



Qualys Container Security v1.x

API Release Notes

Version 1.10

May 10, 2021 (Updated June 3, 2021)

Qualys Container Security API gives you many ways to integrate your programs and API calls with Qualys capabilities.

What's New

[New Activation Key API](#)

[New Vulnerability Report APIs](#)

[Delete APIs Will Now Take UUIDs or Filter as Input Parameters](#)

[Collection of Kubernetes Cluster Attributes](#)

Qualys API URL

Container Security supports both API server URLs and API gateway URLs for API requests.

The Qualys API server or gateway URL you should use for API requests depends on the Qualys platform where your account is located.

[Click here to identify your Qualys platform and get the API URL](#)

This documentation uses the API URL for Qualys US Platform 2 (<https://gateway.qg2.apps.qualys.com>) in sample API requests. If you're on another platform, please replace this URL with the appropriate server URL for your account.

New Activation Key API

This release introduces a new API endpoint to give customers a way to automate the provisioning of sensors in their Devops pipeline. Use the new API to get the keys required to activate a new sensor. The API response includes your customerId, activationId, and platformUrl.

API Objective	Operator	API Path
Get activation key	GET	/csapi/v1.3/activationkey

Get activation key

/csapi/v1.3/activationkey

[GET]

API request:

```
curl -X GET
'https://gateway.qg2.apps.qualys.com/csapi/v1.3/activationkey' --header
'Authorization: Bearer <token>'
```

Response:

```
{
  "customerId": "198cc974-1e44-cb6c-806e-f78f6221cb0d",
  "activationId": "192cc974-1e22-cb6c-806e-f78f6441cb0d",
  "platformUrl": "https://qualysguard.qualys.com/"
}
```

New Vulnerability Report APIs

Now you can create and manage Image and Container Vulnerability Reports using the API. We've introduced several new API endpoints that will allow you to create new report requests, fetch a list of reports, download a report, and delete reports.

API Objective	Operator	API Path
Create a report request	POST	/csapi/v1.3/reports
Fetch a list of reports	GET	/csapi/v1.3/reports
Download a report	GET	/csapi/v1.3/reports/{reportUuid}/download
Delete reports	DELETE	/csapi/v1.3/reports

Jump to a section to see API samples:

[Create a report request](#)

[Fetch a list of reports](#)

[Download a report](#)

[Delete reports](#)

Create a report request

/csapi/v1.3/reports

[POST]

Use this API to create a new report request.

Input Parameters:

Parameter	Description
name	(Required) Specify a title for your report (up to 150 characters).
description	Specify a description for your report (up to 250 characters).
templateName	(Required) Specify the template for the report you want to create. Valid values are: CS_IMAGE_VULNERABILITY or CS_CONTAINER_VULNERABILITY

filter	Filter the images or containers list for the report by providing a query using Qualys syntax. Refer to the “How to Search” topic in the online help for assistance with creating your query.
displayColumns	Specify the columns to include in the report. Multiple columns should be comma-separated. When unspecified, ALL report columns will be included. When an empty value is provided, only default columns will be included. When the template CS_IMAGE_VULNERABILITY is used, you can include any of these columns: repo, imageId, sha, uuid, created, updated, qid, title, severity, cveids, vendorReference, cvssBase, cvssTemporal, cvss3Base, cvss3Temporal, threat, impact, solution, exploitability, associatedMalwares, category, software, result. Default columns are: qid, imageId. When the template CS_CONTAINER_VULNERABILITY is used, you can include any of these columns: name, containerId, uuid, imageId, created, hostName, hostIp, state, stateChanged, updated, qid, title, severity, cveids, vendorReference, cvssBase, cvssTemporal, cvss3Base, cvss3Temporal, threat, impact, solution, exploitability, associatedMalwares, category, software, result. Default columns are: qid, containerId.

