Oracle Hospitality Integration Platform

Property Systems Room Keys Outbound

Release 25.1
G31641-01
August 2025

Oracle Hospitality Integration Platform Property Systems Room Keys Outbound, Release 25.1

G31641-01

Contents

Preface

1 Business context

2 Recommended Tools

3 Environment

4 Connection

5 Configurations

6 Workflows

Workflow: Create New Room Key Plus Additional Room Key 1
Workflow: Delete Active Room Key(s) 9
Workflow: Create a One-Shot Room Key 13
Workflow: Read a Room Key on a Specific Key Encoder 17
Workflow: Handling of Room Keys on Room Move 19
Workflow: Handling of Room Keys at Reservation Stay Changes 27

7 Anti-patterns

8 Key Terminology

9 FAQ

Preface

OPERA Cloud provides an API operation POST RoomKeysOutbound that manages requests for Door Lock Room Keys. This operation is used by the OPERA Cloud PMS connected to external Door Lock Systems systems through OPERA Cloud Outbound Systems configuration.

Purpose

This guide describes the POST RoomKeysOutbound API operation that manages requests for Door Lock Room Keys. It provides guidance and explains the workflows for the handling of guest room keys at arrival and during a guest's stay as well as at check-out.

Audience

This guide is intended for customers and partners who develop applications with the Oracle Hospitality Integration Platform.

Documentation

Oracle Hospitality product documentation is available on the Oracle Help Center at https://docs.oracle.com/en/industries/hospitality/.

Revision History

Date	Description of Change
August 2025	Initial publication

1

Business context

OPERA Cloud provides an API operation **POST RoomKeysOutbound** that manages requests for Door Lock Room Keys. This operation is used by OPERA Cloud PMS connected to external Door Lock Systems systems through OPERA Cloud Outbound Systems configuration.

The new API operation supports all possible requests to the Door Lock system (DLS); different types of key actions like New Key, Duplicate Key, Delete Key, and Read Key requests, but also Modify and Room move requests (for online key systems).

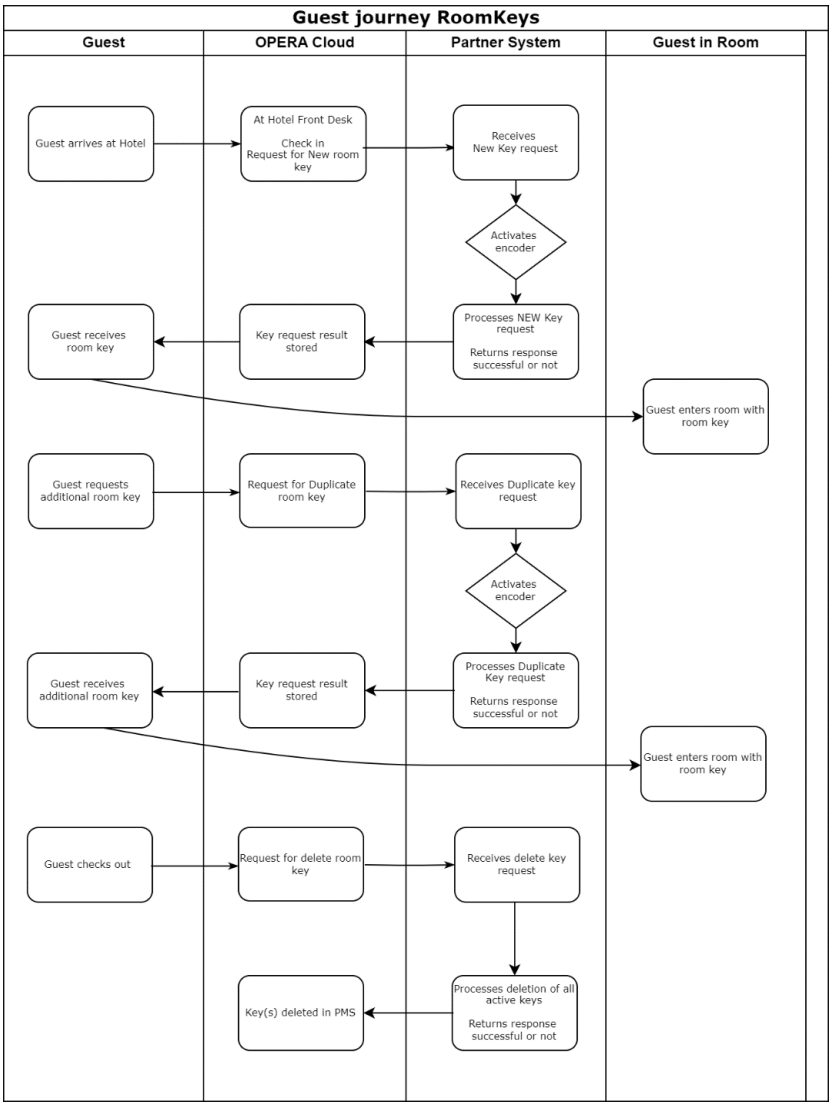
The Room Key API provides the necessary data to create or active a room key for hotel guests prior to check in, at time of check-in, but also while checked in to the hotel.

This implementation guide provides guidance and explains the workflows for the handling of guest room keys at arrival and during a guest's stay as well as at check-out.

The implementation guide will cover the following use cases for room key creation from OPERA Cloud PMS:

	Use Case	Description
1	Create room new room key at reservation Check-in	OPERA Cloud PMS sends request(s) for a new room key or Pin code at time of reservation check in. This can be a request for the first room key or a Pin code to become active.
2	Create additional room key(s) after Check-in	OPERA Cloud PMS sends request(s) for additional room key(s) or Pin code. This is in addtion to the existing room keys or Pin code. This can be one or multiple requests for additional active room keys or an additional Pin code.
3	Delete room key(s) at Check out	OPERA Cloud PMS sends a request to delete all active room keys or Pin codes for a reservation at time of check out of the hotel guest or at any time during the guest stay. This is initiated by the OPERA Cloud user. Request to delete specific room keys through OPERA Cloud PMS is not available.
4	Create One-Shot key	OPERA Cloud PMS sends a request for a room key or a Pin Code which is not related to a reservation and is only valid for a specific time. Usually a one-shot-key is handed out to visitors who want to see if a hotel guest room is issued, for example, to workers who carry out repairs in the room.
5	Read a room key at a key Encoder	OPERA Cloud PMS sends a request to read a room key on a specific key encoder. This is usually done when a room key must be verified if still active and the room must be identified to which the key belongs.
6	Handling of room keys at reservation room move	Description of handling of room key changes when a reservation will move from one room to another room. There are different requests sent by OPERA Cloud PMS to an "offline" door lock system versus an "online" door lock system.
7	Handling of room keys at reservation stay changes	Description of handling of room key changes when a reservation departure date or time will change. There are different requests sent by OPERA Cloud PMS to an "offline" door lock system versus an "online" door lock system.

Workflow diagram:



Journey: Guest arrival

Guest arrives at hotel, with check-in the partner system receives the request to activate a new room key.

Partner system sends request to the Key encoder mentioned in the request call.

The key encoder processes the request and returns result back to the partner system.

Partner system sends back response message with the result (successful or failed) to OPERA Cloud.

If room key is successful the hotel user will hand out the room key to the guest.

Guest can access the room with the room key.

Same flow for partner systems working with Pin codes instead of physical keys, then no Key encoder will be in place, partner system generates the Pin code and sends it back to OPERA Cloud in response message.

OPERA Cloud user hands out the Pin code to the hotel guest.

Guest can access the room with the Pin code.

Journey: Guest asks for additional room key.

Partner system receives the request to activate an additional (duplicate) room key.

Partner system sends request to the Key encoder mentioned in the request call.

The key encoder processes the request and returns result

	back to the partner system. Partner system sends back response message with the result (successful or failed) to OPERA Cloud. If room key is successful the hotel user will hand out the room key to the guest. Guest can access the room with the additional room key. Journey: Guest checks out. Existing room keys /Pin codes become invalid. Partner system receives request to remove active room key(s) or Pin code from its system, respective stop allowing access to the room. Partner system sets ALL room keys and Pin codes inactive for the specific reservation. It is allowed to define a grace time a room key or Pin code will be valid to open the room after receiving the request to remove active room key(s) or Pin code

2

Recommended Tools

The below software tools are recommended for use when following the steps outlined in this guide.

Tool	Description	Links
Github	A GitHub repository containing both Oracle Hospitality REST API specifications and accompanying Postman Collections.	Github and Postman Collections Github
Postman Collections	In the postman-collections folder in GitHub, you can download and use the Postman Collections associated to the use cases explained in this guide.	Postman Collections Postman Collection roomKeysOutbound

3 Environment

The following environment and modules must be available to perform the business cases with the OPERA Cloud APIs.

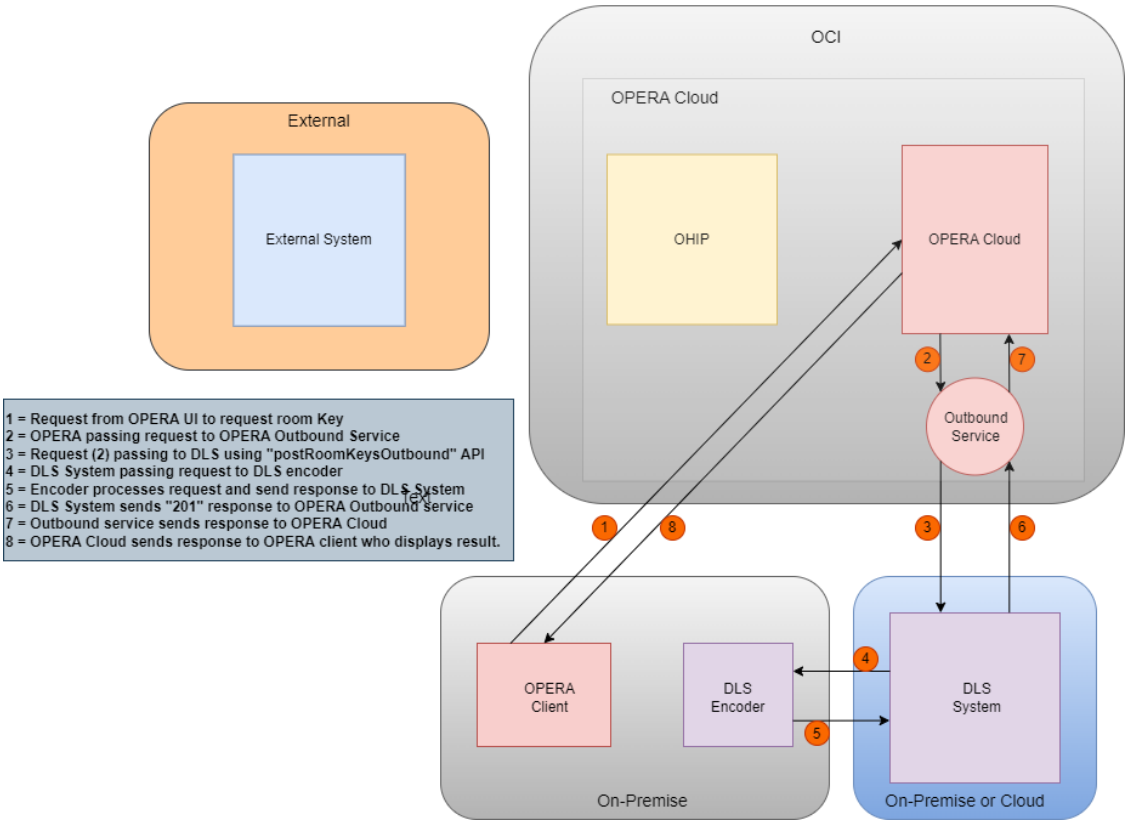
OPERA Cloud Platform Module	Description	Minimum Version
OPERA Cloud PMS	The customer must have a subscription to OPERA Cloud Foundation.	24.2.0.0
Oracle Hospitality Integration Platform (OHIP)	Only required when Door Lock System must retrieve mandatory data (Chain code, HotelId, rooms) automatically through OHIP. For customers, OHIP is included in the subscription to OPERA Cloud Foundation. Partners need a subscription to Oracle Hospitality Integration Cloud Service.	The latest released version. OHIP is a single version product.

