SAMSUNG Knox Documentation

Samsung Knox developer guides

Error codes

On this page

The Knox Guard API uses the conventional HTTP response code to indicate the success or failure of a request. A 2XX response code indicates success, a 4XX response code indicates an error given the inputs of the request, and a 5XX response code indicates a failure with the Knox Guard platform. In addition, errors contain a JSON response body with more information about the error:

HTTP/1.1 200 OK Content-Type: application/json

```
{
    "result" : "SUCCESS",
    objectId" : "SDAKJ23DKCMSDAKENXSSAEQ",
    "requestedId": "111000291332456"
}
```

Async result : POST /kcs/v1.1/kg/transaction

HTTP/1.1 200 OK Content-Type: application/json

```
{
```

```
    "count" : 3,
    "success": 2,
    "fail": 1,
    "resultList": [
        {
            "result": "FAIL",
            "requestedId": "123456789012347"
            "error" : {
                "code" : "4000000",
                "message" : "RESOURCE_INVALID_PARAM",
                "reason" : "RESOURCE_INVALID_PARAM"
            }
        }
    ]
}
```


HTTP/1.1 401 Unauthorized Content-Type: application/json


```
{
    "result" : "FAIL",
    "error" :
    {
        "code": 4010000,
        "message": "AUTHORIZATION_FAIL",
        "reason": "Invalid Api Key"
    }
}
```


Error response body

The error response attributes are:

Property	Type	Required	Description
result	String	**Yes**	SUCCESS/FAIL
error	Error	**Yes**	Error object

Error object

The error objects contain the following attributes:

Property	Type	Required	Description
code	String	**Yes**	A Knox Guard defined error code that serves as a more specific indicator of the error than the HTTP response code.
message	String	**Yes**	A description of the error, intended to aid developers in debugging the error response.
reason	String	**Yes**	A description of the error, intended to aid developers in debugging the error response.

Standard HTTP error codes

The following table lists the most common HTTP error responses:

Code	Name	Description
400	Bad Request	The client has issued an invalid request. This is commonly used to specify validation errors in a request payload.
401	Unauthorized	Authorization for the API is required, but the request has not been authenticated.
403	Forbidden	The request has been authenticated but does not have the appropriate permissions, or the requested resource cannot be found.
404	Not Found	Specifies that the requested path does not exist.
406	Not Acceptable	The client has requested a MIME type via the Accept header for a value not supported by the server.
422	Unprocessable Entity	The client has made a valid request, but the server cannot process it. This is often returned for APIs when certain limits have been exceeded.
429	Too Many Requests	The client has exceeded the number of requests allowed for a given time window.
500	Internal Server Error	An unexpected error on the SmartThings servers has occurred. These errors should be rare.
501	Not Implemented	The client request was valid and understood by the server, but the requested feature has yet to be implemented. These errors should be rare.

Knox Guard error codes

Knox Guard specifies several custom error codes which provide more information than the standard HTTP error response codes. The following table lists the standard Knox Guard error codes and their descriptions:

Code	Message	Description
4010000	AUTHORIZATION_FAIL	The API key is not valid or the restriction data does not match.
4002102	API_KEY_RESTRICTION_INVALID	The IP address or HTTP referer format is incorrect.
4042101	API_KEY_RESTRICTION_NOT_FOUND	The IP address or HTTP referer is empty.
4040100	USER_NOT_FOUND	There is no user linked to the API key.
4040900	TENANT_NOT_FOUND	There is no tenant linked to the API key.
4000000	RESOURCE_INVALID_PARAM	Argument is invalid.
4001809	LICENSE_IS_LACK	There are no additional licenses that can be

Code	Message	Description
		activated.
4040300	DEVICE_NOT_FOUND	The specified device was not found.
4000310	DEVICE_STATE_INVALID	The operation is not permitted in the current state.
4000316	DEVICE_BULK_OPERATION_LIMIT_EXCEEDED	The bulk operation limit has been exceeded.
4040400	PROFILE_NOT_FOUND	Internal profile not found.
4001807	LICENSE_MAX_COUNT_REACHED	License max count reached.
4001805	LICENSE_NAME_ALREADY_EXISTS	License name already exists.
5001804	LICENSE_INTERNAL_SERVER_FAILED	Error communicating with the license server.
4001813	LICENSE_KEY_ALREADY_EXISTS_IN_OTHER_REGION	License key already registered to another region (EU/US).
4001802	LICENSE_MAX_TRIAL_LICENSE_COUNT_REACHED	Max trial license count reached.
4041800	LICENSE_NOT_FOUND	License not found.
4001806	LICENSE_DEVICE_MAPPING_FOUND	Device already registered with a license.
4001812	LICENSE_ALREADY_DELETED	License already deleted.
4030102	USER_IN_INVALID_STATE	User state is invalid.
4030107	USER_REJECTED_STATE	User is rejected by Samsung Admin.
4030104	USER_BLOCKED_STATE	User is blocked by Samsung Admin.
4030106	USER_PENDING_STATE	User is not approved yet.

Is this page helpful?

YES **No**

KNOX DEVELOPER DOCUMENTATION

SUPPORT

SAMSUNG

PRIVACY POLICY | LEGAL