API request:

```
curl -X POST 'https://gateway.qg2.apps.qualys.com/csapi/v1.3/reports' -d
'{"description":"Demo Report","name":"My Container
Report","templateName":"CS_CONTAINER_VULNERABILITY",
"filter":"status:running","displayColumns":["containerid","uuid","qid"]"
}' --header 'Authorization: Bearer <token>'
```

Response:

```
response code 200
{
  "reportUuid": "037570f0-0193-11ea-9327-8fbbd2104c9c"
}
```

Fetch a list of reports

/csapi/v1.3/reports

[GET]

Use this API to fetch a list of reports from your account.

Input Parameters:

Parameter	Description
filter	Filter the reports list by providing a query using Qualys syntax. Currently, you can only filter by reportName. Refer to the “How to Search” topic in the online help for assistance with creating your query.
pageSize	(Required) The number of records per page to be included in the response. The default value is 50.
pageNumber	(Required) The page to be returned. Page numbers start with 1. The default value is 1.
sort	Sort the results using a Qualys token. You can sort by status or reportName only. Default value is status:desc. Refer to the “Sortable tokens” topic in the online help for more information.

API request:

```
curl -X GET
'https://gateway.qg2.apps.qualys.com/csapi/v1.3/reports?pageNumber=1&page
Size=50' --header 'Authorization: Bearer <token>'
```

Response:

```
{
  "data": [
    {
      "reportUuid": "4738d060-5b02-11eb-a819-9b3f1886eb36",
      "createdAt": "2021-01-20T09:31:27.000Z",
      "reportName": "sample-image-report",
      "description": null,
      "fileFormat": "csv",
      "templateName": "CS_IMAGE_VULNERABILITY",
      "status": "FAILED",
      "reportType": "ON_DEMAND",
      "filter": null,
      "displayColumns": ["ALL"]
    },
    {
      "reportUuid": "42699db0-9eb1-11eb-bbb6-833197dc0c5b",
```

```

    "createdAt": "2021-04-16T12:42:52.000Z",
    "reportName": "sample-container-report",
    "description": null,
    "fileFormat": "csv",
    "templateName": "CS_CONTAINER_VULNERABILITY",
    "status": "COMPLETED",
    "reportType": "ON_DEMAND",
    "filter": "",
    "displayColumns": [
      "name",
      "containerId",
      "imageId",
      "qid",
      "cveids"
    ]
  }
],
"count": 2
}

```

Download a report

/csapi/v1.3/reports/{reportUuid}/download

[GET]

Use this API to download a specific report.

Input Parameters:

Parameter	Description
reportUuid	(Required) The UUID for the report you want to download.

API request:

```

curl -X GET
'https://gateway.qg2.apps.qualys.com/csapi/v1.3/reports/903db2f0-e450-
11e9-9cdc-8362fe1d0c3a/download' --header 'Authorization: Bearer <token>'

```

Response:

```

response code 200

```

Delete reports

/csapi/v1.3/reports

[DELETE]

Use this API to delete one or more reports from your account.

Input Parameters:

Parameter	Description
reportUuids	(Required) One or more report UUIDs for the reports you want to delete. When specifying multiple reports to delete, enter them in this way: reportUuids=value1&reportUuids=value2&reportUuids=value3, and so on.

API request:

```
curl -X DELETE
'https://gateway.qg2.apps.qualys.com/csapi/v1.3/reports?reportUuids=a6476
d20-c69f-11ea-8700-3d057ac643a' --header 'Authorization: Bearer <token>'
```

Response:

```
response code 200
{
  "data": null,
  "message": "Report Successfully deleted."
}
```

Delete APIs Will Now Take UUIDs or Filter as Input Parameters

Delete APIs for Images, Containers and Sensors have been updated to allow users to specify input parameters like UUIDs and filters. Input parameters were not accepted prior to this release. This change is applicable for v1.2 and v1.3 of the API endpoints. Note that the Delete APIs will continue to accept the old method of providing request body without parameters also.

Please see sections below for input parameters and samples.

[Delete Images](#)

[Delete Containers](#)

[Delete Sensors](#)

Delete Images

/v1.3/images

[DELETE]

Images with active containers (CREATED, RUNNING, STOPPED, PAUSED) associated with them cannot be deleted.