4 Connection

The Connection of the Door Lock System with OPERA Cloud PMS for handling guest room key requests from the PMS does not directly work with the Oracle Hospitality Integration Platform (OHIP). It is instead done through a direct connection to the Door Lock system using the OPERA Cloud Outbound System Service.

General Workflow Outbound Service for DLS API

Room Keys Outbound workflow



For the proper connection configuration, the door lock system partner must provide the following to the customer:

- API Base URL path to connect with the external Door Lock system, for example: 'https://doorlockhost.com/v1'.
- Authentication type for a secure connection (will be sent with every API request).
- Optional access areas that can be set in OPERA Cloud per Key.
- The ID of the Key Encoder devices. OPERA Cloud sends this Id with a request to create a room key.

5 Configurations

To properly use the post roomKeysOutbound API, the following configurations must be completed in the OPERA Cloud UI:

In OPERA Cloud PMS, configure an Outbound system interface for Outbound Type **GUESTKEY_GENERIC** per OPERA Cloud Property.

Configuring Outbound Systems

In OPERA Cloud PMS, configure required outbound domain/host entry to allow HTTPS connection from OPERA Cloud to the partner system.

Configuring Outbound domain/host

In OPERA Cloud PMS, configure a Property Interface for IFC TYPE DLS linked to the Outbound Door Lock Interface per OPERA Cloud Property.

Configuring Property Interface DLS IFC

In OPERA Cloud PMS Property Interface, the supported room key action types must be configured (New key, Duplicate key, One-shot key, Read key, Delete key) to display them in the OPERA Cloud PMS 'Create Key' screen.

Configuring room key action types

In OPERA Cloud PMS, configure if the Door Lock system is an **ONLINE KEY** System or **OFFLINE** System, or if it is a system providing PIN Codes instead of "hard keys." Related PMS Functionality will then be enabled.

Configuring DLS system type

In OPERA Cloud PMS, configure additional optional access area codes (KeyOptions). The user can choose these upon the create key request.

Configuring DLS access areas

Mandatory data the Door Lock system needs to retrieve to properly handle Room Keys:

To properly integrate to the OPERA Cloud Outbound system, the Door Lock system must know the Chain Code/Tenant Code and the unique HotelId for an OPERA Cloud property of the customer.

- **HotelId** - The Property Code that identifies the specific hotel. This is part of the URL path and is sent with every API message.
- **PMS RoomId** - The configured room numbers of the hotel. The roomId identifies the door lock for which the room key will be requested.

How to retrieve the HotelId and configured Room numbers:

- The customer can hand out the Chain Code, all hotelIds, and the configured room numbers that are used with the DLS integration.
- Alternatively retrieve the hotelIds and also the configured room numbers through API calls by subscribing to our Oracle Hospitality Integration Portal (OHIP) .

With OHIP, you have the ability to consume more than 3000 APIs and automatically retrieve specific necessary configurations (such as defined Room numbers) or retrieve additional Reservation or Guest information, which is not part of the outbound Door Lock API.

For more information about onboarding and working with OHIP, refer to the Getting Started section of the OHIP User Guide.

Retrieving a List of all HotelIds for a Chain

Use the OperationId: getResortChains (Postman Example)

Use this API to retrieve HotelId information for a selected Chain.

At least ONE HotelId (the Hotel's OPERA Cloud Property Code) must be obtained directly from the customer as this is used with the API call.

Sample: getResortChains

GET **HostName** /lov/v1/listOfValues/ResortChains

Response


```
{
    "listOfValues": {
        "items": [
            {
                "flexfields": [
                    {
                        "parameterName": "ChainCode",
                        "parameterValue": "OHIPSB"
                    },
                    {
                        "parameterName": "ChainName",
                        "parameterValue": "OHIPSB"
                    },
                    {
                        "parameterName": "ChainDesc",
                        "parameterValue": "OHIPSB"
                    }
                ],
                "code": "SAND01",
                "name": "OHIP Sandbox 1",
                "description": "OHIP Sandbox 1",
                "active": true
            },
            {
                "flexfields": [
                    {
                        "parameterName": "ChainCode",
                        "parameterValue": "OHIPSB"
                    },
                    {
                        "parameterName": "ChainName",
                        "parameterValue": "OHIPSB"
                    },
                    {
                        "parameterName": "ChainDesc",
                        "parameterValue": "OHIPSB"
                    }
                ],
                "code": "SAND02",
                "name": "OHIP Sandbox 2",
                "description": "OHIP Sandbox 2",
                "active": true
            },
            {
                "flexfields": [
                    {
                        "parameterName": "ChainCode",
                        "parameterValue": "OHIPSB"
                    },
                    {
                        "parameterName": "ChainName",
```

Response


```
                        "parameterValue": "OHIPSB"
                    },
                    {
                        "parameterName": "ChainDesc",
                        "parameterValue": "OHIPSB"
                    }
                ],
                "code": "SAND03",
                "name": "OHIP Sandbox 3",
                "description": "OHIP Sandbox 3",
                "active": true
            }
        ],
        "lovName": "ResortChains",
        "itemCount": 3
    }
}
```


With OHIP, you can retrieve the configured room numbers of a specific property, which you must configure on your system.

Suggested OHIP API Operation:

Retrieve a simple list of room number of the resort (HotelId).

Use the OperationId **getResortRoomNumbers** (Postman Example).

Use this API to fetch room numbers plus simple information for a selected HotelId. The response includes a list of all configured rooms at a property.

> **Note**
>
> Room numbers can have leading zeroes (for example, "010", "0101", "0002"). Since there is no translation of room numbers planned, the external system (and the Door Lock System) should support room numbers with leading zeroes.

Sample: getResortRoomNumbers

GET **HostName** /lov/v1/listOfValues/hotels/*HotelId*/resortRoomNumbers

Response


```
{
    "listOfValues": {
        "items": [
            {
                "flexfields": [
                    {
                        "parameterName": "Label",
                        "parameterValue": "JSUI"
                    },
                    {
                        "parameterName": "ShortDescription",
                        "parameterValue": "Junior Suite"
                    }
                ],
                "code": "1000",
                "name": "2 Double Beds and 1 Simple Bed",
                "description": "2 Double Beds and 1 Simple Bed",
                "active": true
            },
            {
                "flexfields": [
                    {
                        "parameterName": "Label",
                        "parameterValue": "JSUI"
                    },
                    {
                        "parameterName": "ShortDescription",
                        "parameterValue": "Junior Suite"
                    }
                ],
                "code": "1001",
                "name": "2 Double Beds and 1 Simple Bed",
                "description": "2 Double Beds and 1 Simple Bed",
                "active": true
            },
            {
                "flexfields": [
                    {
                        "parameterName": "Label",
                        "parameterValue": "JSUI"
                    },
                    {
                        "parameterName": "ShortDescription",
                        "parameterValue": "Junior Suite"
                    }
                ],
                "code": "1002",
                "name": "2 Double Beds and 1 Simple Bed",
                "description": "2 Double Beds and 1 Simple Bed",
```

Response

<pre>"active": true
},</pre>

Retrieve a detailed list of physical and/or pseudo rooms for a property:

Pseudo Rooms can be defined in OPERA Cloud PMS as rooms that are checked in for Conference Rooms. Related door Lock Ids are defined in the Door Lock System and can be linked to the Pseudo Rooms in OPERA Cloud Configuration.

Note that room numbers can have leading zeroes (for example, "010", "0101", "0002"). Since there is no translation of room numbers planned, the external system (and the Door Lock System) should support room numbers with leading zeroes.

use the OperationId: getRooms (Postman Example)

Use this API to fetch detailed information of rooms that exist for a specified property. The response will include a list of all existing physical and or pseudo rooms at that property.

Sample: getRooms

GET HostName /rm/config/v1/hotels/ ***hotelId*** /rooms?
limit=&offset=&physical=&pseudo=&roomClassCodes=&roomTypeCode=

Useful query parameters:

Parameter	Type	Description
hotelId	String	Unique ID of hotel.
limit	Number (max 20)	Indicates maximum number of records a Web Service should return.
offset	Number	Index or initial index of the set(page) being requested. If the index goes out of the bounds of the total set count, then no data will be returned.
physical	Boolean (true/false)	Is it a physical room type.
pseudo	Boolean (true/false)	Is it a pseudo room type.
roomClassCodes	Array	
roomTypeCodes	Array	
room	String	Room to be searched.

Response Listing all Rooms


```
{
    "rooms": [
        {
            "room": [

                {
                    "roomType": {
                        "pseudo": false,
                        "suite": false,
                        "roomClass": "STD",
                        "houseKeeping": true,
                        "minimumOccupancy": 1,
                        "maximumOccupancy": 1,
                        "accessible": false,
                        "roomType": "ECO",
                        "meetingRoom": false
                    },
                    "roomFeatures": [
                        {
                            "code": "QBED",
                            "description": "Queen Bed",
                            "orderSequence": 1
                        },
                        {
                            "code": "NEAR",
                            "description": "Near elevator",
                            "orderSequence": 11
                        },
                        {
                            "code": "PET",
                            "description": "Pet room",
                            "orderSequence": 15
                        }
                    ],
                    "roomDescription": "Economy Room",
                    "description": {
                        "defaultText": "Economy Room",
                        "translatedTexts": []
                    },
                    "accessible": true,
                    "roomId": "101",
                    "meetingRoom": true,
                    "roomComponents": [],
                    "connectingRooms": [
                        {
                            "roomType": {
                                "roomType": "ECO"
                            },
                            "description": {
                                "defaultText": "Economy Room"
                            },
```

Response Listing all Rooms


```
                        "roomId": "102"
                    }
                ],
                "rateAmount": {},
                "maximumOccupancy": 1,
                "sellSequence": 1,
                "ownerRoom": false,
                "keyOptions": [],
                "turndownService": false,
                "roomSection": {}
            },
            {
                "roomType": {
                    "pseudo": false,
                    "suite": false,
                    "roomClass": "STD",
                    "houseKeeping": true,
                    "minimumOccupancy": 1,
                    "maximumOccupancy": 1,
                    "accessible": false,
                    "roomType": "ECO",
                    "meetingRoom": false
                },
                "roomDescription": "Economy Room",
                "description": {
                    "defaultText": "Economy Room",
                    "translatedTexts": []
                },
                "accessible": true,
                "roomId": "102",
                "meetingRoom": false,
                "roomComponents": [],
                "connectingRooms": [
                    {
                        "roomType": {
                            "roomType": "ECO"
                        },
                        "description": {
                            "defaultText": "Economy Room"
                        },
                        "roomId": "101"
                    }
                ],
                "rateAmount": {},
                "maximumOccupancy": 1,
                "sellSequence": 1,
                "ownerRoom": false,
                "keyOptions": [],
                "turndownService": false,
                "roomSection": {}
            },
```

