Cloud Infra-Tenants External API for Harmony Mobile Service

Goal

The following guide is targeted to assist developers who need to preform task in Check Point's Harmony Mobile Service within the Infinity Portal programmatically, using REST API calls.

Please note that Cloud-Infra is the cloud-based infrastructure (AKA Infinity Portal) that is hosting all the different security services by Check Point and Harmony Mobile (HM) is one of these Services. For API calls related to access and tenants creation, please refer to the Cloud-Infra API guide.

Prerequisite

1. POSTMAN / CURL commands / any other method used to run API calls – in the example below we will use POSTMAN tool

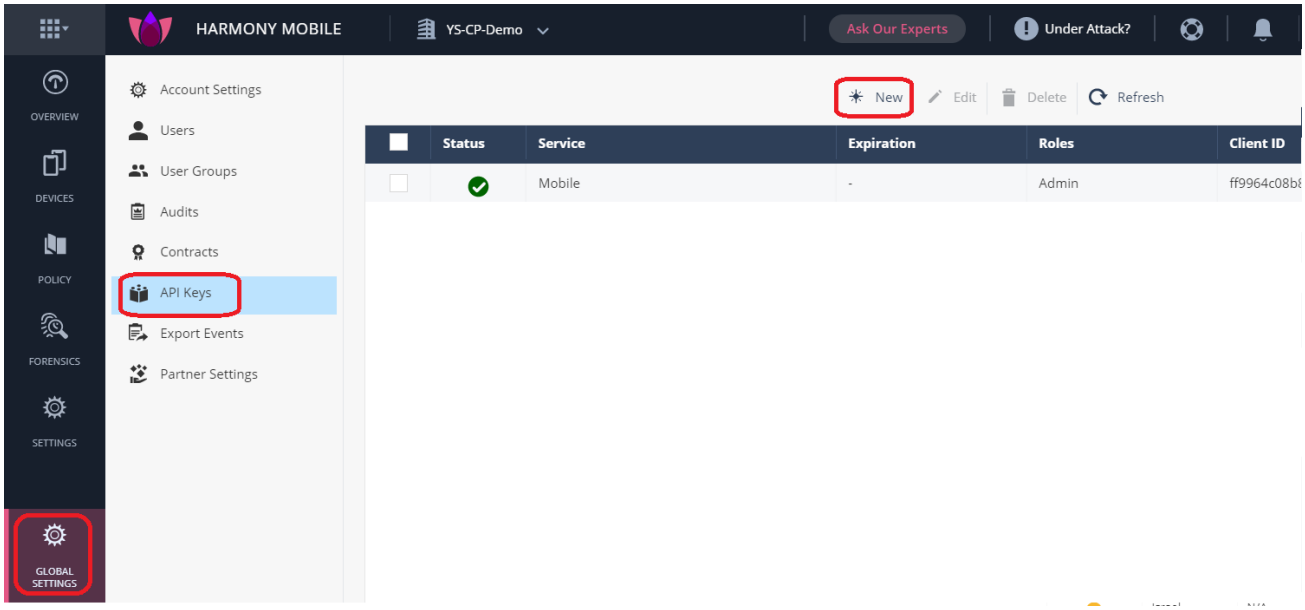
   POSTMAN
2. Admin level Access to Check Point's infinity Portal https://portal.checkpoint.com/signin