Input Parameters:

Parameter	Description
imageIds	One or more image UUIDs for the images you want to delete. When specifying multiple images in the same request, enter them in this way: imageIds=value1&imageIds=value2&imageIds=value3, and so on.
filter	Filter the images list by providing a query using Qualys syntax. Refer to the “How to Search” topic in the online help for assistance with creating your query.

Sample 1 - Delete single image using image UUID

In this sample, we'll delete a single image by specifying the image UUID.

API request:

```
curl -X DELETE  
'https://gateway.qg2.apps.qualys.com/csapi/v1.3/images?imageIds=8b261e4e-  
47f3-3b6a-a5a7-668ff0d6e3eb' --header 'Authorization: Bearer <token>'
```


Response:

```
{  
  "deletionJobId": "ee295423-af59-4f5c-a4a1-7cb035dae61b"  
}
```

Sample 2 - Delete multiple images using image UUIDs

In this sample, we'll delete 2 images in the same request. Specify multiple images by entering `imageIds=value1&imageIds=value2`, and so on.

API request:

```
curl -X DELETE  
'https://gateway.qg2.apps.qualys.com/csapi/v1.3/images?imageIds=8b261e4e-  
47f3-3b6a-a5a7-668ff0d6e3eb&imageIds=9b251e3e-52f3-2b6a-a6a7-  
678ff0d5e2eb' --header 'Authorization: Bearer <token>'
```

Response:

```
{  
  "deletionJobId": "1b54a117-b413-4aa8-8511-61860487619c"  
}
```

Sample 3 - Delete images using a filter

In this sample, we'll delete images based on the filter parameter.

API request:

```
curl -X DELETE  
'https://gateway.qg2.apps.qualys.com/csapi/v1.3/images?filter=imageId%3Ae  
c3b4a64d481' --header 'Authorization: Bearer <token>'
```

Response:

```
{  
  "deletionJobId": "ee295423-af59-4f5c-a4a1-7cb035dae61b"  
}
```

Delete Containers

/csapi/v1.3/containers

[DELETE]

Use this API to delete one or more containers from your account.

Input Parameters:

Parameter	Description
containerIds	One or more container UUIDs for the containers you want to delete. When specifying multiple containers in the same request, enter them in this way: containerIds=value1&containerIds=value2&containerIds=value3, and so on.
filter	Filter the containers list by providing a query using Qualys syntax. Refer to the “How to Search” topic in the online help for assistance with creating your query.

Sample 1 - Delete single container using container UUID

In this sample, we’ll delete a single container by specifying the container UUID.

API request:

```
curl -X DELETE
'https://gateway.qg2.apps.qualys.com/csapi/v1.3/containers?containerIds=7
7161d39-386c-35dd-85b2-a80bd86111b6' --header 'Authorization: Bearer
<token>'
```

Response:

```
{
  "deletionJobId": "951dca63-254b-4f5c-ab76-7671dd8f1528"
}
```

Sample 2 - Delete multiple containers using container UUIDs

In this sample, we’ll delete 2 containers in the same request. Specify multiple containers by entering containerIds=value1&containerIds=value2, and so on.

API request:

```
curl -X DELETE
'https://gateway.qg2.apps.qualys.com/csapi/v1.3/containers?containerIds=6
8cc933e-4994-3d9b-8232-b1c78b3b3c9d&containerIds=4e7d40ce-6ddc-3fb7-ab55-
e0b48a60208a' --header 'Authorization: Bearer <token>'
```

Response:

```
{  
  "deletionJobId": "2b38a127-b413-4aa7-8421-61760487722c"  
}
```

Sample 3 - Delete containers using a filter

In this sample, we'll delete containers based on the filter parameter.