Response Listing all Rooms


```
{
    "roomType": {
        "pseudo": true,
        "suite": false,
        "roomClass": "ALL",
        "houseKeeping": false,
        "maximumOccupancy": 1,
        "accessible": false,
        "roomType": "PM",
        "meetingRoom": false
    },
    "roomDescription": "Posting Master",
    "description": {
        "defaultText": "PM IFC other 15",
        "translatedTexts": []
    },
    "accessible": false,
    "roomId": "9015",
    "meetingRoom": false,
    "roomComponents": [],
    "rateAmount": {},
    "maximumOccupancy": 1,
    "ownerRoom": false,
    "keyOptions": [],
    "turndownService": false,
    "roomSection": {}
},
{
    "roomType": {
        "pseudo": true,
        "suite": false,
        "roomClass": "PSE",
        "houseKeeping": false,
        "accessible": false,
        "roomType": "DPDP",
        "meetingRoom": false
    },
    "roomDescription": "DP",
    "description": {
        "defaultText": "DP",
        "translatedTexts": []
    },
    "accessible": false,
    "roomId": "9200",
    "meetingRoom": false,
    "roomComponents": [],
    "rateAmount": {},
    "ownerRoom": false,
    "keyOptions": [],
    "turndownService": false,
```

Response Listing all Rooms


```
                    "roomSection": {}
                },
                {
                    "roomType": {
                        "pseudo": true,
                        "suite": false,
                        "roomClass": "ALL",
                        "houseKeeping": false,
                        "building": "1",
                        "maximumOccupancy": 300,
                        "accessible": false,
                        "roomType": "BALLROOM",
                        "meetingRoom": true
                    },
                    "roomDescription": "Grand Ballroom",
                    "description": {
                        "defaultText": "West Sunset Room 1",
                        "translatedTexts": []
                    },
                    "building": "1",
                    "accessible": true,
                    "roomId": "BALLROOM",
                    "meetingRoom": true,
                    "roomComponents": [],
                    "rateAmount": {},
                    "ownerRoom": false,
                    "keyOptions": [],
                    "turndownService": false,
                    "roomSection": {}
                }
            ],
            "hotelId": "SAND01"
        }
```

Example Guest Room


```
{
                    "roomType": {
                        "pseudo": false,
                        "suite": false,
                        "roomClass": "STD",
                        "houseKeeping": true,
                        "minimumOccupancy": 1,
                        "maximumOccupancy": 1,
                        "accessible": false,
                        "roomType": "ECO",
                        "meetingRoom": false
                    },
                    "roomFeatures": [
                        {
                            "code": "QBED",
                            "description": "Queen Bed",
                            "orderSequence": 1
                        },
                        {
                            "code": "NEAR",
                            "description": "Near elevator",
                            "orderSequence": 11
                        },
                        {
                            "code": "PET",
                            "description": "Pet room",
                            "orderSequence": 15
                        }
                    ],
                    "roomDescription": "Economy Room",
                    "description": {
                        "defaultText": "Economy Room",
                        "translatedTexts": []
                    },
                    "accessible": true,
                    "roomId": "101",
                    "meetingRoom": true,
                    "roomComponents": [],
                    "connectingRooms": [
                        {
                            "roomType": {
                                "roomType": "ECO"
                            },
                            "description": {
                                "defaultText": "Economy Room"
                            },
                            "roomId": "102"
                        }
                    ],
                    "rateAmount": {},
                    "maximumOccupancy": 1,
```

Example Guest Room

```
                    "sellSequence": 1,
                    "ownerRoom": false,
                    "keyOptions": [],
                    "turndownService": false,
                    "roomSection": {}
                },
```

Example Post Master (Room Type PM)


```
{
                    "roomType": {
                        "pseudo": true,
                        "suite": false,
                        "roomClass": "ALL",
                        "houseKeeping": false,
                        "maximumOccupancy": 1,
                        "accessible": false,
                        "roomType": "PM",
                        "meetingRoom": false
                    },
                    "roomDescription": "Posting Master",
                    "description": {
                        "defaultText": "PM IFC other 15",
                        "translatedTexts": []
                    },
                    "accessible": false,
                    "roomId": "9015",
                    "meetingRoom": false,
                    "roomComponents": [],
                    "rateAmount": {},
                    "maximumOccupancy": 1,
                    "ownerRoom": false,
                    "keyOptions": [],
                    "turndownService": false,
                    "roomSection": {}
                },
                {
                    "roomType": {
                        "pseudo": true,
                        "suite": false,
                        "roomClass": "PSE",
                        "houseKeeping": false,
                        "accessible": false,
                        "roomType": "DPDP",
                        "meetingRoom": false
                    },
                    "roomDescription": "DP",
                    "description": {
                        "defaultText": "DP",
                        "translatedTexts": []
                    },
                    "accessible": false,
                    "roomId": "9200",
                    "meetingRoom": false,
                    "roomComponents": [],
                    "rateAmount": {},
                    "ownerRoom": false,
                    "keyOptions": [],
                    "turndownService": false,
```

Example Post Master (Room Type PM)

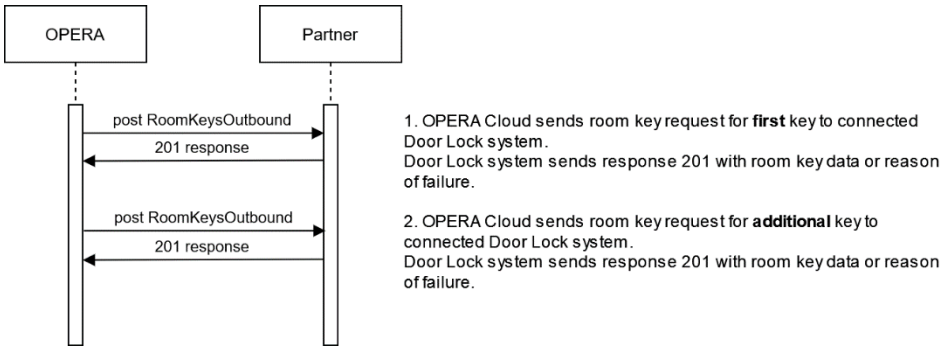
```
                "roomSection": {}
            },
```

6
Workflows

Workflow: Create New Room Key Plus Additional Room Key

A typical process to create a room key plus additional room key for a reservation is as follows:

1. **Sequence Diagram**



2. Description of Steps:

	Description	Sample Call

1	Request for a new room key or Pincode from OPERA Cloud PMS UI is sent to the door lock system.
	The door lock system must respond within 20 seconds with a "201" status message containing room key data or (in case of an unsuccessful request) a "201" response containing message information on why the request could not be processed.
	A new key can be requested at anytime. Even before the reservation checks in, the OPERA user can send a request for a new key (plus additional duplicate keys). It is expected that the door lock system supports requests for keys being valid in the future.
	Within the check-in process, the OPERA Cloud user is prompted to create room keys. The related "Make Key" form will open and the user can choose to create new or duplicate (additional) room key(s).
	After setting all room key options, such as the additional rooms and any associated Access Areas (KeyOptions) it should grant access to, the postRoomKeyOutbound API that includes the keyType value "New" and the specific encoderId will be transmitted to the door lock system.
	The door lock system will pass the request to the key encoder, which then processes the room key request.
	After the key encoder processed the request, the door lock system sends back a 201 http response message to OPERA Cloud containing the result of the request.
	If the request was successful, the door lock
		New Room Key Request
		POST {{HostName}}/fof/v1/hotels/{{HotelId}}/roomKeysOutbound
		
		`{`
		`  "keyType": "New",`
		`  "encoderTerminal": "TERM1",`
		`  "encoderId": "1",`
		`  "roomKeyType": "HardKey",`
		`  "keyUserType": "Guest",`
		`  "keyRequestTime": "2024-02-26T16:16:10.295Z",`
		`  "roomId": "101",`
		`  "oldRoomId": "",`
		`  "noOfKeys": 1,`
		`  "reservationIdList": [`
		`  {`
		`  "id": "{{ReservationId}}",`
		`  "type": "Reservation"`
		`  },`
		`  {`
		`  "id": "2468024",`
		`  "type": "Confirmation"`
		`  }`
		`  ],`
		`  "guestShareFlag": "false",`
		`  "OldGuestShareFlag": "",`
		`  "keyValidityStart": "2024-02-26T11:16:16.000Z",`
		`  "keyValidityEnd": "2024-02-27T12:00:00.000Z",`
		`  "keyTrack": {`
		`  "keyTrack1": "",`
		`  "keyTrack2": "",`
		`  "keyTrack3": ""`
		`  },`
		`  "additionalRooms": [`
		`  "102",`
		`  "103"`
		`  ],`
		`  "oldAdditionalRooms": [`
		`  "",`
		`  ""`
		`  ],`
		`  "userId": "PMSUser1",`
		`  "keyOptions": [`
		`  {`
		`  "keyOptionsCode": "SAU",`
		`  "keyOptionDescription": "Sauna"`
		`  },`
		`  {`
		`  "keyOptionsCode": "GAR",`
		`  "keyOptionDescription": "Garage"`
		`  }`
		`  ],`
		`  "cardSerialNumber": "",`

	Description	Sample Call

	system returns as much key data information as possible, such as the activated KeyOptions (Access areas), the time the key has been generated/ activated, and a short response text that appears to the user in OPERA Cloud.
	It is necessary to specify both the PMS 'encoderTerminal' and the Key Service system's coder (encoderId) in cases where more than one PMS workstation might be addressing one key coder.
	In OPERA Cloud, the Encoder ID will be configured so the PMS user can choose which encoder is activated to execute the related request.
	ORACLE Cloud PMS does NOT send another request command automatically should a negative response be received. The user sees the response and decides if another try should be made.
	With a request for a new room key (keyType: "New"), it is expected that all previously active room keys for this reservation/ room must be set inactive.
	Note: there will be ONE separate postRoomKeysOutbound call per room key to create. The "Number of keys" value is always "1"!
		<pre> "printData": {
		    "printText": "",
		    "printFile": "",
		    "printPicture": ""
		  }
		}</pre>
		New Room Key Response
		
		<pre>{
		  "roomId": "101",
		  "responseCode": "OK",
		  "encoderTerminal": "TERM1",
		  "roomKeyType": "HardKey",
		  "keyUserType": "Guest",
		  "keyValidityStart": "2024-02-26T11:16:16.000Z",
		  "keyValidityEnd": "2024-02-27T12:00:00.000Z",
		  "keyCreationTime": "2024-02-26T11:16:16.138Z",
		  "noOfKeys": "1",
		  "encoderId": "1",
		  "reservationIdList": [
		    {
		      "id": "{{ReservationId}}",
		      "type": "Reservation"
		    },
		    {
		      "id": "2468024",
		      "type": "Confirmation"
		    }
		  ],
		  "keyTrack": [
		    {
		      "keyTrack2": "98765A43B21C",
		    }
		  ],
		  "additionalRooms": [
		    "102",
		    "103"
		  ],
		  "responseText": "Key successfully created",
		  "keyOptions": [
		    {
		      "keyOptionsCode": "SAU",
		      "keyOptionDescription": "Sauna",
		      "isEnabled": "true",
		      "default": "false"
		    },
		    {
		      "keyOptionsCode": "GAR",
		      "keyOptionDescription": "Garage",</pre>

	Description	Sample Call

		<pre>"isEnabled": "true",
		"default": "false"
		},
		{
		"keyOptionsCode": "MAIN",
		"keyOptionDescription": "MainEntrance",
		"isEnabled": "true",
		"default": "true"
		}
		],
		"userId": "PMSUser1"
		}</pre>
		New Pincode Request
		POST {{HostName}}/fof/v1/hotels/{{HotelId}}/roomKeysOutbound
		
		<pre>{
		"keyType": "New",
		"encoderTerminal": "TERM1",
		"encoderId": "1",
		"roomKeyType": "PinCode",
		"keyUserType": "Guest",
		"keyRequestTime": "2024-02-26T16:16:10.295Z",
		"roomId": "101",
		"oldRoomId": "",
		"noOfKeys": 1,
		"reservationIdList": [
		{
		"id": "{{ReservationId}}",
		"type": "Reservation"
		},
		{
		"id": "2468024",
		"type": "Confirmation"
		}
		],
		"guestShareFlag": "false",
		"OldGuestShareFlag": "",
		"keyValidityStart": "2024-02-26T11:16:16.000Z",
		"keyValidityEnd": "2024-02-27T12:00:00.000Z",
		"keyTrack": {
		"keyTrack1": "",
		"keyTrack2": "",
		"keyTrack3": ""
		},
		"additionalRooms": [
		"",
		""
		],
		"oldAdditionalRooms": [
		"",</pre>