Process

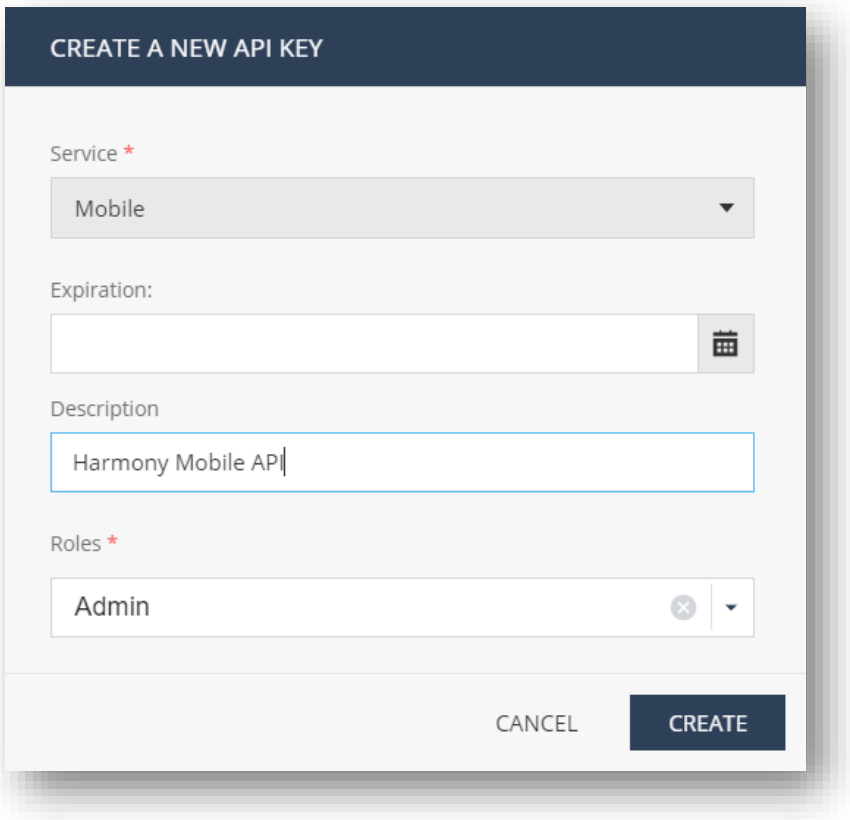
1. Get your Client ID and Secret Key

The following can also be accomplished using Cloud-Infra API

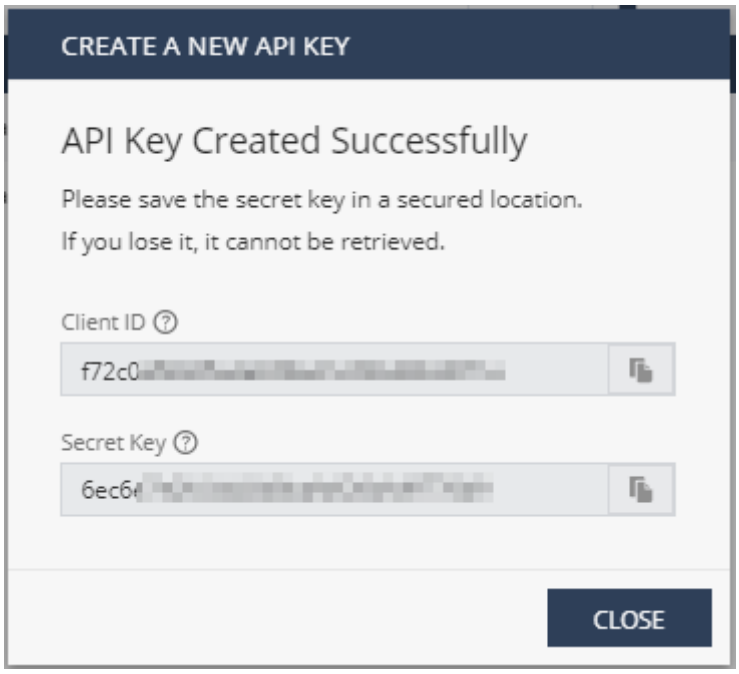
a. Connect to your Infinity Portal dashboard
b. Go to ***Global Settings-->API Keys***, and click "New":



c. Select Service “Mobile”, give Description, Roles “Admin” and click “Create”:



d. You will receive a Client ID and a Secret Key.
Save them on your computer, as there will be no access to this information after you click on “Close”

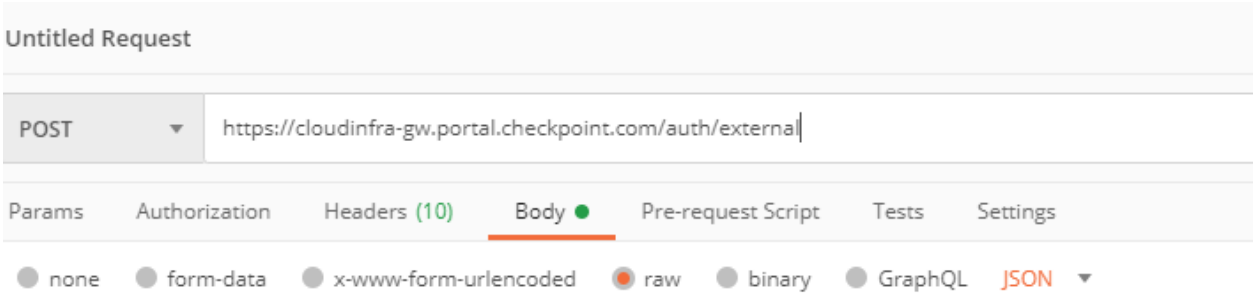


2. Get your Token

In Postman, create a POST request:

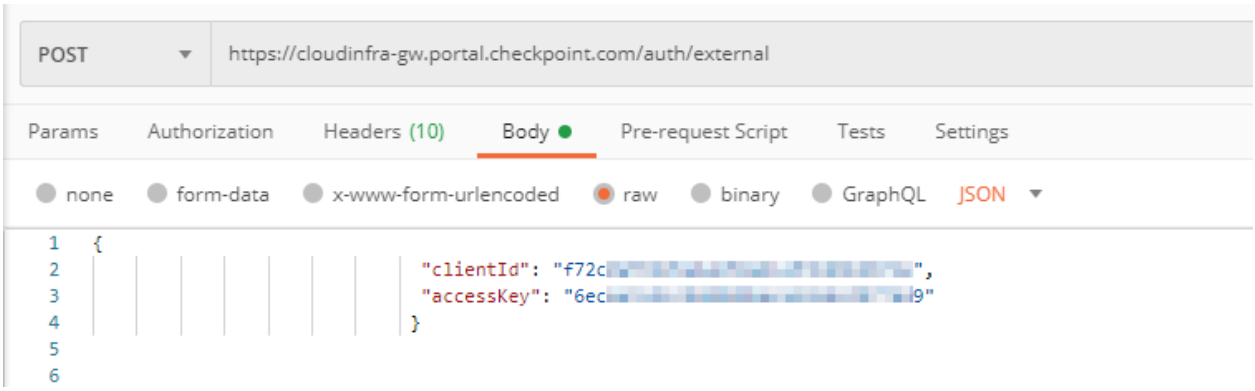
a. The request should be directed to: https://cloudinfra-gw.portal.checkpoint.com/auth/external

b. Click on “Body” and select format “Raw” and “JSON”:



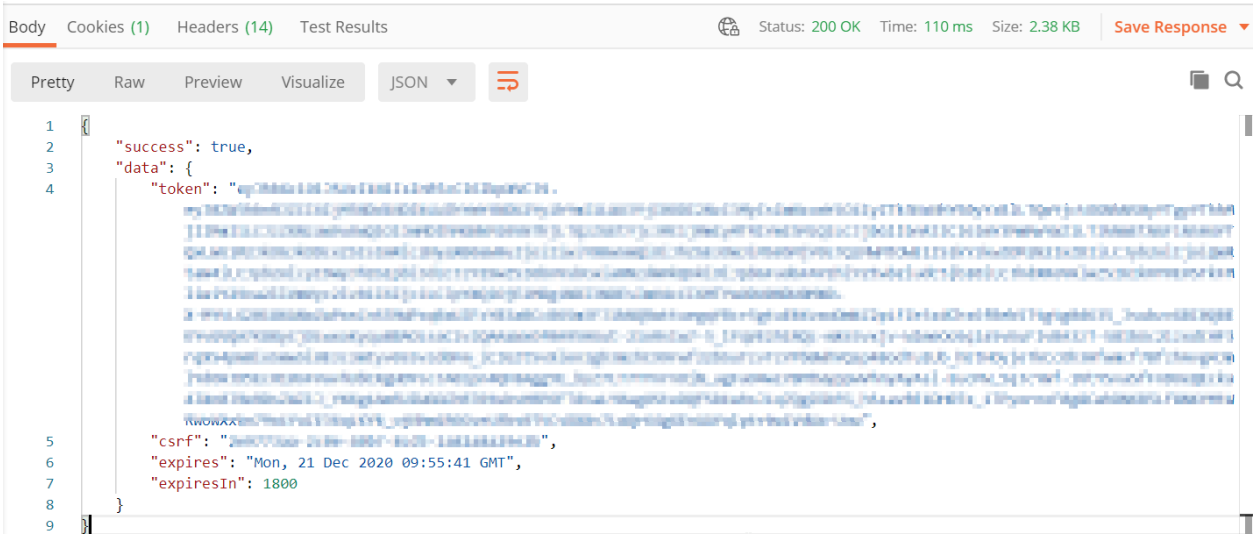
c. In the Body enter the following:

```
{"clientId": <”Your Client ID”>,
 "accessKey": <”Your Access Key”>
}
```



Click on “Send”

d. The response for this request is your Token:



3. Add a Group to the dashboard

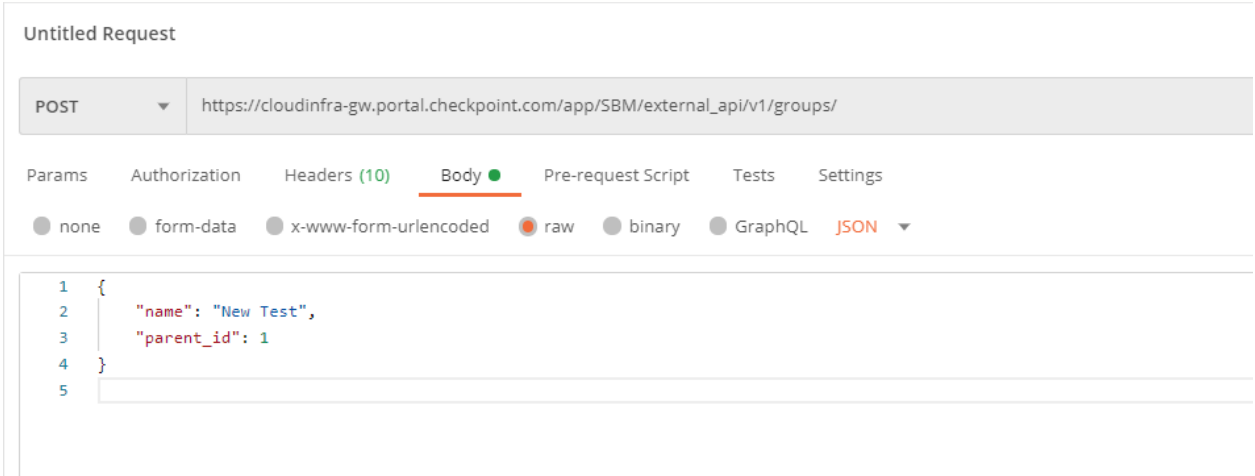
a. Create a POST request
The request should be:
https://cloudinfra-gw.portal.checkpoint.com/app/SBM/external_api/v1/groups/

b. In Headers, add a key “Authorization” with a value of “<Bearer +*Your token*>”