API request:

```
curl -X DELETE  
'https://gateway.qg2.apps.qualys.com/csapi/v1.3/containers?filter=contain  
erId%3A194c52ca4d31' --header 'Authorization: Bearer <token>'
```

Response:

```
{  
  "deletionJobId": "8a47532f-4f22-46e5-b772-58f9664173fe"  
}
```

Delete Sensors

/v1.3/sensors

[DELETE]

You can only delete sensors with UNKNOWN status.

Input Parameters:

Parameter	Description
sensorIds	One or more sensor UUIDs for the sensors you want to delete. When specifying multiple sensors in the same request, enter them in this way: sensorIds=value1&sensorIds=value2&sensorIds=value3, and so on.
filter	Filter the sensors list by providing a query using Qualys syntax. Refer to the “How to Search” topic in the online help for assistance with creating your query.

Sample 1 - Delete single sensor using sensor UUID

In this sample, we will delete a single sensor by specifying the sensor UUID.

API request:

```
curl -X DELETE
'https://gateway.qg2.apps.qualys.com/csapi/v1.3/sensors?sensorIds=dba5571
8-736b-4bac-ae9a-38bb59643b45' --header 'Authorization: Bearer <token>'
```

Response:

```
{
  "deletionJobId": "8b04c56a-33ad-44ba-a128-fc643b0af3c7"
}
```

Sample 2 - Delete multiple sensors using sensor UUIDs

In this sample, we'll delete 2 sensors in the same request. Specify multiple sensors by entering sensorIds=value1&sensorIds=value2, and so on.

API request:

```
curl -X DELETE
'https://gateway.qg2.apps.qualys.com/csapi/v1.3/sensors?sensorIds=dba5571
8-736b-4bac-ae9a-38bb59643b45&sensorIds=bcd55718-736b-3bac-ae7a-
34bb59644b32' --header 'Authorization: Bearer <token>'
```

Response:

```
{
  "deletionJobId": "6e04c56a-42ad-44ba-a678-fc642b0af3c9"
}
```

Sample 3 - Delete sensors using a filter

In this sample, we'll delete sensors based on the filter parameter.

API request:

```
curl -X DELETE
'https://gateway.qg2.apps.qualys.com/csapi/v1.3/sensors?filter=sensorId%3
A4f97332e3a23' --header 'Authorization: Bearer <token>'
```

Response:

```
{
  "deletionJobId": "7a93e37e-b2c6-40ac-8f18-70f87fcf3957"
}
```

Collection of Kubernetes Cluster Attributes

In this release we have added collection of Kubernetes cluster attributes and made this information searchable in the UI. Kubernetes cluster attributes include node details, pod details (name, uuid, namespace, labels), controller details (name, uuid, type) and more.

Important - Kubernetes attributes will only be processed for containers discovered after the Container Security version 1.10 release. Kubernetes attributes are collected as part of container inspect processing when containers are discovered for the first time. To fetch Kubernetes cluster attributes for an existing deployment in Kubernetes, you will have to “rollout restart” the existing deployment, which will create new containers and this will start the container inspect processing. Kubernetes attributes will get collected for the newly created containers on Kubernetes clusters.

Use the following command for the “rollout restart”:

```
kubectl rollout restart deployment <deployment-name> -n <namespace>
```

API Changes

We updated the following Container Security APIs to append the Kubernetes cluster attributes to the API output. Jump to a section below to see a sample.

- [Fetch container details](#)
- [Fetch a detailed container list](#)
- [Fetch sensor details](#)

Kubernetes Cluster Attributes

You’ll see these attributes in the API output:

- Cluster type (Kubernetes)
- Cluster version
- Project name (collected for projects in Google Cloud Platform)
- Node name and flag indicating whether the node is the master node
- Pod name
- Pod UUID
- Pod namespace
- Pod labels (key and value pairs)
- Controller name
- Controller UUID
- Controller type (e.g. DaemonSet, Deployment, ReplicaSet, etc)

Fetch container details

/csapi/v1.3/containers/{containerSha}

[GET]

API request:

```
curl -X GET
'https://gateway.qg2.apps.qualys.com/csapi/v1.3/containers/fe23f264e3dd2b
5ebca5a5a9accdf5e4ac0d3ba3144bf097f6c5ba69e00d5fed' --header
'Authorization: Bearer <token>'
```

