



Qualys CloudView v1.x

API Release Notes

Version 1.16.1.0

June 22, 2021

The Qualys CloudView API provides automation and integration capabilities for your Qualys subscription. You'll find all the details in our user guides, available at the time of release. Just log in to your Qualys account and go to Help > Resources.

What's New

[Enable Remediation through API](#)

URL to the Qualys API Server

Qualys maintains multiple Qualys platforms. The Qualys API server URL that you should use for API requests depends on the platform where your account is located.

Account Location	API Server URL
Qualys US Platform 1	https://qualysguard.qualys.com
Qualys US Platform 2	https://qualysguard.qg2.apps.qualys.com
Qualys US Platform 3	https://qualysguard.qg3.apps.qualys.com
Qualys EU Platform 1	https://qualysguard.qualys.eu
Qualys EU Platform 2	https://qualysguard.qg2.apps.qualys.eu
Qualys India Platform 1	https://qualysguard.qg1.apps.qualys.in
Qualys Canada Platform	https://qualysguard.qg1.apps.qualys.ca
Qualys AE Platform	https://qualysguard.qg1.apps.qualys.ae
Qualys Private Cloud Platform	https://qualysguard.<customer_base_url>

The Qualys API documentation and sample code use the API server URL for the Qualys US Platform 1. If your account is located on another platform, please replace this URL with the appropriate server URL for your account.

Enable Remediation through API

APIs updated	/rest/v1/aws/connectors POST /rest/v1/azure/connectors POST /rest/v1/gcp/connectors POST /rest/v1/aws/connectors PUT /rest/v1/azure/connectors PUT /rest/v1/gcp/connectors PUT Note: If remediation is enabled on a connector, the response includes <code>remediationEnabled: true</code>.
New or Updated APIs	Updated
Operator	POST, PUT

We have now introduced a new optional flag to enable remediation through API. To create a new connector with remediation enabled, you need to set the `remediationEnabled` flag to `true`. This is an optional flag. If this flag is not specified in the create connector request, remediation is not enabled for the connector. However, you can enable or disable remediation for an existing connector by setting the flag to `true` using the update request. By default, remediation is disabled for connectors.

Note: The remediation feature is available only to Cloud Security Assessment (CSA) subscribers and is enabled by default.

Input Parameters

Parameter	Description
connectorBody	Set the flag to true to enable remediation. By default, the flag is set to false. If the flag is not specified in the create request, then the remediation is disabled for the connector. During the update request, if the flag is not specified, the current remediation status is reflected in the response. AWS: <pre>{ "arn": "string", "description": "string", "externalId": "string", "isChinaRegion": true, "isGovCloud": false, "isPortalConnector": true, "name": "string", "remediationEnabled": true "pollingFrequency": { "hours": 0, "minutes": 0", } }</pre> Azure: <pre>{ "applicationId": "string", "authenticationKey": "string", "description": "string", "directoryId": "string", "isGovCloud": false, "name": "string", "remediationEnabled": true "subscriptionId": "string" "pollingFrequency": { "hours": 0, "minutes": 0" } }</pre> GCP: For GCP connectors, set the input parameter remediationEnabled to true. "remediationEnabled": true

Let us view the examples for API changes in detail.

AWS

[AWS: Create Connector with remediation enabled](#)

Update AWS Connector to Enable/Disable Remediation

Azure

Azure: Create Connector with Remediation Enabled

Azure: Update Connector to Enable Remediation

GCP

GCP: Create Connector with Remediation Enabled

GCP: Update Connector to Enable Remediation

AWS: Create Connector with remediation enabled

Let us create a connector and enable remediation for the same. To enable remediation, the `remediationEnabled` flag needs to be set to true. This flag is optional. By default, this flag is configured to false.