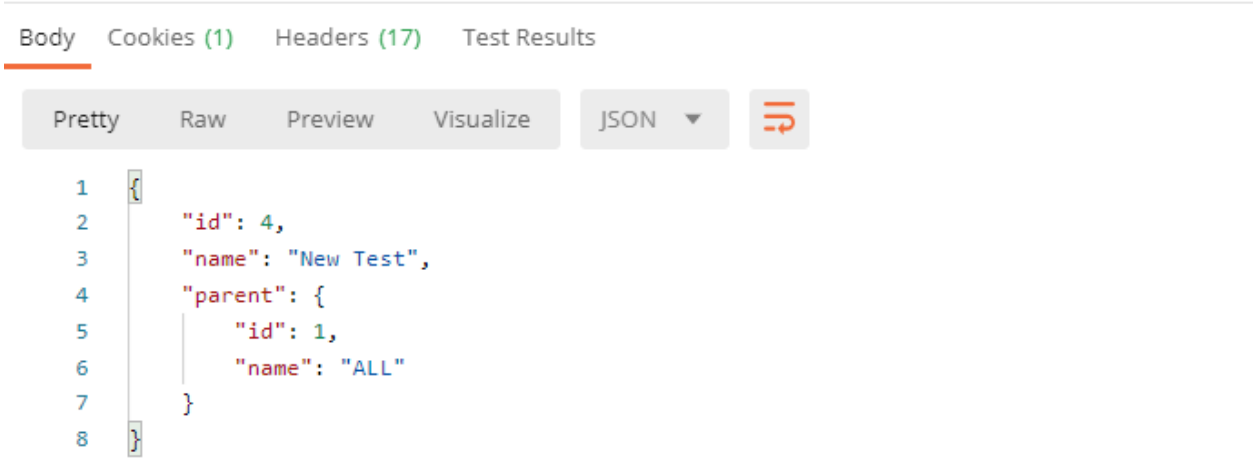
c. Click on Body and select format “Raw” and “JSON”.
In the Body add information about the group you want to create:

```
{"name": "<Your new group's name>",
"parent_id": "<Parent ID group (optional - default to the 'ALL' group)>"
}
```

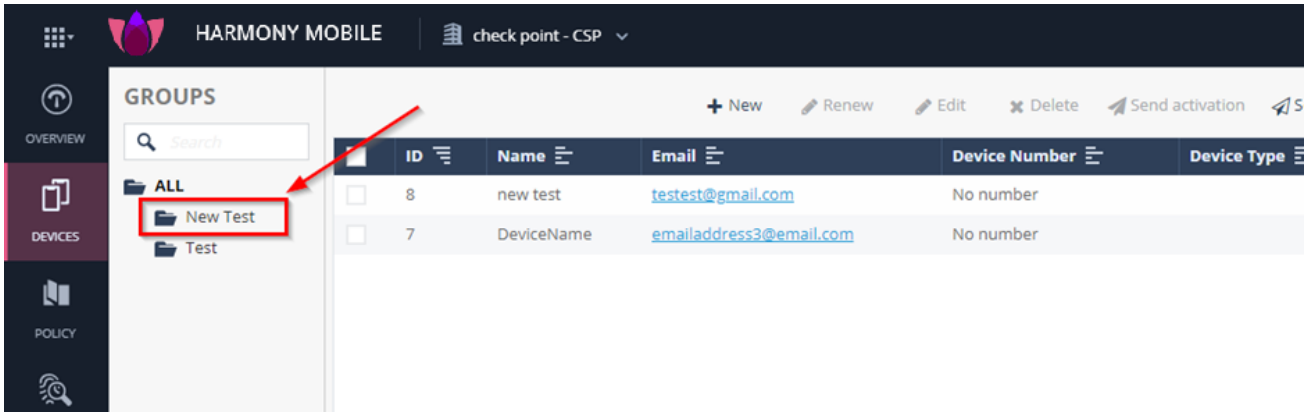
Example: *Here the parent group will be ALL with ID=1*



d. The response will contain the new group's ID:



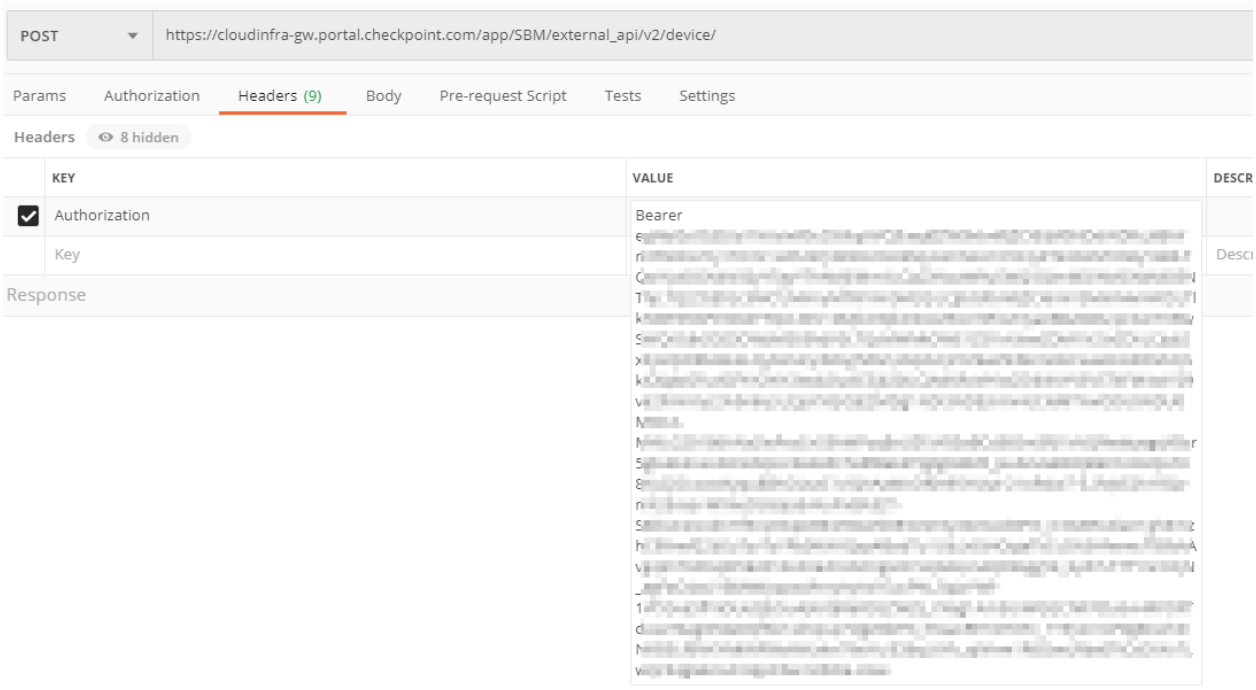
e. In your dashboard, you can see the new group "*New Test*" created under the group *ALL*:



4. Add a device to the group

a. Create a POST request
The request should be:
https://cloudinfra-gw.portal.checkpoint.com/app/SBM/external_api/v2/device/

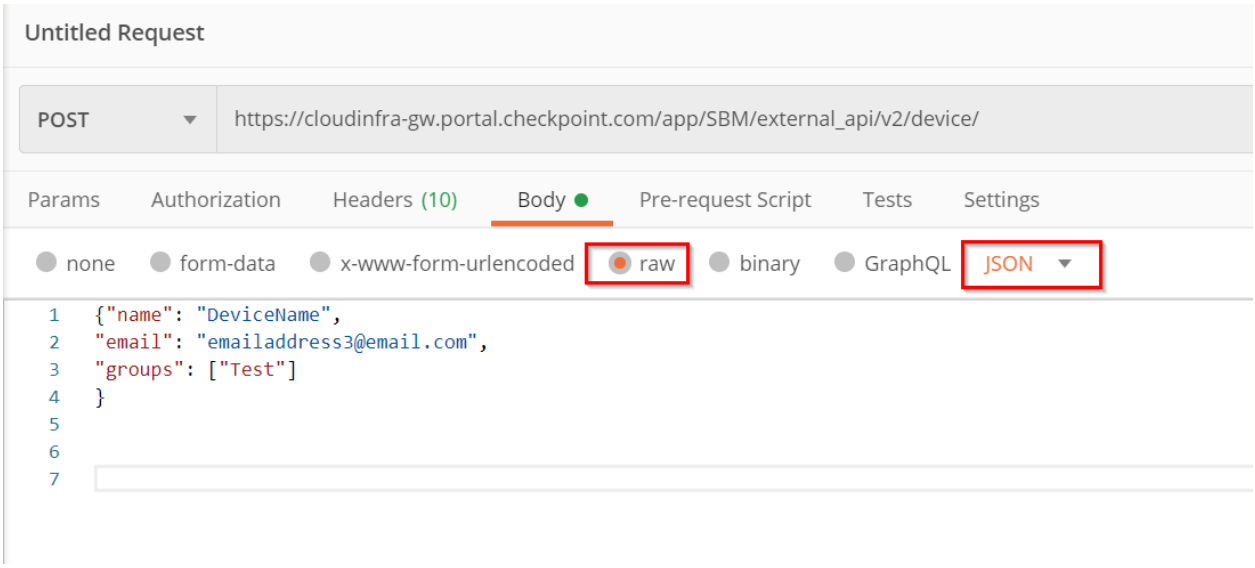
b. In Headers, add a key "Authorization" with a value of "<Bearer +*Your token*>"



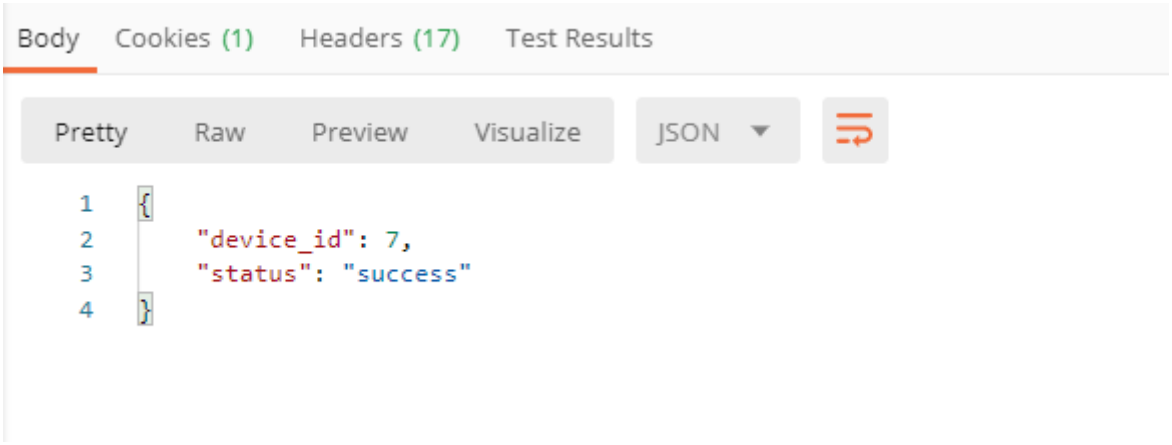
c. Click on “Body” and select format “Raw” and “JSON”.
In the Body add information about the device you want to create:

```
{"name": "<Device/User name>",
"email": "<User email address>",
"groups": ["<Specify the group name in case you have a dedicated group for this new deice in your dashboard>"]
}
```

