Swagger Editor
Supported by **SMARTBEAR**

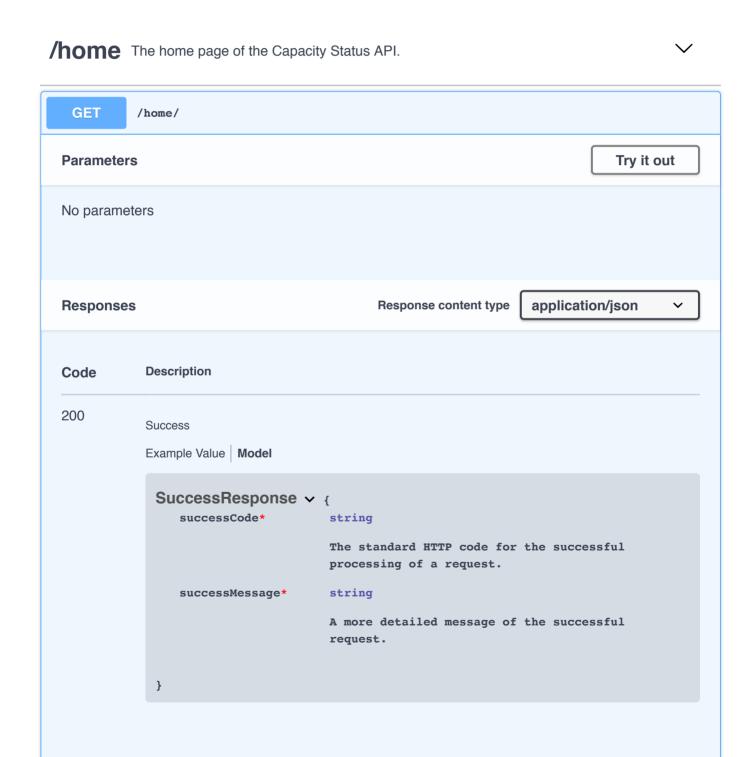File ▾   Edit ▾   Generate Server ▾   Generate Client ▾

Capacity Status API [0.0.1]

[Base URL: /]

The Capacity Status API is an API to enable UEC service providers and DoS Leads to change the RAG status (also known as the Capacity Status) of their services in DoS.

/home The home page of the Capacity Status API. ⌄

GET	/home/

Parameters   **Try it out**

No parameters

Responses   Response content type   application/json ⌄

Code	Description
200	Success

Example Value | **Model**

```
SuccessResponse ⌄ {
    successCode*          string
                          The standard HTTP code for the successful
                          processing of a request.
    successMessage*       string
                          A more detailed message of the successful
                          request.

}
```

Code	Description

401

Unauthorized - when a user is unauthorised to use this API.

Example Value | **Model**

ErrorResponse ∨ {
    errorCode*   string

                          A specific error code identifying the actual error that occurred.

    errorMessage*   string

                          A human readable message for the error.

}

status/update The Capacity Status API update service   ∨

POST   **/status/update/**

This endpoint will update the capacity (RAG) status of a single service in DoS as specified in the request body JSON payload. The format of the JSON payload is described further down in this document.

Business Logic

Upon processing a successful request, the API will return the success response (see below for the format of the success response) and will update DoS with the following information:

- **RAG status** - this is the capacity status of the service, and will be updated to the service status defined in the request payload. The API will allow an authenticated and authorised user to update the status of a service to Red, Amber, or Green.
- **Reset time** - this is the time when the service will go back into a Green state having been set to either Amber or Red by the request. The reset time will be set to the date and time when the service status is updated (or maintained) plus the amount of time in minutes as defined by the reset time value in the request payload. If this value is 0 or not provided, the reset time will default to 4 hours from the time the service status is updated. A state change to Red or Amber for a service will persist for a maximum period of 24 hours per request before automatically being reset back to Green. This means that if a service needs to be Red or Amber for over 24 hours, multiple requests through this API for that service will be required.
- **Notes** - a free text field providing the opportunity for any additional ad-hoc notes to be added regarding the state change. Notes defined in the request payload will be appended to the text: RAG status set by the Capacity Service API -.

- **Last updated date/time** - this is a timestamp of when the service was last updated and will be set to the date and time that the service was updated.
- **By** - this is a record of who last updated the service, and will be updated to the user identifier for the user that posted the request to the API.

Requests that fail the request validation rules will not be processed and will return the validation error response (see below for the validation rules and the format of the validation error response).

Requests that fail to be processed due to an error or failure will return an error response (see below for the format of the error response)

Request Validation

Values specified in the request payload will be validated by the API. The following business validation rules are in place:

- **VAL-0001** - ServiceIdentifier must be provided - returned when no ServiceIdentifer has been specified in the request payload.
- **VAL-0002** - ServiceState must be provided - returned when no ServiceState has been specified in the request payload.
- **VAL-0003** - Invalid ServiceState value - returned when a service state other than that defined by the Service Update Model is specified.
- **VAL-0004** - Invalid ResetTime value - returned when a reset time other than that defined by the Service Update Model is specified.
- **VAL-0005** - ResetTime outside of limits - returned when the reset time specified is outside of the minimum or maximum limits defined by the Service Update Model.
- **VAL-0006** - Notes outside of limits - returned when the notes specified are outside of the maximum limit defined by the Service Update Model.