Response:

```
{
  "portMapping": [
    {
      "protocol": "tcp",
      "port": 5000,
      "hostIp": "0.0.0.0",
      "hostPort": 5000
    }
  ],
  "imageId": "5c4008a25e05",
  "created": "1615380901000",
  "updated": "1615440716646",
  "label": null,
  "uuid": "2866748c-acf2-3d09-8384-8ab0ed522048",
  "sha":
"fe23f264e3dd2b5ebca5a5a9accdf5e4ac0d3ba3144bf097f6c5ba69e00d5fed",
  "privileged": false,
  "path": "/entrypoint.sh",
  "imageSha":
"5c438a25e0565faf88dd3f060a8d7688f8c41c539a3ee9c693b05f08a12ebf9",
  "macAddress": "02:42:ac:11:00:02",
  "customerUuid": "74d66479-27e6-73bf-8033-34037f109fc9",
  "ipv4": "10.10.10.10",
  "ipv6": null,
  "name": "registry",
  "host": {
    "sensorUuid": "07ef65f6-472e-4d4c-9383-591b551f87ab",
    "hostname": "cloudagent",
    "ipAddress": "10.11.11.11",
    "uuid": "1fcb95c1-f881-4574-a452-febc3f55809a",
    "lastUpdated": "2021-03-10T12:56:01.213Z"
  },
  "state": "STOPPED",
  "imageUuid": "bfbdb86f3-e99a-32da-b73d-5d1c0cb77eaa",
  "containerId": "fe14f222e3dd",
```

```
"stateChanged": "1615440815994",
"services": null,
"users": null,
"operatingSystem": null,
"lastScanned": null,
"source": "GENERAL",
"isInstrumented": null,
"environment": [
  "PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin"
],
"arguments": [
  "/etc/docker/registry/config.yml"
],
"command": "/etc/docker/registry/config.yml",
"drift": null,
"vulnerabilities": null,
"softwares": null,
"isDrift": false,
"isRoot": true,
"lastComplianceScanned": null,
"cluster": {
  "type": "kubernetes",
  "version": "v1.1.1",
  "project": "k8s-project",

  "node": {
    "name": "k8s-node",
    "isMaster": true
  },
  "pod": {
    "name": "pod-name",
    "uuid": "63225d4f-bf0c-3488-8144-89c7d03dfacf",
    "namespace": "cs",
    "label": [
      {
        "key": "com.docker.compose.container-number",
        "value": "1"
      },
      {
        "key": "com.docker.compose.service",
        "value": "lb"
      }
    ]
  },
  "controllers": [
    {
      "uuid": "a3145d4f-bf0c-3488-8144-89c7d03dfacf",
      "name": "deployment-name",
      "type": "DEPLOYMENT"
    },
  ],
}
```

```
        {
          "uuid": "b3145d4f-bf0c-3488-8144-89c7d03dfacf",
          "name": "replicaset-name",
          "type": "REPLICASET"
        }
      ]
    }
  }
}
```

Fetch a detailed container list

/csapi/v1.3/containers/list

[GET]

API request:

```
curl -X GET
'https://gateway.qg2.apps.qualys.com/csapi/v1.3/containers/list?limit=2'
--header 'Authorization: Bearer <token>'
```

Response:

Response Headers:

```
{
  "content-type": "application/json;charset=UTF-8",
  "date": "Wed, 26 Aug 2020 07:56:59 GMT",
  "link":
"<http://qualysapi.qualys.com/csapi/v1.3/containers/list?limit=2&paginati
onQuery=updated%3E%3D1594876830694+AND+not+uuid%3A+%5B65eff01b-7775-3a0a-
bd68-e6dale7a39f5%5D>;rel=next",
  "referrer-policy": "same-origin",
  "server": "Qualys",
  "transfer-encoding": "chunked",
  "x-content-type-options": "nosniff",
  "x-frame-options": "sameorigin",
  "x-permitted-cross-domain-policies": "none",
  "x-powered-by": "Qualys",
  "x-xss-protection": "1; mode=block"
}
```

