest.v2+json or application/vnd.docker.distribution.manifest.v1+json. |
| <b>os</b>           | <b>string</b>  | OS specifies the operating system, for example <b>linux</b> .                                                                                                                                                             |
| <b>variant</b>      | <b>string</b>  | Variant is an optional field representing a variant of the CPU, for example v6 to specify a particular CPU variant of the ARM CPU.                                                                                        |

### 10.1.6. .image.signatures

#### Description

Signatures holds all signatures of the image.

#### Type

**array**

### 10.1.7. .image.signatures[]

#### Description

ImageSignature holds a signature of an image. It allows to verify image identity and possibly other claims as long as the signature is trusted. Based on this information it is possible to restrict runnable images to those matching cluster-wide policy. Mandatory fields should be parsed by clients doing

image verification. The others are parsed from signature's content by the server. They serve just an informative purpose.

Compatibility level 1: Stable within a major release for a minimum of 12 months or 3 minor releases (whichever is longer).

## Type

**object**

## Required

- **type**
- **content**

| Property             | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                   |
|----------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>apiVersion</b>    | <b>string</b> | APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources</a> |
| <b>conditions</b>    | <b>array</b>  | Conditions represent the latest available observations of a signature's current state.                                                                                                                                                                                                                                                                                                                        |
| <b>conditions[]</b>  | <b>object</b> | SignatureCondition describes an image signature condition of particular kind at particular probe time.                                                                                                                                                                                                                                                                                                        |
| <b>content</b>       | <b>string</b> | Required: An opaque binary string which is an image's signature.                                                                                                                                                                                                                                                                                                                                              |
| <b>created</b>       | <b>Time</b>   | If specified, it is the time of signature's creation.                                                                                                                                                                                                                                                                                                                                                         |
| <b>imageIdentity</b> | <b>string</b> | A human readable string representing image's identity. It could be a product name and version, or an image pull spec (e.g. "registry.access.redhat.com/rhel7/rhel:7.2").                                                                                                                                                                                                                                      |

| Property            | Type                   | Description                                                                                                                                                                                                                                                                                                                                                                                                      |
|---------------------|------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>issuedBy</b>     | <b>object</b>          | SignatureIssuer holds information about an issuer of signing certificate or key.                                                                                                                                                                                                                                                                                                                                 |
| <b>issuedTo</b>     | <b>object</b>          | SignatureSubject holds information about a person or entity who created the signature.                                                                                                                                                                                                                                                                                                                           |
| <b>kind</b>         | <b>string</b>          | Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds</a> |
| <b>metadata</b>     | <b>ObjectMeta_v2</b>   | metadata is the standard object's metadata. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#metadata</a>                                                                                                                                                   |
| <b>signedClaims</b> | <b>object (string)</b> | Contains claims from the signature.                                                                                                                                                                                                                                                                                                                                                                              |
| <b>type</b>         | <b>string</b>          | Required: Describes a type of stored blob.                                                                                                                                                                                                                                                                                                                                                                       |

### 10.1.8. .image.signatures[].conditions

#### Description

Conditions represent the latest available observations of a signature's current state.

#### Type

**array**

### 10.1.9. .image.signatures[].conditions[]

#### Description

SignatureCondition describes an image signature condition of particular kind at particular probe time.

#### Type

**object**

**Required**

- **type**
- **status**

| Property                  | Type          | Description                                                      |
|---------------------------|---------------|------------------------------------------------------------------|
| <b>lastProbeTime</b>      | <b>Time</b>   | Last time the condition was checked.                             |
| <b>lastTransitionTime</b> | <b>Time</b>   | Last time the condition transit from one status to another.      |
| <b>message</b>            | <b>string</b> | Human readable message indicating details about last transition. |
| <b>reason</b>             | <b>string</b> | (brief) reason for the condition's last transition.              |
| <b>status</b>             | <b>string</b> | Status of the condition, one of True, False, Unknown.            |
| <b>type</b>               | <b>string</b> | Type of signature condition, Complete or Failed.                 |

**10.1.10. .image.signatures[].issuedBy****Description**

SignatureIssuer holds information about an issuer of signing certificate or key.