	Description	Sample Call

		<pre> ""
		  ],
		  "userId": "PMSUser1",
		  "keyOptions": [
		    {
		      "keyOptionsCode": "SAU",
		      "keyOptionDescription": "Sauna"
		    },
		    {
		      "keyOptionsCode": "GAR",
		      "keyOptionDescription": "Garage"
		    }
		  ],
		  "cardSerialNumber": "",
		  "printData": {
		    "printText": "",
		    "printFile": "",
		    "printPicture": ""
		  }
		}</pre>
		New Pincode Response
		
		<pre>{
		  "roomId": "101",
		  "responseCode": "OK",
		  "encoderTerminal": "TERM1",
		  "roomKeyType": "PinCode",
		  "keyUserType": "Guest",
		  "keyValidityStart": "2024-02-26T11:16:16.000Z",
		  "keyValidityEnd": "2024-02-27T12:00:00.000Z",
		  "keyCreationTime": "2024-02-26T11:16:16.138Z",
		  "noOfKeys": "1",
		  "encoderId": "1",
		  "reservationIdList": [
		    {
		      "id": "{{ReservationId}}",
		      "type": "Reservation"
		    },
		    {
		      "id": "2468024",
		      "type": "Confirmation"
		    }
		  ],
		  "responseText": "Pincode successfully created",
		  "keyTrack": {
		    "keyTrack3": "642086"
		  },
		  "keyOptions": [
		    {</pre>

	Description	Sample Call

		"keyOptionsCode": "SAU",
		"keyOptionDescription": "Sauna",
		"isEnabled": "true",
		"default": "false"
		},
		{
		"keyOptionsCode": "GAR",
		"keyOptionDescription": "Garage",
		"isEnabled": "true",
		"default": "false"
		},
		{
		"keyOptionsCode": "MAIN",
		"keyOptionDescription": "MainEntrance",
		"isEnabled": "true",
		"default": "true"
		}
		],
		"userId": "PMSUser1"
		}

	Description	Sample Call

2	Request for an additional room key from OPERA Cloud PMS UI.
	At any time after having successfully requested one room key (a New key), the OPERA Cloud PMS allows you to request additional room keys for the same room/reservation.
	By using the same postRoomKeysOutbound operation, OPERA Cloud PMS sends the request with the keyType value "Duplicate" to the door lock system.
	The door lock system will pass the request to the key encoder, which then processes the room key request.
	After key encoder processes the request, the door lock system sends back a 201 http response message to OPERA Cloud containing the result of the request.
	If the request was successful, then the door lock system sends back as much key data information as possible, such as the activated KeyOptions (Access areas), the time the key has been generated/ activated, and a short response text that appears to the user in OPERA Cloud.
	It is possible to send multiple requests for a "Duplicate" room key. OPERA Cloud PMS has no limit on sending such requests.
	If a Duplicate key is requested for a room or reservation where the door lock system does not have an active key, it is expected that the door lock system sends a corresponding response with a related resultText. This is a note that appears to the PMS user.
		Additional Room Key Request
		POST {{HostName}}/fof/v1/hotels/{{HotelId}}/roomKeysOutbound
		
		`{`
		`  "keyType": "Duplicate",`
		`  "encoderTerminal": "TERM1",`
		`  "encoderId": "1",`
		`  "roomKeyType": "HardKey",`
		`  "keyUserType": "Guest",`
		`  "keyRequestTime": "2024-02-26T16:16:10.295Z",`
		`  "roomId": "101",`
		`  "oldRoomId": "",`
		`  "noOfKeys": 1,`
		`  "reservationIdList": [`
		`  {`
		`  "id": "{{ReservationId}}",`
		`  "type": "Reservation"`
		`  },`
		`  {`
		`  "id": "2468024",`
		`  "type": "Confirmation"`
		`  }`
		`  ],`
		`  "guestShareFlag": "false",`
		`  "OldGuestShareFlag": "",`
		`  "keyValidityStart": "2024-02-26T11:16:16.000Z",`
		`  "keyValidityEnd": "2024-02-27T12:00:00.000Z",`
		`  "keyTrack": {`
		`  "keyTrack1": "",`
		`  "keyTrack2": "",`
		`  "keyTrack3": ""`
		`  },`
		`  "additionalRooms": [`
		`  "102",`
		`  "103"`
		`  ],`
		`  "oldAdditionalRooms": [`
		`  "",`
		`  ""`
		`  ],`
		`  "userId": "PMSUser1",`
		`  "keyOptions": [`
		`  {`
		`  "keyOptionsCode": "SAU",`
		`  "keyOptionDescription": "Sauna"`
		`  },`
		`  {`
		`  "keyOptionsCode": "GAR",`
		`  "keyOptionDescription": "Garage"`
		`  }`
		`  ],`
		`  "cardSerialNumber": "",`

	Description	Sample Call

		<pre> "printData": {
		    "printText": "",
		    "printFile": "",
		    "printPicture": ""
		  }
		}</pre>
		Additional Room Key Response
		
		<pre>{
		  "roomId": "101",
		  "responseCode": "OK",
		  "encoderTerminal": "TERM1",
		  "roomKeyType": "HardKey",
		  "keyUserType": "Guest",
		  "keyValidityStart": "2024-02-26T11:16:16.000Z",
		  "keyValidityEnd": "2024-02-27T12:00:00.000Z",
		  "keyCreationTime": "2024-02-26T11:16:16.138Z",
		  "noOfKeys": "1",
		  "encoderId": "1",
		  "reservationIdList": [
		    {
		      "id": "{{ReservationId}}",
		      "type": "Reservation"
		    },
		    {
		      "id": "2468024",
		      "type": "Confirmation"
		    }
		  ],
		  "keyTrack": [
		    {
		      "keyTrack2": "98765A43B21C",
		    }
		  ],
		  "responseText": "Key successfully created",
		  "keyOptions": [
		    {
		      "keyOptionsCode": "SAU",
		      "keyOptionDescription": "Sauna",
		      "isEnabled": "true",
		      "default": "false"
		    },
		    {
		      "keyOptionsCode": "GAR",
		      "keyOptionDescription": "Garage",
		      "isEnabled": "true",
		      "default": "false"
		    },
		    {</pre>

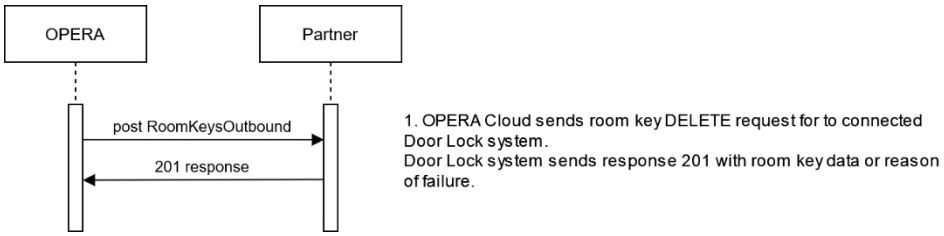
	Description	Sample Call

		<pre>"keyOptionsCode": "MAIN",
		"keyOptionDescription": "MainEntrance",
		"isEnabled": "true",
		"default": "true"
		}
		],
		"userId": "PMSUser1"
		}</pre>

Workflow: Delete Active Room Key(s)

A typical process to send a Delete room key request from out the OPERA Cloud UI for a reservation is as follows:

Sequence Diagram



Description of Steps:

	Description	Sample Call

1	Request for deleting all keys or Pin code for a reservation is sent to the door lock system from the OPERA Cloud PMS UI.
	The door lock system must respond within 20 seconds with a "201" status message that the room key(s) have been set inactive, or if an unsuccessful request, a "201" response containing message information on why the request could not be processed.
	A PostRoomKeysOutbound request message containing the keyType "Remove" requests the connected Door Lock system to set all active room keys, Pin code, or mobile key inactive for a specific reservation/room.
	After the Door lock system processed the request, it sends back a 201 http response message to OPERA Cloud containing the result of the request.
	ORACLE Cloud PMS does NOT send another request command automatically should a negative response be received.
	When the "Remove" request is sent during a reservation Check out and the Door Lock system sends back an unsuccessful response, OPERA Cloud will not attempt to repeat the request. In OPERA Cloud, the room key is marked as "deleted."
	The request message contains the reservationId and the roomId.
	It will be up to the Door Lock system to handle room keys based on the reservationId (recommended) or by the roomId.
	A "Remove" key request is typically sent at reservation check-out and is meant to
		Remove Room Key Request
		POST {{HostName}}/fof/v1/hotels/{{HotelId}}/roomKeysOutbound
		
		`{`
		`  "keyType": "Remove",`
		`  "encoderTerminal": "TERM1",`
		`  "encoderId": "1",`
		`  "roomKeyType": "HardKey",`
		`  "keyUserType": "Guest",`
		`  "keyRequestTime": "2024-02-26T16:16:10.295Z",`
		`  "roomId": "101",`
		`  "oldRoomId": "",`
		`  "noOfKeys": 1,`
		`  "reservationIdList": [`
		`  {`
		`  "id": "{{ReservationId}}",`
		`  "type": "Reservation"`
		`  },`
		`  {`
		`  "id": "2468024",`
		`  "type": "Confirmation"`
		`  }`
		`  ],`
		`  "guestShareFlag": "false",`
		`  "OldGuestShareFlag": "",`
		`  "keyValidityStart": "2024-02-26T11:16:16.000Z",`
		`  "keyValidityEnd": "2024-02-27T12:00:00.000Z",`
		`  "keyTrack": {`
		`  "keyTrack1": "",`
		`  "keyTrack2": "",`
		`  "keyTrack3": ""`
		`  },`
		`  "additionalRooms": [`
		`  "",`
		`  ""`
		`  ],`
		`  "oldAdditionalRooms": [`
		`  "",`
		`  ""`
		`  ],`
		`  "userId": "PMSUser1",`
		`  "keyOptions": [`
		`  {`
		`  "keyOptionsCode": "SAU",`
		`  "keyOptionDescription": "Sauna"`
		`  },`
		`  {`
		`  "keyOptionsCode": "GAR",`
		`  "keyOptionDescription": "Garage"`
		`  }`
		`  ],`
		`  "cardSerialNumber": "",`

	Description	Sample Call

	set all active room keys inactive.
	When the Door Lock system supports it, a key might be valid for a defined time after the Valid Until Date/ Time.
	OPERA PMS has no control on a room key being invalid. The "Remove" keyType in the request call only indicates a "request" to set a room key inactive.
		<pre> "printData": {
		    "printText": "",
		    "printFile": "",
		    "printPicture": ""
		  }
		}</pre>
		Remove Room Key Response
		
		<pre>{
		  "roomId": "101",
		  "responseCode": "OK",
		  "encoderTerminal": "TERM1",
		  "roomKeyType": "HardKey",
		  "keyUserType": "Guest",
		  "keyValidityStart": "2024-02-26T11:16:16.000Z",
		  "keyValidityEnd": "2024-02-27T12:00:00.000Z",
		  "keyCreationTime": "2024-02-26T11:16:16.138Z",
		  "noOfKeys": "1",
		  "encoderId": "1",
		  "reservationIdList": [
		    {
		      "id": "{{ReservationId}}",
		      "type": "Reservation"
		    },
		    {
		      "id": "2468024",
		      "type": "Confirmation"
		    }
		  ],
		  "responseText": "Key successfully deleted/
		removed",
		  "userId": "PMSUser1"
		}</pre>
		Remove Pincode Request
		POST {{HostName}}/fof/v1/hotels/{{HotelId}}/roomKeysOutbound
		
