

Application Management Commands

- rabbitmqadmin, on page 2
- smartagentctl, on page 5

rabbitmqadmin

Use the **rabbitmqadmin** command to check the queue status for the **RabbitAdmin** service running on the Cisco Cyber Vision Center.

rabbitmqadmin [*options*] *subcommands*

Syntax Description

`-C CONFIG`, `--config=CONFIG`	Specifies the configuration file to use. Default: `~/.rabbitmqadmin.conf`
`-N NODE`, `--node=NODE`	Specifies the node described in the configuration file. Default: `'default'` (only if a configuration file is specified).
`-H HOST`, `--host=HOST`	Specifies the hostname of the RabbitMQ server. Default: `localhost`
`-P PORT`, `--port=PORT`	Specifies the port number for the RabbitMQ HTTP API. Default: `15672`
`--path-prefix=PATH_PREFIX`	Specifies a custom URI path prefix for the RabbitMQ HTTP API. `api` and the operation path will be appended to it. Default: blank string.
`-V VHOST`, `--vhost=VHOST`	Specifies the virtual host (vhost) to connect to. Default: All vhosts for listing commands and `/` for declaration commands.
`-u USERNAME`, `--username=USERNAME`	Specifies the username for authentication. Default: `guest`
`-p PASSWORD`, `--password=PASSWORD`	Specifies the password for authentication. Default: `guest`
`-U URI`, `--base-uri=URI`	Specifies a base HTTP API URI. `/api` and the operation path will be appended to it. Overrides `--host`, `--port`, and `--path-prefix`. Separately provide `-vhost`.
`-q`, `--quiet`	Suppresses status messages. Default: `True`
`-s`, `--ssl`	Enables SSL/TLS connection. Default: `False`
`--ssl-key-file=SSL_KEY_FILE`	Specifies the path to the PEM-formatted private key file for SSL/TLS.

`--ssl-cert-file=SSL_CERT_FILE`	Specifies the path to the PEM-formatted certificate file for SSL/TLS.
`-ssl-ca-cert-file=SSL_CA_CERT_FILE`	Specifies the path to the PEM-formatted CA certificate file for SSL/TLS.
`--ssl-disable-hostname-verification`	Disables hostname verification for SSL/TLS connections.
`-k`, `--ssl-insecure`	Disables all SSL/TLS validation (similar to `curl -k`).
`-t REQUEST_TIMEOUT`, `--request-timeout=REQUEST_TIMEOUT`	Sets the HTTP request timeout in seconds. Default: `120`
`-f FORMAT`, `--format=FORMAT`	Specifies the output format.
`S SORT`, `--sort=SORT`	Specifies the sort key for listing commands.
`-R`, `--sort-reverse`	Reverses the sort order.
`-d DEPTH`, `--depth=DEPTH`	Specifies the maximum recursion depth for listing tables. Default: 1

Command History

Release	Modification
4.0	This command was introduced.

This example displays how to check RabbitMQ queues:

```
root@center100:~# rabbitmqadmin list queues
+-----------------------------------------------------------------------------+----------+
|                                    name                                     | messages |
+-----------------------------------------------------------------------------+----------+
| ccv.queue.activediscovery_scanrequest.049e52db-0178-4df6-9a63-05a04cb60736 | 0        |
| ccv.queue.activediscovery_scanresult                                       | 0        |
| ccv.queue.authorized_sensors                                               | 0        |
| ccv.queue.authorized_sensors.burrow                                        | 0        |
| ccv.queue.authorized_sensors.republisher                                   | 0        |
| ccv.queue.burrow_results                                                   | 0        |
| ccv.queue.decode_errors                                                    | 0        |
| ccv.queue.device                                                           | 0        |
| ccv.queue.dpad                                                             | 0        |
| ccv.queue.events                                                           | 0        |
| ccv.queue.flow_properties_stats                                            | 0        |
| ccv.queue.flowtables.049e52db-0178-4df6-9a63-05a04cb60736                  | 0        |
| ccv.queue.flowtables.049e52db-0178-4df6-9a63-05a04cb60736.burrow           | 0        |
| ccv.queue.flowtables.expired                                               | 0        |
| ccv.queue.flowtables.unknown                                               | 0        |
| ccv.queue.jobs.049e52db-0178-4df6-9a63-05a04cb60736                        | 0        |
| ccv.queue.jobs_results.aspic                                               | 0        |
| ccv.queue.jobs_results.backend                                             | 0        |
| ccv.queue.jobs_results.extensionapid                                       | 0        |
| ccv.queue.jobs_results.inputd                                              | 0        |
| ccv.queue.jobs_results.worker                                              | 0        |
| ccv.queue.snort_events                                                     | 0        |
| ccv.queue.sysinfo                                                          | 0        |
```

```
| kraken.initial_modifications                                                  | 0         |
| kraken.modifications                                                          | 0         |
| kraken.queue.lc.certificate_renewal                                           | 0         |
+-------------------------------------------------------------------------------+-----------+
```

smartagentctl

Use the **smartagentctl** command to manage the license information.

smartagentctl [**COMMAND**]

Syntax Description

`--register <registration-token-file-path>`	Registers the agent using the registration token file provided.
`--reregister <registration-token-file-path>`	Reregisters the agent using the registration token file provided.
`--deregister`	Deregisters the agent.
`-request-entitlement <entitlement-tag> <component-count>`	Requests an entitlement with the specified tag and component count.
`--set-transport-mode <mode> [<host>] [<port>] [<username>] [<password>]`	Sets the transport mode and optionally provides connection details (host, port, username, password).
`--get-transport-mode`	Retrieves the current transport mode.
`--get-license-summary`	Retrieves a summary of the license information.
`--enable-reservation`	Enables license reservation.
`--disable-reservation`	Disables license reservation.
`--get-reservation-request-code`	Retrieves the license reservation request code.
`--cancel-reservation-request`	Cancels the license reservation request.
`--get-reservation-confirmation-code`	Retrieves the license reservation confirmation code.
`--get-reservation-return-code`	Retrieves the license reservation return code.
`--install-reservation <authorization-code-file-path>`	Installs a license reservation using the authorization code file provided.
`-return-reservation-authorization-code`	Returns the license reservation authorization code.
`--utility-mode-is-enabled`	Checks if utility mode is enabled.
`--enable-utility-mode`	Enables utility mode.
`--disable-utility-mode`	Disables utility mode.
`-set-customer-info <info-type> <info-data>`	Sets customer information of the specified type.
`--get-customer-info`	Retrieves customer information.
`--get-tech-support`	Retrieves technical support information.

Command History

Release	Modification
4.0	This command was introduced.

This example displays how to disable a license:

```
root@center100:~# smartagentctl --disable-reservation
{"status":"SUCCESS","message":"Operation Success."}
```

This example displays how to enable a license service:

```
root@center100:~# smartagentctl --enable-reservation
{"status":"SUCCESS","message":"Success"}
```

This example displays how to restart the smart agent service:

```
root@center100:~# systemctl restart sbs-backend.service
```