**Type**

**object**

| Property            | Type          | Description                                   |
|---------------------|---------------|-----------------------------------------------|
| <b>commonName</b>   | <b>string</b> | Common name (e.g. openshift-signing-service). |
| <b>organization</b> | <b>string</b> | Organization name.                            |

**10.1.11. .image.signatures[].issuedTo****Description**

SignatureSubject holds information about a person or entity who created the signature.

**Type**

**object**

**Required**

- **publicKeyID**

| Property            | Type          | Description                                                                                                                                                                                                       |
|---------------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>commonName</b>   | <b>string</b> | Common name (e.g. openshift-signing-service).                                                                                                                                                                     |
| <b>organization</b> | <b>string</b> | Organization name.                                                                                                                                                                                                |
| <b>publicKeyID</b>  | <b>string</b> | If present, it is a human readable key id of public key belonging to the subject used to verify image signature. It should contain at least 64 lowest bits of public key's fingerprint (e.g. 0x685ebe62bf278440). |

**10.1.12. .spec****Description**

TagReference specifies optional annotations for images using this tag and an optional reference to an ImageStreamTag, ImageStreamImage, or DockerImage this tag should track.

**Type**

**object**

**Required**

- **name**

| Property           | Type                   | Description                                                                                                                                                                                                                              |
|--------------------|------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>annotations</b> | <b>object (string)</b> | Optional; if specified, annotations that are applied to images retrieved via ImageStreamTags.                                                                                                                                            |
| <b>from</b>        | <b>ObjectReference</b> | Optional; if specified, a reference to another image that this tag should point to. Valid values are ImageStreamTag, ImageStreamImage, and DockerImage. ImageStreamTag references can only reference a tag within this same ImageStream. |

| Property               | Type           | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|------------------------|----------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>generation</b>      | <b>integer</b> | Generation is a counter that tracks mutations to the spec tag (user intent). When a tag reference is changed the generation is set to match the current stream generation (which is incremented every time spec is changed). Other processes in the system like the image importer observe that the generation of spec tag is newer than the generation recorded in the status and use that as a trigger to import the newest remote tag. To trigger a new import, clients may set this value to zero which will reset the generation to the latest stream generation. Legacy clients will send this value as nil which will be merged with the current tag generation. |
| <b>importPolicy</b>    | <b>object</b>  | TagImportPolicy controls how images related to this tag will be imported.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| <b>name</b>            | <b>string</b>  | Name of the tag                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| <b>reference</b>       | <b>boolean</b> | Reference states if the tag will be imported. Default value is false, which means the tag will be imported.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| <b>referencePolicy</b> | <b>object</b>  | TagReferencePolicy describes how pull-specs for images in this image stream tag are generated when image change triggers in deployment configs or builds are resolved. This allows the image stream author to control how images are accessed.                                                                                                                                                                                                                                                                                                                                                                                                                          |

### 10.1.13. .spec.importPolicy

#### Description

TagImportPolicy controls how images related to this tag will be imported.

#### Type

**object**

| Property          | Type           | Description                                                                                                             |
|-------------------|----------------|-------------------------------------------------------------------------------------------------------------------------|
| <b>importMode</b> | <b>string</b>  | ImportMode describes how to import an image manifest.                                                                   |
| <b>insecure</b>   | <b>boolean</b> | Insecure is true if the server may bypass certificate verification or connect directly over HTTP during image import.   |
| <b>scheduled</b>  | <b>boolean</b> | Scheduled indicates to the server that this tag should be periodically checked to ensure it is up to date, and imported |

#### 10.1.14. .spec.referencePolicy

##### Description

TagReferencePolicy describes how pull-specs for images in this image stream tag are generated when image change triggers in deployment configs or builds are resolved. This allows the image stream author to control how images are accessed.