		<pre>{
		  "keyType": "Remove",
		  "encoderTerminal": "TERM1",
		  "encoderId": "1",
		  "roomKeyType": "PinCode",
		  "keyUserType": "Guest",
		  "keyRequestTime": "2024-02-26T16:16:10.295Z",
		  "roomId": "101",
		  "oldRoomId": "",
		  "noOfKeys": 1,
		  "reservationIdList": [</pre>

Description	Sample Call


```
    {
      "id": "{{ReservationId}}",
      "type": "Reservation"
    },
    {
      "id": "2468024",
      "type": "Confirmation"
    }
  ],
  "guestShareFlag": "false",
  "OldGuestShareFlag": "",
  "keyValidityStart": "2024-02-26T11:16:16.000Z",
  "keyValidityEnd": "2024-02-27T12:00:00.000Z",
  "keyTrack": {
    "keyTrack1": "",
    "keyTrack2": "",
    "keyTrack3": ""
  },
  "additionalRooms": [
    "",
    ""
  ],
  "oldAdditionalRooms": [
    "",
    ""
  ],
  "userId": "PMSUser1",
  "keyOptions": [
    {
      "keyOptionsCode": "SAU",
      "keyOptionDescription": "Sauna"
    },
    {
      "keyOptionsCode": "GAR",
      "keyOptionDescription": "Garage"
    }
  ],
  "cardSerialNumber": "",
  "printData": {
    "printText": "",
    "printFile": "",
    "printPicture": ""
  }
}
```


Remove Pincode Response


```
{
  "roomId": "101",
  "responseCode": "OK",
```

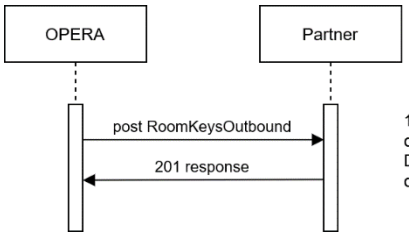
	Description	Sample Call

		"encoderTerminal": "TERM1",
		"roomKeyType": "PinCode",
		"keyUserType": "Guest",
		"keyValidityStart": "2024-02-26T11:16:16.000Z",
		"keyValidityEnd": "2024-02-27T12:00:00.000Z",
		"keyCreationTime": "2024-02-26T11:16:16.138Z",
		"noOfKeys": "1",
		"encoderId": "1",
		"reservationIdList": [
		{
		"id": "{{ReservationId}}",
		"type": "Reservation"
		},
		{
		"id": "2468024",
		"type": "Confirmation"
		}
		],
		"responseText": "Pincode successfully deleted/
		removed",
		"keyTrack": {
		"keyTrack3": "642086"
		},
		"keyOptions": [
		{
		"keyOptionsCode": "SAU",
		"keyOptionDescription": "Sauna",
		"isEnabled": "true",
		"default": "false"
		},
		{
		"keyOptionsCode": "GAR",
		"keyOptionDescription": "Garage",
		"isEnabled": "true",
		"default": "false"
		},
		{
		"keyOptionsCode": "MAIN",
		"keyOptionDescription": "MainEntrance",
		"isEnabled": "true",
		"default": "true"
		}
		],
		"userId": "PMSUser1"
		}

Workflow: Create a One-Shot Room Key

Follow this process to request a room key or a Pin code that is not related to a guest reservation and that is only valid for a specific time.

Sequence Diagram



1. OPERA Cloud sends room key request for a **One-Shot** room key to connected Door Lock system.
Door Lock system sends response 201 with room key data or reason of failure.

Description of Steps:

	Description	Sample Call

1	Request for a One-Shot room key or Pin code is sent to the door lock system from the OPERA Cloud PMS UI.
	The door lock system must respond within 20 seconds with a "201" status message containing room key data, or if an unsuccessful request, a "201" response containing message information on why the request could not be processed.
	A "One-Shot" room key is a key that is usually valid only for a few hours and would usually be created for showing a guest room or for workers who must repair something in the guest room.
	One-Shot keys are usually NOT related to a reservation. However, they can be created for rooms which have an active reservation and also active guest room keys.
	A One-Shot room key should not make other active room keys for a room invalid.
		One-Shot Key Request
		POST {{HostName}}/fof/v1/hotels/{{HotelId}}/roomKeysOutbound
		
		`{`
		`  "keyType": "OneShot",`
		`  "encoderTerminal": "TERM1",`
		`  "encoderId": "1",`
		`  "roomKeyType": "HardKey",`
		`  "keyUserType": "Guest",`
		`  "keyRequestTime": "2024-02-26T16:16:10.295Z",`
		`  "roomId": "101",`
		`  "oldRoomId": "",`
		`  "noOfKeys": 1,`
		`  "reservationIdList": [`
		`  {`
		`  "id": "{{ReservationId}}",`
		`  "type": "Reservation"`
		`  },`
		`  {`
		`  "id": "2468024",`
		`  "type": "Confirmation"`
		`  }`
		`  ],`
		`  "guestShareFlag": "false",`
		`  "OldGuestShareFlag": "",`
		`  "keyValidityStart": "2024-02-26T11:16:16.000Z",`
		`  "keyValidityEnd": "2024-02-27T12:00:00.000Z",`
		`  "keyTrack": {`
		`  "keyTrack1": "",`
		`  "keyTrack2": "",`
		`  "keyTrack3": ""`
		`  },`
		`  "additionalRooms": [`
		`  "",`
		`  ""`
		`  ],`
		`  "oldAdditionalRooms": [`
		`  "",`
		`  ""`
		`  ],`
		`  "userId": "PMSUser1",`
		`  "keyOptions": [`
		`  {`
		`  "keyOptionsCode": "SAU",`
		`  "keyOptionDescription": "Sauna"`
		`  },`
		`  {`
		`  "keyOptionsCode": "GAR",`
		`  "keyOptionDescription": "Garage"`
		`  }`
		`  ],`
		`  "cardSerialNumber": "",`

	Description	Sample Call

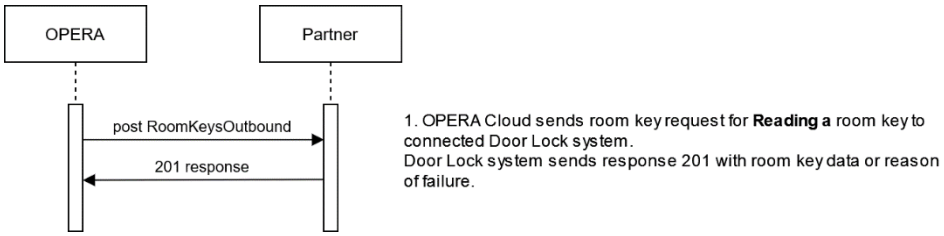
		"printData": {
		"printText": "",
		"printFile": "",
		"printPicture": ""
		}
		}
		
		One-Shot Key Response
		
		{
		"roomId": "101",
		"responseCode": "OK",
		"encoderTerminal": "TERM1",
		"roomKeyType": "HardKey",
		"keyUserType": "Guest",
		"keyValidityStart": "2024-02-26T11:16:16.000Z",
		"keyValidityEnd": "2024-02-27T12:00:00.000Z",
		"keyCreationTime": "2024-02-26T11:16:16.138Z",
		"noOfKeys": "1",
		"encoderId": "1",
		"reservationIdList": [
		{
		"id": "{{ReservationId}}",
		"type": "Reservation"
		},
		{
		"id": "2468024",
		"type": "Confirmation"
		}
		],
		"keyTrack": [
		{
		"keyTrack2": "98765A43B21C",
		}
		],
		"responseText": "One Shot Key successfully created",
		"keyOptions": [
		{
		"keyOptionsCode": "SAU",
		"keyOptionDescription": "Sauna",
		"isEnabled": "true",
		"default": "false"
		},
		{
		"keyOptionsCode": "GAR",
		"keyOptionDescription": "Garage",
		"isEnabled": "true",
		"default": "false"
		},

	Description	Sample Call

		<pre>{
		"keyOptionsCode": "MAIN",
		"keyOptionDescription": "MainEntrance",
		"isEnabled": "true",
		"default": "true"
		}
		],
		"userId": "PMSUser1"
		}</pre>

Workflow: Read a Room Key on a Specific Key Encoder

Sequence Diagram



Description of Steps:

	Description	Sample Call

1	Request for a "Read" room key is sent to the door lock system from the OPERA Cloud PMS UI.
	The door lock system has to respond within 20 seconds with a "201" status message containing room key data, or if an unsuccessful request, a "201" response containing message information on why the request could not be processed.
	A "ReadKey" request from OPERA Cloud is a request to read the data of a room key.
	Usually this request will be made by a hotel front desk user to see if a room key is still valid and is properly working.
	Upon receiving the request call, the Door Lock system activates the key encoder mentioned in the request and responds with the available room key data.
	OPERA Cloud displays some major key data to the user.
	The received key data will not be stored in OPERA Cloud and appears to the user for information purposes only.
		Read Room Key Request
		POST {{HostName}}/fof/v1/hotels/{{HotelId}}/roomKeysOutbound
		
		`{`
		`  "encoderId": "1",`
		`  "encoderTerminal": "TERM1",`
		`  "noOfKeys": 1,`
		`  "keyType": "Read",`
		`  "keyRequestTime": "2024-03-12T14:16:04.000Z",`
		`  "roomKeyType": "HardKey",`
		`  "keyUserType": "Guest",`
		`  "userId": "PMSUser1"`
		`}`
		
		Read Room Key Response
		
		`{`
		`  "keyType": "Read",`
		`  "encoderTerminal": "TERM1",`
		`  "encoderId": "1",`
		`  "roomKeyType": "HardKey",`
		`  "keyUserType": "Guest",`
		`  "keyRequestTime": "",`
		`  "roomId": "101",`
		`  "reservationIdList": [`
		`  {`
		`  "id": "{{ReservationId}}",`
		`  "type": "Reservation"`
		`  },`
		`  {`
		`  "id": "2468024",`
		`  "type": "Confirmation"`
		`  }`
		`  ],`
		`  "guestShareFlag": "false",`
		`  "OldGuestShareFlag": "",`
		`  "keyValidityStart": "2024-02-26T11:16:16.000Z",`
		`  "keyValidityEnd": "2024-02-27T12:00:00.000Z",`
		`  "additionalRooms": [`
		`  "102",`
		`  "103"`
		`  ],`
		`  "userId": "PMSUser1",`
		`  "keyOptions": [`
		`  {`
		`  "keyOptionsCode": "SAU",`
		`  "keyOptionDescription": "Sauna",`
		`  "isEnabled": "true",`
		`  "default": "false"`

	Description	Sample Call

		<pre>},
		{
		"keyOptionsCode": "GAR",
		"keyOptionDescription": "Garage",
		"isEnabled": "true",
		"default": "false"
		},
		{
		"keyOptionsCode": "MAIN",
		"keyOptionDescription": "MainEntrance",
		"isEnabled": "true",
		"default": "true"
		}
		]
		}</pre>

Workflow: Handling of Room Keys on Room Move

OPERA Cloud supports two ways of informing the connected Door Lock system about guests moving to another room and the change of room keys.