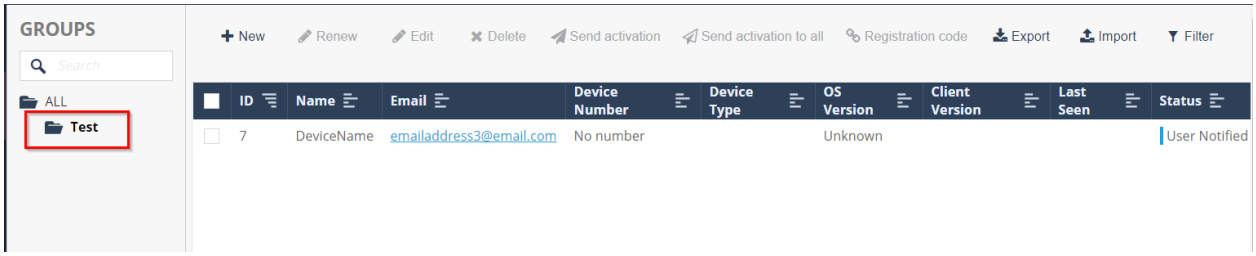
Example:



d. The response will contain the device ID:



e. In your dashboard, you can see the newly created device with a status of *User Notified*:



Additional API for the HM Service are documented in the appendix below.

Appendix – All Harmony Mobile Service APIs

Below is the updated API documentation of Harmony Mobile Service from Infinity Portal with all possible External API calls (REST API)

0. URI: https://cloudinfra-gw.portal.checkpoint.com/auth/external	**GET CI TOKEN**
HTTP METHOD	POST
- H: Content-Type	application/json
REQUEST BODY	

{
“clientId”: ${client_id},
“accessKey”: ${secret}
}
	In order to generate client ID and secret, log into CI at https://portal.checkpoint.com, then go to Global Settings > API Keys > New.

RESPONSE	

{
“success”: true,
“data”: {
“token”: ${CI_TOKEN},
“csrf”: ${csrf},
“expires”: Date,
“expiresIn”: Number
}
}
	Use ${CI_TOKEN} in External APIs.

Remember always to use In Headers, the “Authorization” key with a value of “<Bearer *+Your token*>”

1. URI: https://cloudinfra-gw.portal.checkpoint.com/app/SBM/external_api/v1/device_status/${device_id}/?format=json	**Get device details by Check Point's device ID.**
HTTP Method:	GET
Format:	Json
${device_id}	Check Point's internal identification of the device.
Response	
client_version	
device_type	
email	User email
internal_id	Internal Check Point's ID

groups: {
id,
name,
parent: {
id,
name
	User’s groups {hierarchical structure}

} }	
last_connection	Last time the device contacted the system.
mail_sent	True if registration mail sent, else, False
mdm: {uuid }	In case the device was provisioned by a MDM then it will contain the unique identifier as reported by the MDM.
model	Device model
name	User name
number	Phone Number
os_type	Operation system name. Can be either “Android” or “IOS”
os_version	Version of the operation system
risk	The current risk of the device.
status	The status of the device, is it connected to the system, User Notified registration, etc.

2. URI: https://cloudinfra-gw.portal.checkpoint.com/app/SBM/external_api/v1/device_status/?mdm__uuid=${mdmuuid}&format=json&groups__in=<>	**Get device details by its associated groups.**
HTTP Method:	GET
Format:	Json
use filter groups__in or groups__exact like so: groups__in=1,2,3 - devices in group id's 1 **or** 2 **or** 3 groups__exact=1,2,3 - devices in groups id's 1 **and** 2 **and** 3	Get all device details. You can filter by groups. The list supplied in groups__in and groups__exact should be comma ‘,’ separated
Response	
"meta": {"limit": 20,"next": null,"offset": 0,"previous": null,"total_count": 1}	Each response will contain a header indicating how many pages the request contains and how many results are found in the current response.
"objects":	The results are part of an array.
client_version	
device_type	
email	User email
internal_id	Internal Check Point's ID

groups: {
id,
name,
parent: {
id,
name
}
}
	User’s groups {hierarchical structure}

last_connection	Last time the device contacted the system.
mail_sent	True if registration mail sent, else, False

mdm: {uuid }	In case the device was provisioned by a MDM then it will contain the unique identifier as reported by the MDM.
model	Device model
name	User name
number	Phone Number
os_type	Operation system name. Can be either “Android” or “IOS”
os_version	Version of the operation system
risk	The current risk of the device.
status	The status of the device, is it connected to the system, registration, etc.