##### Type

**object**

##### Required

- **type**

| Property | Type | Description |
|----------|------|-------------|
|----------|------|-------------|

| Property    | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|-------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>type</b> | <b>string</b> | Type determines how the image pull spec should be transformed when the image stream tag is used in deployment config triggers or new builds. The default value is <b>Source</b> , indicating the original location of the image should be used (if imported). The user may also specify <b>Local</b> , indicating that the pull spec should point to the integrated container image registry and leverage the registry's ability to proxy the pull to an upstream registry. <b>Local</b> allows the credentials used to pull this image to be managed from the image stream's namespace, so others on the platform can access a remote image but have no access to the remote secret. It also allows the image layers to be mirrored into the local registry which the images can still be pulled even if the upstream registry is unavailable. |

### 10.1.15. .status

#### Description

NamedTagEventList relates a tag to its image history.

#### Type

**object**

#### Required

- **tag**
- **items**

| Property          | Type         | Description                                                            |
|-------------------|--------------|------------------------------------------------------------------------|
| <b>conditions</b> | <b>array</b> | Conditions is an array of conditions that apply to the tag event list. |

| Property            | Type          | Description                                                                                        |
|---------------------|---------------|----------------------------------------------------------------------------------------------------|
| <b>conditions[]</b> | <b>object</b> | TagEventCondition contains condition information for a tag event.                                  |
| <b>items</b>        | <b>array</b>  | Standard object's metadata.                                                                        |
| <b>items[]</b>      | <b>object</b> | TagEvent is used by ImageStreamStatus to keep a historical record of images associated with a tag. |
| <b>tag</b>          | <b>string</b> | Tag is the tag for which the history is recorded                                                   |

### 10.1.16. .status.conditions

#### Description

Conditions is an array of conditions that apply to the tag event list.

#### Type

**array**

### 10.1.17. .status.conditions[]

#### Description

TagEventCondition contains condition information for a tag event.

#### Type

**object**

#### Required

- **type**
- **status**
- **generation**

| Property                  | Type           | Description                                                                           |
|---------------------------|----------------|---------------------------------------------------------------------------------------|
| <b>generation</b>         | <b>integer</b> | Generation is the spec tag generation that this status corresponds to                 |
| <b>lastTransitionTime</b> | <b>Time</b>    | LastTransitionTime is the time the condition transitioned from one status to another. |

| Property       | Type          | Description                                                                                         |
|----------------|---------------|-----------------------------------------------------------------------------------------------------|
| <b>message</b> | <b>string</b> | Message is a human readable description of the details about last transition, complementing reason. |
| <b>reason</b>  | <b>string</b> | Reason is a brief machine readable explanation for the condition's last transition.                 |
| <b>status</b>  | <b>string</b> | Status of the condition, one of True, False, Unknown.                                               |
| <b>type</b>    | <b>string</b> | Type of tag event condition, currently only ImportSuccess                                           |

### 10.1.18. .status.items

#### Description

Standard object's metadata.

#### Type

**array**

### 10.1.19. .status.items[]

#### Description

TagEvent is used by ImageStreamStatus to keep a historical record of images associated with a tag.