1. Room move for an "offline" Door Lock System
2. Room move for an "online" Door Lock System

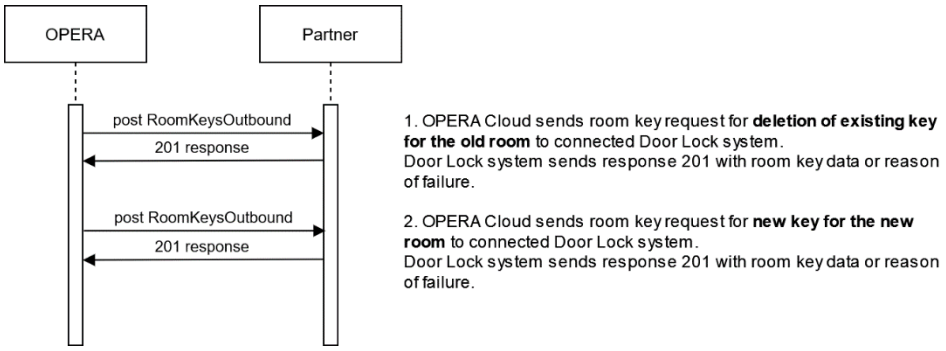
When the Door lock system is an offline system:

An "offline" Door Lock system typically handles room keys based on the room numbers. Room key data is written on the physical key.

For a room move of a checked in reservation, OPERA Cloud sends out a request with keyType "Remove" for the old room followed by a request with keyType "New" for the new room containing the new Room number as well as new "additionalRooms" and/or new "KeyOptions."

Additional key requests for the new room are sent after receiving a successful response for a New key.

Sequence Diagram



Description of Steps:

	Description	Sample Call

1	Request to "Remove" room key(s) for the old room is sent to the door lock system from OPERA Cloud.
	The door lock system must respond within 20 seconds with a "201" status message that the room key(s) have been set to inactive, or if an unsuccessful request, a "201" response containing message information on why the request could not be processed.
	As the room move is processing in OPERA Cloud PMS, the guest moves to another room, and the old room key becomes invalid and should no longer work.
	The guest might or might not hand the room key back to the hotel staff even if the guest receives a new room key.
		Offline - Room Move Remove Room Key Request
		POST {{HostName}}/fof/v1/hotels/{{HotelId}}/roomKeysOutbound
		
		{
		"keyType": "Remove",
		"encoderTerminal": "TERM1",
		"encoderId": "1",
		"roomKeyType": "HardKey",
		"keyUserType": "Guest",
		"keyRequestTime": "2024-02-26T16:16:10.295Z",
		"roomId": "101",
		"oldRoomId": "",
		"noOfKeys": 1,
		"reservationIdList": [
		{
		"id": "{{ReservationId}}",
		"type": "Reservation"
		},
		{
		"id": "2468024",
		"type": "Confirmation"
		}
		],
		"guestShareFlag": "false",
		"OldGuestShareFlag": "",
		"keyValidityStart": "2024-02-26T11:16:16.000Z",
		"keyValidityEnd": "2024-02-27T12:00:00.000Z",
		"keyTrack": {
		"keyTrack1": "",
		"keyTrack2": "",
		"keyTrack3": ""
		},
		"additionalRooms": [
		"",
		""
		],
		"oldAdditionalRooms": [
		"",
		""
		],
		"userId": "PMSUser1",
		"keyOptions": [
		{
		"keyOptionsCode": "SAU",
		"keyOptionDescription": "Sauna"
		},
		{
		"keyOptionsCode": "GAR",
		"keyOptionDescription": "Garage"
		}
		],
		"cardSerialNumber": "",

	Description	Sample Call

		<pre> "printData": {
		    "printText": "",
		    "printFile": "",
		    "printPicture": ""
		  }
		}</pre>
		Offline - Room Move Remove Room Key Response
		
		<pre>{
		  "roomId": "101",
		  "responseCode": "OK",
		  "encoderTerminal": "TERM1",
		  "roomKeyType": "HardKey",
		  "keyUserType": "Guest",
		  "keyValidityStart": "2024-02-26T11:16:16.000Z",
		  "keyValidityEnd": "2024-02-27T12:00:00.000Z",
		  "keyCreationTime": "2024-02-26T11:16:16.138Z",
		  "noOfKeys": "1",
		  "encoderId": "1",
		  "reservationIdList": [
		    {
		      "id": "{{ReservationId}}",
		      "type": "Reservation"
		    },
		    {
		      "id": "2468024",
		      "type": "Confirmation"
		    }
		  ],
		  "responseText": "Key successfully deleted/
		removed",
		  "userId": "PMSUser1"
		}</pre>

	Description	Sample Call

2	Now a request for a "New" room key or Pincode for the new room is sent to the Door Lock system from OPERA Cloud.
	The door lock system must respond within 20 seconds with a "201" status message that the room key(s) have been set to inactive, or if an unsuccessful request, a "201" response containing message information on why the request could not be processed.
	With the move to another Room, the guest receives a room key with the new room data.
	This can either be a new room key or Pincode, or the "old" room key can be used.
	However, the room key must be encoded with the new data (roomId, Additional rooms, Key Options).
	Optional: Additional room keys can also be requested.
		Offline - Room Move New Room Key Request
		POST {{HostName}}/fof/v1/hotels/{{HotelId}}/roomKeysOutbound
		
		{
		"keyType": "New",
		"encoderTerminal": "TERM1",
		"encoderId": "1",
		"roomKeyType": "HardKey",
		"keyUserType": "Guest",
		"keyRequestTime": "2024-02-26T16:16:10.295Z",
		"roomId": "201",
		"oldRoomId": "",
		"noOfKeys": 1,
		"reservationIdList": [
		{
		"id": "{{ReservationId}}",
		"type": "Reservation"
		},
		{
		"id": "2468024",
		"type": "Confirmation"
		}
		],
		"guestShareFlag": "false",
		"OldGuestShareFlag": "",
		"keyValidityStart": "2024-02-26T11:16:16.000Z",
		"keyValidityEnd": "2024-02-27T12:00:00.000Z",
		"keyTrack": {
		"keyTrack1": "",
		"keyTrack2": "",
		"keyTrack3": ""
		},
		"additionalRooms": [
		"202",
		"203"
		],
		"oldAdditionalRooms": [
		"",
		""
		],
		"userId": "PMSUser1",
		"keyOptions": [
		{
		"keyOptionsCode": "SAU",
		"keyOptionDescription": "Sauna"
		},
		{
		"keyOptionsCode": "GAR",
		"keyOptionDescription": "Garage"
		}
		],
		"cardSerialNumber": "",

	Description	Sample Call

		<pre> "printData": {
		    "printText": "",
		    "printFile": "",
		    "printPicture": ""
		  }
		}</pre>
		Offline - Room Move New Room Key Response
		
		<pre>{
		  "roomId": "201",
		  "responseCode": "OK",
		  "encoderTerminal": "TERM1",
		  "roomKeyType": "HardKey",
		  "keyUserType": "Guest",
		  "keyValidityStart": "2024-02-26T11:16:16.000Z",
		  "keyValidityEnd": "2024-02-27T12:00:00.000Z",
		  "keyCreationTime": "2024-02-26T11:16:16.138Z",
		  "noOfKeys": "1",
		  "encoderId": "1",
		  "reservationIdList": [
		    {
		      "id": "{{ReservationId}}",
		      "type": "Reservation"
		    },
		    {
		      "id": "2468024",
		      "type": "Confirmation"
		    }
		  ],
		  "keyTrack": [
		    {
		      "keyTrack2": "98765A43B21C",
		    }
		  ],
		  "additionalRooms": [
		    "202",
		    "203"
		  ],
		  "responseText": "Key successfully created",
		  "keyOptions": [
		    {
		      "keyOptionsCode": "SAU",
		      "keyOptionDescription": "Sauna",
		      "isEnabled": "true",
		      "default": "false"
		    },
		    {
		      "keyOptionsCode": "GAR",
		      "keyOptionDescription": "Garage",
		      "isEnabled": "true",</pre>

	Description	Sample Call

		<pre>"default": "false"
		},
		{
		"keyOptionsCode": "MAIN",
		"keyOptionDescription": "MainEntrance",
		"isEnabled": "true",
		"default": "true"
		}
		],
		"userId": "PMSUser1"
		}</pre>

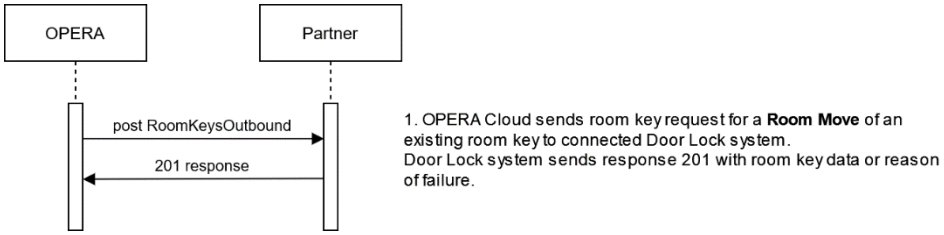
When the Door lock system is an online system:

The room keys can be handled based on the room and reservationId. Key data is not written on the physical key but stored in the Door Lock System database.

In this case, OPERA Cloud sends out a request with keyType "RoomMove" for the existing reservationId and sends the old and the new room number as well as the new "additionalRooms and/or new "keyOptions."

It is expected that the Door Lock system will update the existing room key data with the new roomId plus optionally update it with "additionalRooms" and "keyOptions."

Sequence Diagram



Description of Steps:

	Description	Sample Call

1	Request for a "RoomMove" of existing active room key(s) or Pin code is sent to the door lock system from the OPERA Cloud PMS UI.
	The door lock system must respond within 20 seconds with a "201" status message that the room key(s) have been set to inactive, or if an unsuccessful request, a "201" response containing message information on why the request could not be processed.
	As the existing room Key information is not stored on the room key(s) itself but stored on Door Lock Systems database, the new room data (new roomId, new additional rooms, and new KeyOptions) must be updated on the database only. The new room data does not need to be encoded on the room key(s).
	The hotel guest does not need a new room key(s) or Pincode, but the guest can use the existing one for the old room.
	The old room must no longer be accessible with the room key(s) or Pin code after successful processing of the "RoomMove" request.
		Online - Room Move Room Key Request
		POST {{HostName}}/fof/v1/hotels/{{HotelId}}/roomKeysOutbound
		
		{
		"keyType": "RoomMove",
		"encoderTerminal": "TERM1",
		"encoderId": "1",
		"roomKeyType": "HardKey",
		"keyUserType": "Guest",
		"keyRequestTime": "2024-02-26T16:16:10.295Z",
		"roomId": "201",
		"oldRoomId": "101",
		"noOfKeys": 1,
		"reservationIdList": [
		{
		"id": "{{ReservationId}}",
		"type": "Reservation"
		},
		{
		"id": "2468024",
		"type": "Confirmation"
		}
		],
		"guestShareFlag": "false",
		"OldGuestShareFlag": "",
		"keyValidityStart": "2024-02-26T11:16:16.000Z",
		"keyValidityEnd": "2024-02-27T12:00:00.000Z",
		"keyTrack": {
		"keyTrack1": "",
		"keyTrack2": "",
		"keyTrack3": ""
		},
		"additionalRooms": [
		"202",
		"203"
		],
		"oldAdditionalRooms": [
		"102",
		"103"
		],
		"userId": "PMSUser1",
		"keyOptions": [
		{
		"keyOptionsCode": "SAU",
		"keyOptionDescription": "Sauna"
		},
		{
		"keyOptionsCode": "GAR",
		"keyOptionDescription": "Garage"
		}
		],
		"cardSerialNumber": "",

	Description	Sample Call

		<pre> "printData": {
		    "printText": "",
		    "printFile": "",
		    "printPicture": ""
		  }
		}</pre>
		Online - Room Move Room Key Response
		
		<pre>{
		  "roomId": "201",
		  "responseCode": "OK",
		  "encoderTerminal": "TERM1",
		  "roomKeyType": "HardKey",
		  "keyUserType": "Guest",
		  "keyValidityStart": "2024-02-26T11:16:16.000Z",
		  "keyValidityEnd": "2024-02-27T12:00:00.000Z",
		  "keyCreationTime": "2024-02-26T11:16:16.138Z",
		  "noOfKeys": "1",
		  "encoderId": "1",
		  "reservationIdList": [
		    {
		      "id": "{{ReservationId}}",
		      "type": "Reservation"
		    },
		    {
		      "id": "2468024",
		      "type": "Confirmation"
		    }
		  ],
		  "keyTrack": [
		    {
		      "keyTrack1": "",
		      "keyTrack2": "98765A43B21C",
		      "keyTrack3": ""
		    }
		  ],
		  "responseText": "Key successfully changed",
		  "additionalRooms": [
		    "202",
		    "203"
		  ],
		  "keyOptions": [
		    {
		      "keyOptionsCode": "SAU",
		      "keyOptionDescription": "Sauna",
		      "isEnabled": "true",
		      "default": "false"
		    },
		    {</pre>

	Description	Sample Call

		<pre>"keyOptionsCode": "GAR",
		"keyOptionDescription": "Garage",
		"isEnabled": "true",
		"default": "false"
		},
		{
		"keyOptionsCode": "MAIN",
		"keyOptionDescription": "MainEntrance",
		"isEnabled": "true",
		"default": "true"
		}
		],
		"userId": "PMSUser1"
		}</pre>

Workflow: Handling of Room Keys at Reservation Stay Changes

OPERA Cloud supports two ways of informing the connected Door Lock system about changes of Departure date, which forces the adjustment of the Valid Until Dates for existing room keys.