API request:

```
curl -k -X GET -u <username>:<password>  
'https://<QualysURL>/cloudview-  
api/rest/v1/aws/connectors?pageNo=0&pageSize=50'
```

Request POST Data:

```
{  
  "arn": "arn:aws:iam::XXXXXXXXXXXX:role/Sample_CV_role",  
  "description": "Connector created through API automation",  
  "externalId": "pod01-438846-1629479060624",  
  "isPortalConnector": false,  
  "name": "AWS Connector",  
  "remediationEnabled": true  
}
```

Response:

```
{  
  "name": "AWS Connector",  
  "connectorId": "XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX",  
  "description": "Connector created through API automation",  
  "provider": "AWS",  
  "state": "PENDING",  
  "totalAssets": 0,  
  "lastSyncedOn": "Mon Jul 05 13:26:17 UTC 2021",  
  "nextSyncedOn": "Mon Jul 05 17:26:17 UTC 2021",  
  "remediationEnabled": true,  
  "isGovCloud": false,  
  "isChinaRegion": false,  
  "awsAccountId": "XXXXXXXXXXXX",  
  "isDisabled": false,  
  "groups": [],  
  "pollingFrequency": {  
    "hours": 4,  
    "minutes": 0  
  },  
  "error": "",  
  "externalId": "pod01-438846-1629479060624",  
  "arn": "arn:aws:iam::XXXXXXXXXXXX:role/Sample_CV_role",  
  "isPortalConnector": false  
}
```

Update AWS Connector to Enable/Disable Remediation

To enable remediation, set the flag `remediationEnabled` to `true`. If remediation is enabled on a connector, the response includes `"remediationEnabled": true`. If you do not set the flag in update request, the existing remediation status of the connector is displayed in the response.

API request:

```
curl -k -X POST -u <username>:<password>
'https://<QualysURL>/cloudview-api/rest/v1/aws/connectors'
```

Request POST Data:

```
{
  "arn": "arn:aws:iam::XXXXXXXXXXXX:role/CloudViewTest",
  "description": "Sample External ID",
  "externalId": "USPOD01-4765-9011278609223",
  "isChinaRegion": true,
  "isGovCloud": true,
  "isPortalConnector": true,
  "name": "ABC",
  "remediationEnabled": true,
  "pollingFrequency": {
    "hours": 1,
    "minutes": 1
  }
}
```

Response:

```
{
  "name": "AWS Connector 2",
  "connectorId": "1111a111-1111-11a1-a1a1-1aa1a1111111",
  "description": "",
  "provider": "AWS",
  "state": "SUCCESS",
  "totalAssets": 333,
  "lastSyncedOn": "Thu May 20 11:52:00 UTC 2021",
  "nextSyncedOn": "Thu May 20 13:50:52 UTC 2021",
  "remediationEnabled": true,
  "isGovCloud": false,
  "isChinaRegion": false,
  "awsAccountId": "XXXXXXXXXXXX",
  "accountAlias": "SAMPLE-ACCOUNT",
  "isDisabled": false,
  "groups": [
    {
      "name": "AWS Connector",
      "uuid": "0ee655f8-2680-3ef1-98f7-d3d7b34aa485"
    },
    ...
  ]
}
```

```
,
"pollingFrequency": {
  "hours": 2,
  "minutes": 0
},
"baseAccountId": "XXXXXXXXXXXX",
"externalId": "pod04-2722092-1626634575829",
"arn": "arn:aws:iam:XXXXXXXXXXXX:role/sample_role",
"isPortalConnector": false
}
```

Let us consider another example where you do not specify remediation flag in the request. If you do not set the flag in update request, the existing remediation status of the connector is displayed in the response.

API request:

```
curl -k -X POST -u <username>:<password>
'https://<QualysURL>/cloudview-api/rest/v1/aws/connectors'
```

Request POST Data:

```
{
  "arn": "arn:aws:iam:XXXXXXXXXXXX:role/CloudViewTest",
  "description": "Sample External ID",
  "externalId": "USPOD01-4765-9011278609223",
  "isChinaRegion": true,
  "isGovCloud": true,
  "isPortalConnector": true,
  "name": "ABC",
  "pollingFrequency": {
    "hours": 1,
    "minutes": 1
  }
}
```

Response:

```
{
  "name": "AWS Connector 2",
  "connectorId": "1111a111-1111-11a1-a1a1-1a1a11111111",
  "description": "",
  "provider": "AWS",
  "state": "SUCCESS",
  "totalAssets": 333,
  "lastSyncedOn": "Thu May 20 11:52:00 UTC 2021",
  "nextSyncedOn": "Thu May 20 13:50:52 UTC 2021",
  "remediationEnabled": false,
  "isGovCloud": false,
  "isChinaRegion": false,
  "awsAccountId": "XXXXXXXXXXXX",
```

```

"accountAlias": "SAMPLE-ACCOUNT",
"isDisabled": false,
"groups": [
  {
    "name": "AWS Connector",
    "uuid": "0ee655f8-2680-3ef1-98f7-d3d7b34aa485"
  },
  ...
],
"pollingFrequency": {
  "hours": 2,
  "minutes": 0
},
"baseAccountId": "XXXXXXXXXXXX",
"externalId": "pod04-2722092-1626634575829",
"arn": "arn:aws:iam::XXXXXXXXXXXX:role/sample_role",
"isPortalConnector": false
}

```

Azure: Create Connector with Remediation Enabled

Let us create a connector and enable remediation for the same. To enable remediation, the `remediationEnabled` flag needs to be set to true. This flag is optional. By default, this flag is configured to false.