#### Type

**object**

#### Required

- **created**
- **dockerImageReference**
- **image**
- **generation**

| Property       | Type        | Description                                     |
|----------------|-------------|-------------------------------------------------|
| <b>created</b> | <b>Time</b> | Created holds the time the TagEvent was created |

| Property                    | Type           | Description                                                                   |
|-----------------------------|----------------|-------------------------------------------------------------------------------|
| <b>dockerImageReference</b> | <b>string</b>  | DockerImageReference is the string that can be used to pull this image        |
| <b>generation</b>           | <b>integer</b> | Generation is the spec tag generation that resulted in this tag being updated |
| <b>image</b>                | <b>string</b>  | Image is the image                                                            |

## 10.2. API ENDPOINTS

The following API endpoints are available:

- **/apis/image.openshift.io/v1/imagetags**
  - **GET**: list objects of kind ImageTag
- **/apis/image.openshift.io/v1/namespaces/{namespace}/imagetags**
  - **GET**: list objects of kind ImageTag
  - **POST**: create an ImageTag
- **/apis/image.openshift.io/v1/namespaces/{namespace}/imagetags/{name}**
  - **DELETE**: delete an ImageTag
  - **GET**: read the specified ImageTag
  - **PATCH**: partially update the specified ImageTag
  - **PUT**: replace the specified ImageTag

### 10.2.1. /apis/image.openshift.io/v1/imagetags

HTTP method

**GET**

Description

list objects of kind ImageTag

Table 10.1. HTTP responses

| HTTP code | Reponse body                        |
|-----------|-------------------------------------|
| 200 - OK  | <a href="#">ImageTagList</a> schema |

| HTTP code          | Reponse body |
|--------------------|--------------|
| 401 - Unauthorized | Empty        |

### 10.2.2. /apis/image.openshift.io/v1/namespaces/{namespace}/imagetags

HTTP method

**GET**

Description

list objects of kind ImageTag

Table 10.2. HTTP responses

| HTTP code          | Reponse body                        |
|--------------------|-------------------------------------|
| 200 - OK           | <a href="#">ImageTagList</a> schema |
| 401 - Unauthorized | Empty                               |

HTTP method

**POST**

Description

create an ImageTag

Table 10.3. Query parameters

| Parameter     | Type          | Description                                                                                                                                                                                                                                              |
|---------------|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dryRun</b> | <b>string</b> | When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed |

| Parameter              | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|------------------------|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>fieldValidation</b> | <b>string</b> | fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: <ul style="list-style-type: none"> <li>- Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23.</li> <li>- Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+</li> <li>- Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered.</li> </ul> |

Table 10.4. Body parameters

| Parameter   | Type                   | Description |
|-------------|------------------------|-------------|
| <b>body</b> | <b>ImageTag</b> schema |             |

Table 10.5. HTTP responses

| HTTP code          | Reponse body           |
|--------------------|------------------------|
| 200 - OK           | <b>ImageTag</b> schema |
| 201 - Created      | <b>ImageTag</b> schema |
| 202 - Accepted     | <b>ImageTag</b> schema |
| 401 - Unauthorized | Empty                  |

### 10.2.3. /apis/image.openshift.io/v1/namespaces/{namespace}/imagetags/{name}

Table 10.6. Global path parameters

| Parameter   | Type          | Description          |
|-------------|---------------|----------------------|
| <b>name</b> | <b>string</b> | name of the ImageTag |

**HTTP method****DELETE****Description**

delete an ImageTag

**Table 10.7. Query parameters**

| Parameter     | Type          | Description                                                                                                                                                                                                                                              |
|---------------|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dryRun</b> | <b>string</b> | When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed |

**Table 10.8. HTTP responses**

| HTTP code          | Response body                    |
|--------------------|----------------------------------|
| 200 - OK           | <a href="#">Status_v5</a> schema |
| 202 - Accepted     | <a href="#">Status_v5</a> schema |
| 401 - Unauthorized | Empty                            |

**HTTP method****GET****Description**

read the specified ImageTag

**Table 10.9. HTTP responses**

| HTTP code          | Response body                   |
|--------------------|---------------------------------|
| 200 - OK           | <a href="#">ImageTag</a> schema |
| 401 - Unauthorized | Empty                           |

**HTTP method****PATCH****Description**

partially update the specified ImageTag

**Table 10.10. Query parameters**

| Parameter              | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|------------------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dryRun</b>          | <b>string</b> | When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <b>fieldValidation</b> | <b>string</b> | fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered. |