1. Modify Stay information for an "offline" Door Lock System
2. Modify Stay information for an "online" Door Lock System

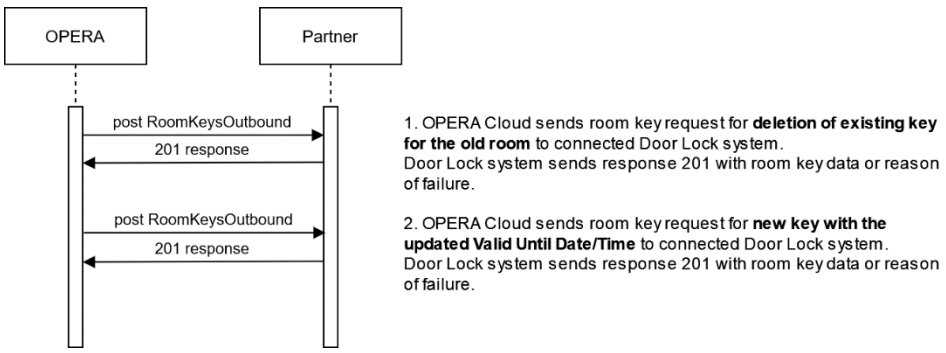
When the Door Lock system is an offline system:

An "offline" Door Lock system typically handles room keys based on the room numbers. Room key data will be written on the physical key.

For a change of Departure Date of a checked in reservation, OPERA Cloud sends out a request with keyType "Remove" for the old room followed by a request with keyType "New" for the new room containing the new Room number as well as new "additionalRooms" and/or new "KeyOptions."

Additional key requests for the new room must be sent after receiving a successful response for a New key.

Sequence Diagram



Description of Steps:

	Description	Sample Call

1	Request to "Remove" room key(s) for the old room is sent to the door lock system from OPERA Cloud.
	The door lock system must respond within 20 seconds with a "201" status message that the room key(s) have been set to inactive, or if an unsuccessful request, a "201" response containing message information on why the request could not be processed.
	With a change of the Departure Date of a checked in reservation, the room key must be updated. As the new ValidUntilDate must be written to the room key, the current key becomes invalid with the "Remove" room key request.
	The guest might or might not hand the room key back to the hotel staff even if the guest receives a new room key.
		Offline - Modify Stay Remove Room Key Request
		POST {{HostName}}/fof/v1/hotels/{{HotelId}}/roomKeysOutbound
		
		{
		"keyType": "Remove",
		"encoderTerminal": "TERM1",
		"encoderId": "1",
		"roomKeyType": "HardKey",
		"keyUserType": "Guest",
		"keyRequestTime": "2024-02-26T16:16:10.295Z",
		"roomId": "101",
		"oldRoomId": "",
		"noOfKeys": 1,
		"reservationIdList": [
		{
		"id": "{{ReservationId}}",
		"type": "Reservation"
		},
		{
		"id": "2468024",
		"type": "Confirmation"
		}
		],
		"guestShareFlag": "false",
		"OldGuestShareFlag": "",
		"keyValidityStart": "2024-02-26T11:16:16.000Z",
		"keyValidityEnd": "2024-02-27T12:00:00.000Z",
		"keyTrack": {
		"keyTrack1": "",
		"keyTrack2": "",
		"keyTrack3": ""
		},
		"additionalRooms": [
		"",
		""
		],
		"oldAdditionalRooms": [
		"",
		""
		],
		"userId": "PMSUser1",
		"keyOptions": [
		{
		"keyOptionsCode": "SAU",
		"keyOptionDescription": "Sauna"
		},
		{
		"keyOptionsCode": "GAR",
		"keyOptionDescription": "Garage"
		}
		],
		"cardSerialNumber": "",

	Description	Sample Call

		<pre> "printData": {
		    "printText": "",
		    "printFile": "",
		    "printPicture": ""
		  }
		}</pre>
		Offline - Modify Stay Remove Room Key Response
		
		<pre>{
		  "roomId": "101",
		  "responseCode": "OK",
		  "encoderTerminal": "TERM1",
		  "roomKeyType": "HardKey",
		  "keyUserType": "Guest",
		  "keyValidityStart": "2024-02-26T11:16:16.000Z",
		  "keyValidityEnd": "2024-02-27T12:00:00.000Z",
		  "keyCreationTime": "2024-02-26T11:16:16.138Z",
		  "noOfKeys": "1",
		  "encoderId": "1",
		  "reservationIdList": [
		    {
		      "id": "{{ReservationId}}",
		      "type": "Reservation"
		    },
		    {
		      "id": "2468024",
		      "type": "Confirmation"
		    }
		  ],
		  "responseText": "Key successfully deleted/
		removed",
		  "userId": "PMSUser1"
		}</pre>

	Description	Sample Call

2	Now a request for a "New" room key or Pincode for the new room is sent to the Door Lock system from OPERA Cloud.
	The door lock system must respond within 20 seconds with a "201" status message that the room key(s) have been set to inactive, or if an unsuccessful request, a "201" response containing message information on why the request could not be processed.
	With the change of the Departure Date, the guest receives a new room key with the new "validUntilDate" and receives new "additionalRooms" and/or "keyOptions" values.
	This can either be a new room key or Pincode, or the "old" room key can be used.
	However, the room key must be encoded with the new data (roomId, Additional rooms, Key Options).
	Optional: Additional room keys can also be requested.
		Offline - Modify Stay New Room Key Request
		POST {{HostName}}/fof/v1/hotels/{{HotelId}}/roomKeysOutbound
		
		`{`
		`"keyType": "New",`
		`"encoderTerminal": "TERM1",`
		`"encoderId": "1",`
		`"roomKeyType": "HardKey",`
		`"keyUserType": "Guest",`
		`"keyRequestTime": "2024-02-26T16:16:10.295Z",`
		`"roomId": "101",`
		`"oldRoomId": "",`
		`"noOfKeys": 1,`
		`"reservationIdList": [`
		`{`
		`"id": "{{ReservationId}}",`
		`"type": "Reservation"`
		`},`
		`{`
		`"id": "2468024",`
		`"type": "Confirmation"`
		`}`
		`],`
		`"guestShareFlag": "false",`
		`"OldGuestShareFlag": "",`
		`"keyValidityStart": "2024-02-26T11:16:16.000Z",`
		`"keyValidityEnd": "2024-02-27T12:00:00.000Z",`
		`"keyTrack": {`
		`"keyTrack1": "",`
		`"keyTrack2": "",`
		`"keyTrack3": ""`
		`},`
		`"additionalRooms": [`
		`"102",`
		`"103"`
		`],`
		`"oldAdditionalRooms": [`
		`"",`
		`""`
		`],`
		`"userId": "PMSUser1",`
		`"keyOptions": [`
		`{`
		`"keyOptionsCode": "SAU",`
		`"keyOptionDescription": "Sauna"`
		`},`
		`{`
		`"keyOptionsCode": "GAR",`
		`"keyOptionDescription": "Garage"`
		`}`
		`],`
		`"cardSerialNumber": "",`

	Description	Sample Call

		<pre> "printData": {
		    "printText": "",
		    "printFile": "",
		    "printPicture": ""
		  }
		}</pre>
		Offline - Modify Stay New Room Key Response
		
		<pre>{
		  "roomId": "101",
		  "responseCode": "OK",
		  "encoderTerminal": "TERM1",
		  "roomKeyType": "HardKey",
		  "keyUserType": "Guest",
		  "keyValidityStart": "2024-02-26T11:16:16.000Z",
		  "keyValidityEnd": "2024-02-29T12:00:00.000Z",
		  "keyCreationTime": "2024-02-26T11:16:16.138Z",
		  "noOfKeys": "1",
		  "encoderId": "1",
		  "reservationIdList": [
		    {
		      "id": "{{ReservationId}}",
		      "type": "Reservation"
		    },
		    {
		      "id": "2468024",
		      "type": "Confirmation"
		    }
		  ],
		  "keyTrack": [
		    {
		      "keyTrack2": "98765A43B21C",
		    }
		  ],
		  "additionalRooms": [
		    "102",
		    "103"
		  ],
		  "responseText": "Key successfully created",
		  "keyOptions": [
		    {
		      "keyOptionsCode": "SAU",
		      "keyOptionDescription": "Sauna",
		      "isEnabled": "true",
		      "default": "false"
		    },
		    {
		      "keyOptionsCode": "GAR",
		      "keyOptionDescription": "Garage",</pre>

	Description	Sample Call

		<pre>"isEnabled": "true",
		"default": "false"
		},
		{
		"keyOptionsCode": "MAIN",
		"keyOptionDescription": "MainEntrance",
		"isEnabled": "true",
		"default": "true"
		}
		],
		"userId": "PMSUser1"
		}</pre>

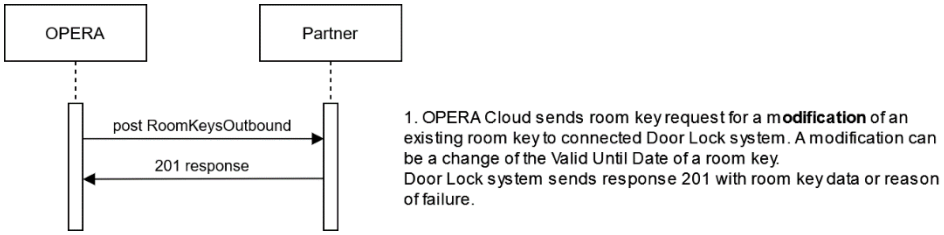
When the Door Lock system is an online system:

The room keys can be handled based on the room AND reservationId. Key data is not written on the physical key, but it is stored in the Door Lock System database.

For a change of Departure Date of a checked in reservation, OPERA Cloud sends out a request with keyType "ModifyStay" for the existing reservationId and sends the updated "validUntilDate" value.

It is expected that the Door Lock system will update the existing room key data with the updated data and leave all keys or Pincodes active.