API request:

```

curl -k -X POST -u <username>:<password>
'https://<QualysURL>/cloudview-api/rest/v1/azure/connectors'

```

REQUEST POST DATA:

```

{
  "applicationId": "XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX",
  "authenticationKey": "XXXXXXXXXXXXXXXX",
  "description": "This is test description",
  "directoryId": "XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX",
  "isGovCloud": true,
  "name": "My Azure Connector",
  "remediationEnabled": true
  "subscriptionId": "XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX",
  "pollingFrequency": {
    "hours": 4,
    "minutes": 0
  },
}

```

Response:

```
{
  "applicationId": "XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX",
  "connectorId": "XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX",
  "description": "This is test description",
  "directoryId": "XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX",
  "error": "string",
  "groups": [],
  "isGovCloud": true,
  "lastSyncedOn": "Fri Apr 1 10:33:08 UTC 2020", "name": "My Azure
Connector",
  "provider": "AZURE",
  "remediationEnabled": true,
  "state": "PENDING",
  "subscriptionId": "XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX",
  "totalAssets": 0,
  "pollingFrequency": {
    "hours": 4,
    "minutes": 0
  },
  "nextSyncedOn": "Fri Apr 1 14:33:08 UTC 2020"
}
```

Azure: Update Connector to Enable Remediation

To enable remediation, set the flag `remediationEnabled` to `true`. If remediation is enabled on a connector, the response includes `"remediationEnabled": true`. If you do not set the flag in update request, the existing remediation status of the connector is displayed in the response.

API request:

```
curl -k -X PUT -u <username>:<password>
{
  "applicationId": "XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX",
  "authenticationKey": "XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX",
  "description": "Updating Connector created through API automation",
  "directoryId": "XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX",
  "name": "Azure API Connector",
  "subscriptionId": "XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX",
  "remediationEnabled": true
}
'https://<QualysURL>/cloudview-api/rest/v1/azure/connectors/7d80a840-
40b6-11e9-9078-111111111111'
```

Response:

```
{
  "name": "Azure API Connector",
  "connectorId": "11111111-1111-1111-1111-111111111111",
  "description": "Updating Connector created through API automation",
```

```

    "provider": "AZURE",
    "state": "PENDING",
    "totalAssets": 0,
    "lastSyncedOn": "Mon Jul 05 08:41:39 UTC 2021",
    "nextSyncedOn": "Mon Jul 05 12:37:40 UTC 2021",
    "remediationEnabled": true,
    "isGovCloud": false,
    "isDisabled": false,
    "applicationId": "XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX",
    "subscriptionId": "XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX",
    "subscriptionName": "sample_subscription",
    "groups": [],
    "pollingFrequency": {
      "hours": 4,
      "minutes": 0
    },
    "directoryId": "XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX"
  }
}

```

Let us consider another example where you do not specify remediation flag in the request. If you do not set the flag in update request, the existing remediation status of the connector is displayed in the response.

API request:

```

curl -k -X PUT -u <username>:<password>
{
  "applicationId": "XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX",
  "authenticationKey": "XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX",
  "description": "Updating Connector created through API automation",
  "directoryId": "XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX",
  "name": "Azure API Connector",
  "subscriptionId": "XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX",
}
'https://<QualysURL>/cloudview-api/rest/v1/azure/connectors/7d80a840-40b6-11e9-9078-111111111111'

```

Response:

```

{
  "name": "Azure API Connector",
  "connectorId": "11111111-1111-1111-1111-111111111111",
  "description": "Updating Connector created through API automation",
  "provider": "AZURE",
  "state": "PENDING",
  "totalAssets": 0,
  "lastSyncedOn": "Mon Jul 05 08:41:39 UTC 2021",
  "nextSyncedOn": "Mon Jul 05 12:37:40 UTC 2021",
  "remediationEnabled": false,
  "isGovCloud": false,
}