3. URI: https://cloudinfra-gw.portal.checkpoint.com/app/SBM/external_api/v1/device_status/?mdm__uuid=${mdmuuid}&format=json	**Get device details by the mdm uuid.**
HTTP Method:	GET
Format:	Json
${mdmuuid}	The mdm unique identifier. The format changes depending on the MDM. If the device wasn’t enrolled via MDM then you’ll get back an empty result
Response	
"meta": {"limit": 20,"next": null,"offset": 0,"previous": null,"total_count": 1}	Each response will contain a header indicating how many pages the request contains and how many results are found in the current response.
"objects":	The results are part of an array.
client_version	
device_type	
email	User email
internal_id	Internal Check Point's ID

groups: {
id,
name,
parent: {
id,
name
}
}
	User’s groups {hierarchical structure}

last_connection	Last time the device contacted the system.
mail_sent	True if registration mail sent, else, False
mdm: {uuid }	In case the device was provisioned by a MDM then it will contain the unique identifier as reported by the MDM.
model	Device model
name	User name
number	Phone Number
os_type	Operation system name. Can be either “Android” or “IOS”

os_version	Version of the operation system
risk	The current risk of the device.
status	The status of the device, is it connected to the system, registration, etc.

4. URI: https://cloudinfra-gw.portal.checkpoint.com/app/SBM/external_api/v1/device_status/?format=json&risk__in=<>&status__in=<>	**Get all device details. You can filter by status and risk level. The list supplied in risk and status should be ‘,’ separated.**
HTTP Method:	GET
Format:	Json
Response	
{"meta": {"limit": 20,"next": null,"offset": 0,"previous": null,"total_count": 1}	Each response will contain a header indicating how many pages the request contains and how many results are found in the current response.
"objects":	The results are part of an array.
status	The status of the device, is it connected to the system, registration, etc.
risk	The current risk of the device.
last_connection	Last time the device contacted the system.
os_type	Operation system name. Can be either “Android” or “IOS”
os_version	Version of the operation system
model	Device model
internal_id	Internal Check Point's ID
mdm: {uuid }	In case the device was provisioned by a MDM then it will contain the unique identifier as reported by the MDM.

5. URI: https://cloudinfra-gw.portal.checkpoint.com/app/SBM/external_api/v1/device_status?status=${status}&time_gte=${time_from}&time_lte=${time_to}&format=json	**Get all devices that had a specific status during time interval**
HTTP Method:	GET
Format:	Json
${status}	A status that a device has. The value is taken from a predefined list of statuses see Appendix.
${time_from}	Epoch time in milliseconds. Inclusive.
${time_to}	Epoch time in milliseconds. Exclusive.
Response	
{"meta": {"limit": 20,"next": null,"offset": 0,"previous": null,"total_count": 1}	Each response will contain a header indicating how many pages the request contains and how many results are found in the current response.
"objects":	The results are part of an array.
status	The status of the device, is it connected to the system, registration, etc.
risk	The current risk of the device.
last_connection	Last time the device contacted the system.

os_type	Operation system name. Can be either “Android” or “IOS”
os_version	Version of the operation system
model	Device model
internal_id	Internal Check Point's ID
mdm: {uuid }	In case the device was provisioned by a MDM then it will contain the unique identifier as reported by the MDM.

6. URI: https://cloudinfra-gw.portal.checkpoint.com/app/SBM/external_api/v1/device_status/${device_id}/resend	**Resend email notification to a specific device id. An email will be sent only to devices that are in User Notified status. see Device Status**
HTTP Method:	POST
${device_id}	Check Point's internal identification of the device.
Request body:	

{
"send_reg_sms": True/False,
"send_reg_email": True/False
}
	Optional

Response	
HTTP 200	Email was sent successfully
HTTP 500	Error while trying to send an email

7. URI: https://cloudinfra-gw.portal.checkpoint.com/app/SBM /external_api/v1/device_status/${device_id}/mark_email_sent/	**Change the device status to "User Notified".**
HTTP Method	Post
${device_id}	Check Point's internal identification of the device.
Response	
HTTP 200	Marked email sent
HTTP 404	Device not found

8. URI: https://cloudinfra-gw.portal.checkpoint.com/app/SBM /external_api/v1/device_status/${device_id}/get_registration_details	**Enable getting device registration code for existing device by device id**
HTTP Method:	GET
${device_id}	Check Point's internal identification of the device
Response	
device_hash	Device registration code
id	Check Point's internal identification of the device

server	Check Point's server

9. URI: https://cloudinfra-gw.portal.checkpoint.com/app/SBM/external_api/v1/device_status/get_registration_details?mi_id=${MI_id}&device_id=${device_id}&email=${email}	**Enable getting device registration code for existing device (by device id, email and/or MI id)**
HTTP Method:	GET
${MI_id}	MDM Device UUID
${device_id}	Check Point's internal identification of the device
${email}	Device owner email
Response	
device_hash	Device registration code
id	Check Point's internal identification of the device
server	Check Point's server

10. URI: https://cloudinfra-gw.portal.checkpoint.com/app/SBM/external_api/v1/device/?user_email=${email}	**Delete device**
HTTP Method:	DELETE
${email}	Device owner email

11. https://cloudinfra-gw.portal.checkpoint.com/app/SBM/external_api/v1/device/${device_id}/	**Delete device**
HTTP Method:	DELETE
${device_id}	Internal device id