Response Formats

Please refer to the models section of this document for the response formats.

Parameters   Try it out

Name	Description
payload * **required** **object** *(body)*	Example Value | **Model**

Name	Description

ServiceUpdate ⌄ {

   serviceId*   **string**

The DoS Service UID of the service to update its capacity (RAG) status.

   serviceState*   **string**
example: Red

The RAG state to set the service to.

Enum:

> Array [9]

   resetTime   **integer**
default: 240
minimum: 0
maximum: 1440

The amount of time, specified in 1 minute blocks up to and including 24 hours (1440 minutes), from the time the service status is updated by the request to reset the RAG status of the service back to GREEN. If no value or 0 is provided, the reset time will default to 4 hours (240 minutes).

   notes   **string**
default: RAG status set by Capacity Service API

Optional field of up to 900 characters to add ad-hoc notes to this status update action. Any notes provided will be appended to the end of the default notes set by the API, e.g. RAG status set by the Capacity Service API – additional notes here.

}

Responses

Response content type   [application/json ⌄]

Code	Description

Code	Description

200

Success

Example Value | **Model**

SuccessResponse ⌄ {
    successCode*   **string**

                      **The standard HTTP code for the successful
                      processing of a request.**

    successMessage*   **string**

                      **A more detailed message of the successful
                      request.**

}

400

Bad Request - upon receiving a corupt request or a request which fails API validation rules.

Example Value | **Model**

ValidationErrorResponse ⌄ {
    validationCode*   **string**

                      **A specific validation code identifying the actual
                      validation error that occurred.**

    validationMessage* **string**

                      **A human readable message for why the validation
                      of the request failed.**

}

401

Unauthorized - when a user is unauthorised to use this API.

Example Value | **Model**

ErrorResponse ⌄ {
    errorCode*   **string**

                      **A specific error code identifying the actual
                      error that occurred.**

    errorMessage*   **string**

                      A human readable message for the error.

}

Code	Description

403

Forbidden - when a user is not authorised to update the status of the requested service.

Example Value | **Model**

ErrorResponse ⌄ {
    errorCode*   string

                    A specific error code identifying the actual error that occurred.

    errorMessage*   string

                    A human readable message for the error.

}

404

Not Found - when the requested service to update could not be found in DoS.

Example Value | **Model**

ErrorResponse ⌄ {
    errorCode*   string

                    A specific error code identifying the actual error that occurred.

    errorMessage*   string

                    A human readable message for the error.

}

408

Request Timeout - when the server times out waiting for a request to be processed.

Example Value | **Model**

ErrorResponse ⌄ {
    errorCode*   string

                    A specific error code identifying the actual error that occurred.

    errorMessage*   string

                    A human readable message for the error.

}

Code	Description
500	Internal Server Error - when an unexpected error is encountered whilst processing the request.

Example Value | **Model**

```
ErrorResponse ∨ {
    errorCode*            string

                          A specific error code identifying the actual
                          error that occurred.

    errorMessage*         string

                          A human readable message for the error.


}
```

Models   ∨

```
ErrorResponse ∨ {
    errorCode*            string

                          A specific error code identifying the actual error that
                          occurred.

    errorMessage*         string

                          A human readable message for the error.


}
```

SuccessResponse ⌄ {

    successCode*   **string**

                                  The standard HTTP code for the successful processing of a request.

    successMessage*   **string**

                                  A more detailed message of the successful request.

}

ServiceUpdate ⌄ {

    serviceId*   **string**

                                  The DoS Service UID of the service to update its capacity (RAG) status.

    serviceState*   **string**
                                  example: Red

                                  The RAG state to set the service to.

                                  Enum:

                                  > Array [9]

    resetTime   **integer**
                                  default: 240
                                  minimum: 0
                                  maximum: 1440

                                  The amount of time, specified in 1 minute blocks up to and including 24 hours (1440 minutes), from the time the service status is updated by the request to reset the RAG status of the service back to GREEN. If no value or 0 is provided, the reset time will default to 4 hours (240 minutes).

    notes   **string**
                                  default: RAG status set by Capacity Service API

                                  Optional field of up to 900 characters to add ad-hoc notes to this status update action. Any notes provided will be appended to the end of the default notes set by the API, e.g. RAG status set by the Capacity Service API – additional notes here.

}

ValidationErrorResponse ∨ {

    validationCode*   string

                        A specific validation code identifying the actual validation error that occurred.

    validationMessage*   string

                        A human readable message for why the validation of the request failed.

}