Table 10.11. HTTP responses

| HTTP code          | Response body                   |
|--------------------|---------------------------------|
| 200 - OK           | <a href="#">ImageTag</a> schema |
| 201 - Created      | <a href="#">ImageTag</a> schema |
| 401 - Unauthorized | Empty                           |

**HTTP method****PUT****Description**

replace the specified ImageTag

Table 10.12. Query parameters

| Parameter | Type | Description |
|-----------|------|-------------|
|-----------|------|-------------|

| Parameter              | Type          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|------------------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>dryRun</b>          | <b>string</b> | When present, indicates that modifications should not be persisted. An invalid or unrecognized dryRun directive will result in an error response and no further processing of the request. Valid values are: - All: all dry run stages will be processed                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <b>fieldValidation</b> | <b>string</b> | fieldValidation instructs the server on how to handle objects in the request (POST/PUT/PATCH) containing unknown or duplicate fields. Valid values are: - Ignore: This will ignore any unknown fields that are silently dropped from the object, and will ignore all but the last duplicate field that the decoder encounters. This is the default behavior prior to v1.23. - Warn: This will send a warning via the standard warning response header for each unknown field that is dropped from the object, and for each duplicate field that is encountered. The request will still succeed if there are no other errors, and will only persist the last of any duplicate fields. This is the default in v1.23+ - Strict: This will fail the request with a BadRequest error if any unknown fields would be dropped from the object, or if any duplicate fields are present. The error returned from the server will contain all unknown and duplicate fields encountered. |

Table 10.13. Body parameters

| Parameter   | Type                   | Description |
|-------------|------------------------|-------------|
| <b>body</b> | <b>ImageTag</b> schema |             |

Table 10.14. HTTP responses

| HTTP code          | Response body          |
|--------------------|------------------------|
| 200 - OK           | <b>ImageTag</b> schema |
| 201 - Created      | <b>ImageTag</b> schema |
| 401 - Unauthorized | Empty                  |

## CHAPTER 11. SECRETLIST [IMAGE.OPENSIFT.IO/V1]

### Description

SecretList is a list of Secret.

### Type

**object**

### Required

- **items**

## 11.1. SPECIFICATION

| Property          | Type                  | Description                                                                                                                                                                                                                                                                                                                                                                                                      |
|-------------------|-----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>apiVersion</b> | <b>string</b>         | APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources</a>    |
| <b>items</b>      | <b>array (Secret)</b> | Items is a list of secret objects. More info: <a href="https://kubernetes.io/docs/concepts/configuration/secret">https://kubernetes.io/docs/concepts/configuration/secret</a>                                                                                                                                                                                                                                    |
| <b>kind</b>       | <b>string</b>         | Kind is a string value representing the REST resource this object represents. Servers may infer this from the endpoint the client submits requests to. Cannot be updated. In CamelCase. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds</a> |
| <b>metadata</b>   | <b>ListMeta</b>       | Standard list metadata. More info: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-kinds</a>                                                                                                                                                                 |

## 11.2. API ENDPOINTS

The following API endpoints are available:

- **/apis/image.openshift.io/v1/namespaces/{namespace}/imagestreams/{name}/secrets**
  - **GET**: read secrets of the specified ImageStream

### 11.2.1. /apis/image.openshift.io/v1/namespaces/{namespace}/imagestreams/{name}/

Table 11.1. Global path parameters

| Parameter   | Type          | Description            |
|-------------|---------------|------------------------|
| <b>name</b> | <b>string</b> | name of the SecretList |

#### HTTP method

**GET**

#### Description

read secrets of the specified ImageStream

Table 11.2. HTTP responses

| HTTP code          | Response body            |
|--------------------|--------------------------|
| 200 - OK           | <b>SecretList</b> schema |
| 401 - Unauthorized | Empty                    |