Response JSON:

```
{
  "data": [
    {
      "portMapping": null,
      "imageId": "3aabd9e6d667",
      "created": "1593759719000",
```



```
    "updated": "1594876830316",
    "label": null,
    "uuid": "cdb22685-9fbb-3971-90aa-706086a64079",
    "sha":
"a249ee950e639492563f2dd7caa231adc5fc3fe5714f1d82a23cc2f8bbcd22f",
    "privileged": false,
    "path": "/bin/sh",
    "imageSha":
"3abcd9e6d6670283b5e2df06bb77354b379b17ac7b940e8ea9edb202d423af13",
    "macAddress": "",
    "customerUuid": "a1d17d52-03fb-c803-83a6-1048acccaca9",
    "ipv4": null,
    "ipv6": null,
    "name": "container-1",
    "host": {
      "sensorUuid": "213d6f22-f8cc-42ad-814a-49b4b5c38d64",
      "hostname": "cloudagent",
      "ipAddress": "10.11.12.13",
      "uuid": "52d08bfe-2799-4f5c-81fb-8c0950d80f5e",
      "lastUpdated": null
    },
    "state": "DELETED",
    "imageUuid": "e2c22565-d1d9-3f9c-ac83-ad9b09f110ef",
    "containerId": "a978ee930e63",
    "stateChanged": "1593759720784",
    "services": null,
    "users": null,
    "operatingSystem": null,
    "lastScanned": null,
    "source": "GENERAL",
    "isInstrumented": null,
    "environment": [
      "VAULT_VERSION=0.9.3",
"PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin"
    ],
    "arguments": [
      "-c",
      "sed -i 's/\\r//g' /usr/local/bin/init.sh"
    ],
    "command": "/bin/sh -c sed -i 's/\\r//g' /usr/local/bin/init.sh",
    "drift": null,
    "vulnerabilities": null,
    "softwares": null,
    "isDrift": null,
    "isRoot": true
  },
  {
    "portMapping": null,
```

```
"imageId": "dd2f1712bc5a",
"created": "1593759720000",
"updated": "1594876830694",
"label": null,
"uuid": "62eff01b-7775-4a0a-bd68-e6db1e7a39f5",
"sha":
"c0db6c214c3ec6cd6620ad07e4e06da540c50c3c0c1372f84116dbe8aa03dd8d",
"privileged": false,
"path": "/bin/sh",
"imageSha":
"dd6d1713ac6a55dffe5faff50e7408f51f8f11081f7c4c43e8e84697becfa38b",
"macAddress": "",
"customerUuid": "a1d17f51-05fb-c803-84a6-1048accaca9",
"ipv4": null,
"ipv6": null,
"name": "container-2",
"host": {
  "sensorUuid": "213d6f22-f8cc-43bd-814a-49b4d5c38e64",
  "hostname": "cloudagent",
  "ipAddress": "10.11.12.13",
  "uuid": "53d08bfe-2899-4f5c-81fa-8c0950d80f5e",
  "lastUpdated": null
},
"state": "DELETED",
"imageUuid": "ca5f92da-7785-344b-8681-8c64f30a043b",
"containerId": "c0da6c314c2e",
"stateChanged": "1593759721924",
"services": null,
"users": null,
"operatingSystem": null,
"lastScanned": null,
"source": "GENERAL",
"isInstrumented": null,
"environment": [
  "VAULT_VERSION=0.9.3",

"PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin"
],
"arguments": [
  "-c",
  "sed -i 's/\\r//g' /usr/local/bin/docker-entrpoint.sh"
],
"command": "/bin/sh -c sed -i 's/\\r//g' /usr/local/bin/docker-entrpoint.sh",
"drift": null,
"vulnerabilities": null,
"softwares": null,
"isDrift": null,
"isRoot": true,
```

```
"cluster": {
  "type": "kubernetes",
  "version": "v1.1.1",
  "project": "k8s-project",

  "node": {
    "name": "k8s-node",
    "isMaster": true
  },
  "pod": {
    "name": "pod-name",
    "uuid": "6325d4f-bf0c-3488-8144-89c7d03dfacf",
    "namespace": "cs",
    "label": [
      {
        "key": "com.docker.compose.container-number",
        "value": "1"
      },
      {
        "key": "com.docker.compose.service",
        "value": "lb"
      }
    ]
  }
  "controllers": [
    {
      "uuid": "a3145d4f-bf0c-3488-8144-89c7d03dfacf",
      "name": "deployment-name",
      "type": "DEPLOYMENT"
    },
    {
      "uuid": "b3145d4f-bf0c-3488-8144-89c7d03dfacf",
      "name": "replicaset-name",
      "type": "REPLICASET"
    }
  ]
}

}

},
"limit": 2
}
```