Sequence Diagram



Description of Steps:

	Description	Sample Call

1	Request for a "ModifyStay" of existing active room key(s) or Pin code is sent to the door lock system from the OPERA Cloud PMS UI.
	The door lock system must respond within 20 seconds with a "201" status message that the room key(s) have been set to inactive, or if an unsuccessful request, a "201" response containing message information on why the request could not be processed.
	As the existing room Key information is not stored on the room key(s) itself but stored in the Door Lock Systems database, the modified data (new Valid Until Date/Time) must be updated on the database only. The new room data does not need to be encoded on the room key(s).
	The hotel guest does not need a new room key(s) or Pincode, but the guest can use the existing one for the old room.
		Online - Modify Stay Room Key
		POST {{HostName}}/fof/v1/hotels/{{HotelId}}/roomKeysOutbound
		
		{
		"keyType": "ModifyStay",
		"encoderTerminal": "TERM1",
		"encoderId": "1",
		"roomKeyType": "HardKey",
		"keyUserType": "Guest",
		"keyRequestTime": "2024-02-26T16:16:10.295Z",
		"roomId": "101",
		"oldRoomId": "",
		"noOfKeys": 1,
		"reservationIdList": [
		{
		"id": "{{ReservationId}}",
		"type": "Reservation"
		},
		{
		"id": "2468024",
		"type": "Confirmation"
		}
		],
		"guestShareFlag": "false",
		"OldGuestShareFlag": "",
		"keyValidityStart": "2024-02-26T11:16:16.000Z",
		"keyValidityEnd": "2024-02-28T12:00:00.000Z",
		"keyTrack": {
		"keyTrack1": "",
		"keyTrack2": "",
		"keyTrack3": ""
		},
		"additionalRooms": [
		"102",
		"103"
		],
		"oldAdditionalRooms": [
		"",
		""
		],
		"userId": "PMSUser1",
		"keyOptions": [
		{
		"keyOptionsCode": "SAU",
		"keyOptionDescription": "Sauna"
		},
		{
		"keyOptionsCode": "GAR",
		"keyOptionDescription": "Garage"
		}
		],
		"cardSerialNumber": "",

	Description	Sample Call

		<pre> "printData": {
		    "printText": "",
		    "printFile": "",
		    "printPicture": ""
		  }
		}</pre>
		Online - Modify Stay Room Key Response
		
		<pre>{
		  "roomId": "101",
		  "responseCode": "OK",
		  "encoderTerminal": "TERM1",
		  "roomKeyType": "HardKey",
		  "keyUserType": "Guest",
		  "keyValidityStart": "2024-02-26T11:16:16.000Z",
		  "keyValidityEnd": "2024-02-27T12:00:00.000Z",
		  "keyCreationTime": "2024-02-26T11:16:16.138Z",
		  "noOfKeys": "1",
		  "encoderId": "1",
		  "reservationIdList": [
		    {
		      "id": "{{ReservationId}}",
		      "type": "Reservation"
		    },
		    {
		      "id": "2468024",
		      "type": "Confirmation"
		    }
		  ],
		  "keyTrack": [
		    {
		      "keyTrack2": "98765A43B21C",
		    }
		  ],
		  "additionalRooms": [
		    "102",
		    "103"
		  ],
		  "responseText": "Key successfully changed",
		  "keyOptions": [
		    {
		      "keyOptionsCode": "SAU",
		      "keyOptionDescription": "Sauna",
		      "isEnabled": "true",
		      "default": "false"
		    },
		    {
		      "keyOptionsCode": "GAR",
		      "keyOptionDescription": "Garage",</pre>

	Description	Sample Call

		<pre>"isEnabled": "true",
		"default": "false"
		},
		{
		"keyOptionsCode": "MAIN",
		"keyOptionDescription": "MainEntrance",
		"isEnabled": "true",
		"default": "true"
		}
		],
		"userId": "PMSUser1"
		}</pre>

7

Anti-patterns

Review the Best Practices chapter in the OHIP user guide to learn more about functional and technical anti-patterns.

8

Key Terminology

You should review the key terminology for Room Keys and OPERA Cloud before using the APIs.

OPERA Cloud Room Keys Action Type	Description
New Key Request	A request to create a new key for a reservation. It is expected that all previous active keys for the reservation or room will become invalid.
Duplicate Key Request	A request to create additional keys for a reservation which already has active keys. It is expected that existing active keys for this reservation or room remain active.
Delete Key Request	A request to delete active key(s) for a reservation or room. It is expected that with this request all active keys for the reservation or room will become inactive or invalid, so guests will not be able to open the room after the defined validity End time.
Read Key Request	A request to the Key Card system to read Key Card information from the requested Key Encoder/reader. The Key Read functionality is of pure informational purpose. None of the received information is stored or processed in the PMS, and it is only to display key information to the user. **Note** It requires related PMS configuration to activate this action type. For more information, refer to Configuring Property Interface Rights in the OPERA Cloud User Guide.
One-Shot Key Request	A request to create or activate a room key for a specific room. It is expected that the room key is only valid for a room for specific time (for maintenance or Show room). **Note** A One-Shot key is never related to a reservation. It requires related PMS configuration to activate this action type.
Modify Stay Request	For Online Door Lock Systems, a request to adjust Room key data when guest changes length of stay (validityEnd). It requires related PMS configuration to activate this action type.

OPERA Cloud Room Keys Action Type	Description
Room Move Request	**Note** For Online Door Lock Systems, a request to adjust Room key data when guest performs room move. It requires related PMS configuration to activate this action type.
Re Encode Request	A request to re-encode a room key. Usually only utilized when PMS has the "Multiple Guest per room" functionality activated, which allows you to choose a specific room key for re-encoding. It is expected that existing active keys for this reservation or room remain active. **Note** It requires related PMS configuration to activate this action type.
Lost Key Request	A request to create a new room key for a reservation or room. Usually only utilized when PMS has the "Multiple Guest per room" functionality activated, which allows you to choose a specific room key for re-encoding. It is expected that all previous active keys for the room will become invalid. **Note** It requires related PMS configuration to activate this action type.
responseCode	Each response message for a corresponding request message must have a responseCode value. This code indicates if the request has been successfully processed or not. Specific codes will indicate a reason why a request failed. Following "responseCode": values are supported: "OK" = Successful "BY" = Encoder Busy "UR" = Unprocessable Request "RY" = Retry
"Offline" and "Online" Door Lock System	
What is the Offline Door Lock System?	OPERA Cloud PMS considers an "Offline Door Lock System" a system where the information necessary to allow access to a door is stored on the room Key (or similar media) that the related door Lock reads.
What is the Online Door Lock System?	OPERA Cloud PMS considers an "Online Door Lock System" a system where the information necessary to allow access to a door is stored in the System's database, despite being stored on the room key (or similar media). Such systems typically have the Door Lock being wired with the Door Lock System's database.

OPERA Cloud Room Keys Action Type	Description
Room Move handling with offline Door Lock System	OPERA Cloud PMS handles Room move requests in a different way depending on whether it is connected to an Offline or Online Door Lock System. When connected to an "Offline Door Lock System," OPERA Cloud PMS sends a "Remove" Key request for the old room number to inactivate current active keys and will then send a Request for "New" (and "Duplicate") keys for the new room number. It is then required that the Hotel guest must visit the Front Desk to get the new room key(s).
Room Move handling with Online Door Lock System	When connected to an "Online Door Lock System," OPERA Cloud PMS sends requests with a different message type in case relevant data that a reservation will change. For example upon "Room move" of a reservation (guest moves to another room), OPERA Cloud will send a request with keyType "RoomMove" containing information of the old room and the new room number, so the Door Lock System would only need to change the data in its database. Thus, there is no need for the guest to have the room key re-encoded at the Front Desk.
Reservation data change with offline Door Lock System	When connected to an "Offline Door Lock System" and, for example, the reservation's check out date changes, OPERA Cloud PMS will send a "Remove" Key request for the reservation or room number to inactivate the current active keys. OPERA Cloud will then send a Request for "New" (and "Duplicate") keys for the reservation or room number with the updated check out date (here the keyValidityEnd). It is then required that the Hotel guest must visit the Front Desk to get the new room key(s).
Reservation data change with Online Door Lock System	When connected to an "Online Door Lock System" and, for example, the reservation's checkout date or time changes, the PMS will send a request message with keyType "ModifyStay" containing the updated checkout date (here the keyValidityEnd). The updated information is stored in the Door Lock System's database.
	Note Above room Key actions for the Create Room Keys menu need to be configured in the Interface Rights section of the Property Interface Configuration. See details in the OPERA Cloud User Guide.
Date & Time field format	The Date/Time format for the "keyValidityStart":, "keyValidityEnd":, "keyRequestTime":, "keyCreationTime": fields in PostRoomKeysOutbound request and response will conform to ISO8601 standard whereby Oracle/ OPERA expects a max 3 character of milliseconds. The expected format for above values is: "yyyy-mm-ddThh:mm:ss[.mmm]Z" (for example, 2025-01-20T22:26:00.000Z).

9 FAQ

Q. What is the maximum time to respond to a POST roomKeysOutbound request message?

- The maximum time to respond to a POST roomKeysOutbound request message is 20 seconds. This is related to the maximum allowed call timeout within the OPERA Outbound service.

Q. The PMS user wants to have 3 room keys encoded/activated for the reservation. Will there be one API call sent with numberOfKeys = "3"?

- The PMS sends out one API request call per room key to encode/activate. It awaits the corresponding response message prior to sending further request calls for the next room key to encode/activate and again waits for response from the door lock system prior to sending the request for the third room key.

Q. If the encoding/activation of a room key fails on the Door Lock System side, will the response be a common HTTP client error response (4xx)?

- No, it is expected that the Door Lock System always sends back a HTTP 201 response message. Within the body of this message, we expect to receive a responseCode value and a user friendly responseText describing the reason of failure.

Q. Is it possible to send multiple roomKeysOutbound requests at the same time with different key EncoderIds to the Door Lock system through the Outbound system connection?

- Yes, it is expected that multiple roomKeysOutbound requests for different key EncoderIds at same time can be handled by the Door Lock System.

Q. Does the Door Lock API include sending Guest or Reservation details, such as Guest Names?

- The Door Lock API roomKeysOutbound only includes the necessary information needed to create a room key. It contains the reservationId but no further guest or reservation information/detail.

  To retrieve such additional information beside the room key information exchanged with this API, your system can subscribe to OHIP and consume related guest profile or reservation APIs.

Q. How can the Door Lock system be notified of a check-in or check-out of a reservation?

- The Door Lock API roomKeysOutbound only includes the necessary information needed to create a room key. It contains the key validity information but it does not notify the external system that a guest / reservation has been checked-in or checked-out.
  To retrieve such additional information beside the room key information exchanged with this API, your system can subscribe to OHIP and consume related check-in/out notification by using the OPERA Cloud Business Events, either pulling these events or receiving them by streaming Business Events (push option).

Q. What are the keyTrack sub tags used for?

- **keyTrack1**: The keyTrack1 tag can be used for a specific data string to be sent from OPERA Cloud PMS to the connected Door Lock System in case OPERA Cloud PMS does support the requested data. Could be room or guest specific information which the Door Lock System requests.

Note that the PMS does not store incoming keyTrack1 data, it might pass the data through to external systems.

Outgoing keyTrack1 data requires configuration in OPERA Cloud PMS interface set up.

- **keyTrack2**: The keyTrack / keyTrack2 tag contains a unique Id which can either be created by OPERA Cloud PMS and is sent with the roomKeysOutbound request to let the door lock system store it on the key device.

  OR

  It will be sent by the door lock system in the related response message.

  It was common that the keyTrack2 was sent by OPERA Cloud PMS in the request, so it was written onto the Mag stripe of the door key card. Today it is more common that the Door Lock System sends the UDID of a key in the response message.

  In both cases, the keyTrack2 value will be stored in PMS and is linked to the reservation so that a guest can be identified by reading the door key “track2” value on another system and send an inquiry to the PMS. Usually used by Restaurant POS Systems.

- **keyTrack3**: The keyTrack3 tag contains the unique Pin code provided by the Door Lock system which does use number pads on the guest room lock to enter a Pin to open the door. The Pin code will be stored in OPERA PMS as to enable the user to display and print the pin code to hand it out to the guest.

  To enable Pin code handling in OPERA Cloud, the related configuration parameter must be set in the OPERA Cloud Interface configuration.