12. URI: https://cloudinfra-gw.portal.checkpoint.com/app/SBM/external_api/v2/device/	**Add device**
HTTP Method:	POST
POST body should contains the following json:	
{"name": ${user_name}	Required
"email": ${user_email}	Required
"number": ${user_phone_number}	Optional.
"send_reg_email":false\true	Optional.
"send_reg_sms":false\true	Optional.
"groups":[\"ALL\",\" \"]}	Optional.
Response	
device_id	Check Point's internal identification of the device
status	The status of the request (success/failure)

13. URI: https://cloudinfra-gw.portal.checkpoint.com/app/SBM/external_api/v3/alert/?id__gt=<alert_id>	**Get dashboards alerts**

HTTP Method:	GET
<alert_id>	Optional.
Response	
id	Check Point's internal identification of the alert
event	
backend_last_updated	
details	
severity	
event_timestamp	The time stamp for the alert update in the device
threat_factors	
attack_vector	
mdm_uuid	The device MDM unique identifier
device_id	
email	
number	The device phone number
name	
device_rooted	
client_version	
device_model	
os_version	
For Suspicious Package (Application):	
app_version	
app_repackage	
App_sha256	
For Network Attack:	
network_certificate	
location	
id__gt (optional field)	Return all the alerts with ID greater than the one provided

14. URI: https://cloudinfra-gw.portal.checkpoint.com/app/SBM/external_api/v1/groups/	**GET ALL GROUPS**
${host-name}	https://cloudinfra-gw.portal.checkpoint.com/app/SBM
HTTP METHOD	GET
- H: Authorization	Bearer ${CI_TOKEN} ${CI_TOKEN} - From step 0
RESPONSE	

}
ID,
Name,
Parent: {
ID,
Name
}
	Group id
	Group name
	Parent ID and name

15. URI: https://cloudinfra-gw.portal.checkpoint.com/app/SBM/external_api/v1/groups/?name=${group-name}	**GET GROUP BY NAME**
${host-name}	https://cloudinfra-gw.portal.checkpoint.com/app/SBM
HTTP METHOD	GET
- H: Authorization	Bearer ${CI_TOKEN} ${CI_TOKEN} - From step 0
${group-name}	Existing group name
RESPONSE	

}
ID,
Name,
Parent: {
ID,
Name
}
	Group id
	Group name
	Parent ID and name

16. URI: https://cloudinfra-gw.portal.checkpoint.com/app/SBM/external_api/v1/groups/	**ADD NEW GROUP**
${host-name}	https://cloudinfra-gw.portal.checkpoint.com/app/SBM
HTTP METHOD	POST
- H: Authorization	Bearer ${CI_TOKEN} ${CI_TOKEN} - From step 0
REQUEST BODY	

{
“name”: ${name},
“parent_id”: ${parent_id}
}
	${name} - new group name
	${parent_id} - parent group id (optional - default to the ‘ALL’ group)

RESPONSE	

}
ID,
Name,
Parent: {
ID,
Name
}
	Group id
	Group name
	Parent ID and name

17. URI: https://cloudinfra-gw.portal.checkpoint.com/app/SBM/external_api/v1/groups/${group-id}/	**EDIT GROUP**
${host-name}	https://cloudinfra-gw.portal.checkpoint.com/app/SBM
HTTP METHOD	PATCH
- H: Authorization	Bearer ${CI_TOKEN} ${CI_TOKEN} - From step 0
${group-id}	Group id
REQUEST BODY	

{
“name”: ${name},
“parent_id”: ${parent_id}
}
	${name} - new group name (optional)
	${parent_id} - parent group id (optional)

RESPONSE	

}
ID,
Name,
Parent: {
ID,
Name
}
	Group id
	Group name
	Parent ID and name

18. URI: https://cloudinfra-gw.portal.checkpoint.com/app/SBM/external_api/v1/groups/${group-id}/	**DELETE GROUP**
${host-name}	https://cloudinfra-gw.portal.checkpoint.com/app/SBM
HTTP METHOD	DELETE
- H: Authorization	Bearer ${CI_TOKEN} ${CI_TOKEN} - From step 0

19. URI: https://cloudinfra-gw.portal.checkpoint.com/app/SBM/external_api/v3/audit/?id__gt=<audit_id> **https://cloudinfra-gw.portal.checkpoint.com/app/SBM/external_api/v3/audit/?time__gt=2021-07-01%2008:08:00&module=Policy**	**Get dashboards audits trail (e.g. by ID, by time & module, etc…)**
${host-name}	https://cloudinfra-gw.portal.checkpoint.com/app/SBM
HTTP METHOD	GET
<audit_id>	Optional.
Response	
id	Check Point's internal identification of the audit
category	
event_details	event column in the audit table
event_details_params	event details column in the audit table
module	
severity	
admin_user	
time	
id__gt (optional field)	Return all the audits with ID greater than the one provided

Table of Values

Device Status/State	Value	Description
Processing	-1	Device was added to the dashboard, but not to the GW
Provisioned/User Notified	0	Device was added to the system, however it did not finish the registration flow
Active	1	Device registered successfully and is under Check Point's protection
In Active	4	User removed Check Point's app from the device

Device Risk	Value	**Description**
None	0	Device has no risk associated with it
Low	1	Device has low risk associated with it
Medium	2	Device has medium risk associated with it
High	3	Device has high risk associated with it