```

```

    "isDisabled": false,
    "applicationId": "XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX",
    "subscriptionId": "XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX",
    "subscriptionName": "sample_subscription",
    "groups": [],
    "pollingFrequency": {
      "hours": 4,
      "minutes": 0
    },
    "directoryId": "XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX"
  }
}

```

GCP: Create Connector with Remediation Enabled

Let us create a connector and enable remediation for the same. To enable remediation, the `remediationEnabled` flag needs to be set to true. This flag is optional. By default, this flag is configured to false.

API request:

```

curl -X POST -u <username>:<password>
-H "accept: */*"
-H "authorization: Basic xxxxxxxxxxxxxxxxxxxxxxxxxxxx"
-H "Content-Type: multipart/form-data"
-F "configFile=@my-project-1513669048551.json;type=application/json"
-F "description=Description"
-F "name=API Connector"
-F "projectId=my-project-XXXXXXXXXXXX"
-F "remediationEnabled=true"
"https://<QualysURL>/cloudview-api/rest/v1/gcp/connectors"

```

Response:

```

{
  "name": "API Connector",
  "connectorId": "11111111-1111-1111-1111-111111111111",
  "description": "Description",
  "provider": "GCP",
  "state": "PENDING",
  "totalAssets": 0,
  "lastSyncedOn": "Mon Jul 05 08:54:11 UTC 2021",
  "nextSyncedOn": "Mon Jul 05 12:54:11 UTC 2021",
  "remediationEnabled": true,
  "isDisabled": false,
  "projectId": "my-project-XXXXXXXXXXXX",
  "groups": [],
  "pollingFrequency": {
    "hours": 4,
    "minutes": 0
  }
}

```

```
}  
}
```

GCP: Update Connector to Enable Remediation

To enable remediation, set the flag `remediationEnabled` to `true`. If remediation is enabled on a connector, the response includes `"remediationEnabled": true`. If you do not set the flag in update request, the existing remediation status of the connector is displayed in the response.

API request:

```
curl -X PUT -u <username>:<password>  
"https://<QualysURL>/cloudview-api/rest/v1/gcp/connectors/11111111-1111-  
1111-1111-111111111111"  
-H "accept: */*"  
-H "authorization: Basic xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx"  
-H "Content-Type: multipart/form-data"  
-H "remediationEnabled=true"  
-d {"description": "Sample Description update"}
```

Response:

```
{  
  "name": "API Connector",  
  "connectorId": "11111111-1111-1111-1111-111111111111",  
  "description": "Sample Description update",  
  "provider": "GCP",  
  "state": "SUCCESS",  
  "totalAssets": 0,  
  "lastSyncedOn": "Mon Jul 05 08:58:23 UTC 2021",  
  "nextSyncedOn": "Mon Jul 05 12:56:37 UTC 2021",  
  "remediationEnabled": true,  
  "isDisabled": false,  
  "projectId": "my-project-XXXXXXXXXXXX",  
  "groups": [],  
  "pollingFrequency": {  
    "hours": 4,  
    "minutes": 0  
  }  
}
```

Let us consider another example where you do not specify remediation flag in the request. If you do not set the flag in update request, the existing remediation status of the connector is displayed in the response.

API request:

```
curl -X PUT -u <username>:<password>  
"https://<QualysURL>/cloudview-api/rest/v1/gcp/connectors/11111111-1111-  
1111-1111-111111111111"  
-H "accept: */*"
```

```
-H "authorization: Basic xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx"
-H "Content-Type: multipart/form-data"
-d {"description":"Sample Description update"}
```

Response:

```
{
  "name": "API Connector",
  "connectorId": "11111111-1111-1111-1111-111111111111",
  "description": "Sample Description update",
  "provider": "GCP",
  "state": "SUCCESS",
  "totalAssets": 0,
  "lastSyncedOn": "Mon Jul 05 08:58:23 UTC 2021",
  "nextSyncedOn": "Mon Jul 05 12:56:37 UTC 2021",
  "remediationEnabled": false,
  "isDisabled": false,
  "projectId": "my-project-XXXXXXXXXXXX",
  "groups": [],
  "pollingFrequency": {
    "hours": 4,
    "minutes": 0
  }
}
```