Fetch sensor details

/csapi/v1.3/sensors/{sensorId}

[GET]

API request:

```
curl -X GET
'https://gateway.qg2.apps.qualys.com/csapi/v1.3/sensors/29d0a0bf6cdf' --
header 'Authorization: Bearer <token>'
```

Response:

```
{
  "uuid": "07ed65f6-474e-4d4c-9393-591b551f87ac",
  "activationUuid": "43f2d8e2-0551-48e0-a8e0-102c7d531922",
  "hostname": "cloudagent",
  "customerUuid": "77d66479-27e6-71bf-8143-35087e107fc9",
  "dockerVersion": "19.03.5",
  "ipv4": "10.11.12.13",
  "os": "CentOS Linux 7 (Core)",
  "ipv6": "",
  "sensorVersion": "1.7.0-6",
  "platform": "LINUX_SENSOR",
  "lastCheckedIn": "1615440575433",
  "label": [
    {
      "key": "build-date",
      "value": "Tue Jan  5 18:28:18 UTC 2021"
    },
    {
      "key": "image-source",
      "value": "SJC-USA"
    }
  ],
  "privileged": "false",
  "macAddress": "00:50:56:aa:fb:9a",
  "vulnSigVersion": null,
  "hostUuid": "1fcf95c1-f881-4584-a452-febe3f55809a",
  "configurationProfile": null,
  "status": "Unknown",
  "registry": "docker.io",
  "sha":
"23d0a0bf6cdf9e513d15b17a4533e1feec976288f422b0f60238326941ebad65",
  "sensorId": "29d0a0bf6cef",
  "name": "qualys-container-sensor",
  "created": "1615380389000",
  "imageId": "9e9ff9b51e30",
  "imageSha":
```

```
"9e9fe9b52e30f0997de62aab796fc60e12e357d4d2c7e0c58298833c3fcf788d",
  "binaryVersion": null,
  "containerIpv4": null,
  "containerIpv6": null,
  "containerMacAddress": "",
  "sensorType": "GENERAL",
  "containerRuntime": "DOCKER",
  "containerRuntimeVersion": "19.03.5",
  "cluster": {
    "type": "kubernetes",
    "version": "v1.1.1",
    "project": "k8s-project",

    "node": {
      "name": "k8s-node",
      "isMaster": true
    },
    "pod": {
      "name": "pod-name",
      "uuid": "6325d4f-bf0c-3488-8144-89c7d03dfacf",
      "namespace": "cs",
      "label": [
        {
          "key": "com.docker.compose.container-number",
          "value": "1"
        },
        {
          "key": "com.docker.compose.service",
          "value": "lb"
        }
      ]
    },
    "controllers": [
      {
        "uuid": "a3145d4f-bf0c-3488-8144-89c7d03dfacf",
        "name": "deployment-name",
        "type": "DEPLOYMENT"
      },
      {
        "uuid": "b3145d4f-bf0c-3488-8144-89c7d03dfacf",
        "name": "replicaset-name",
        "type": "REPLICASET"
      }
    ]
  }
}
```