# UniFi Access API Documentation

Leverage the power of the UniFi Access API

# 1. Introduction

Welcome to the UniFi Access API guide, which provides a detailed step-by-step explanation of how to utilize the API effectively. The UniFi Access API is an open interface protocol that allows developers to interact with a specific service using predefined interfaces.

## 1.1 Create API Token & Download API Documentation

Before you can use the UniFi Access API, you need to obtain a valid API Token for authentication and access control. Follow the steps below to create an API Token:

1. Sign in to your UniFi Portal (https://account.ui.com/login) using your web browser.
2. Select the UniFi Console where the UniFi Access application is installed.
3. Go to **Access > Settings > General > Advanced**.
4. Go to **API Token** to **Download API documentation** and create an API Token.
5. To create an API Token, click **Create New**, enter the key name and validity period, select the permission scopes, and click **Create**.
6. Once the API Token is created, ensure to **Copy API Token** and store it safely for future use. Please note that the key is displayed only once for security purposes.

## 1.2 Obtain Your Hostname

The UniFi Access application can be hosted on a server within the local network and accessed via the LAN port. The hostname can either be the server's IP address or a custom domain name mapped to that IP. Connect using the specified port number (**12445**) and establish an HTTPS protocol for a secure connection.

## 1.3 Use the API

1. Construct the API request as per the instructions in the API documentation.
2. Send the constructed request to the server using the previously obtained API Token and hostname.
3. Parse the server's response to retrieve the required data or perform desired operations.

## 1.4 Security Configuration

1. Set up appropriate security measures to protect the API and the server.
2. Configure firewall rules to allow incoming traffic on the specific ports used by the API.
3. Enable HTTPS encryption using SSL/TLS certificates to secure data transmission.
4. Implement rate limiting and throttling mechanisms to prevent abuse and ensure fair usage.

# 1.5 Important Notes

1. Ensure your UniFi Access version is **1.9.1** or later.
2. **The API is not available after upgrading to Identity Enterprise.**
3. Obtain valid access permissions and adhere to the usage guidelines provided by the service provider before using the UniFi Access API.
4. Safeguard your API Token and do not share it with others to avoid security risks.
5. If you have any questions or concerns, please contact the technical support team of your service provider.

The steps above cover the basic process and essential considerations for using the UniFi Access API. Follow the instructions and refer to the API documentation for smooth development. Enjoy the convenience of using the UniFi Access API!

# 2. Overview

This section provides an introduction to essential concepts related to working with APIs, including API tokens, HTTP status codes, success and error codes, HTTP verbs, and API hosts.

## 2.1 API Token

To authenticate API requests with UniFi Access, you need to utilize API tokens associated with your account. If a request includes a deleted or expired token, the service will return an authentication error. The **Authorization Token** is obtained from UniFi Portal.

## 2.2 HTTP Status Code Reference

| **200** | **OK** | **Everything worked as expected.** |
|---|---|---|
| 400 | Bad Request | The request is unacceptable, often due to missing required parameters. |
| 401 | Unauthorized | The request lacks a valid API token for authentication. |
| 402 | Request Failed | The request contains valid parameters, but failed for some reason. |
| 403 | Forbidden | The API token used does not have the necessary permissions to perform the request. |
| 429 | Too Many Requests | Too many requests were sent to the API in a short amount of time. |
| 500, 502, 503, 504 | Server Errors | Something went wrong on UniFi Access's end during request processing. |

## 2.3 Success Code

{"code":"SUCCESS","msg":"success"}

| **Code** | **Message** |
|---|---|
| SUCCESS | Success |

## 2.4 Error Code

{"code":"CODE_PARAMS_INVALID","msg":"Invalid parameters."}

| Code | Message |
| --- | --- |
| CODE_PARAMS_INVALID | The provided parameters are invalid. |
| CODE_SYSTEM_ERROR | An error occurred on the server's end. |
| CODE_RESOURCE_NOT_FOUND | The requested resource was not found. |
| CODE_OPERATION_FORBIDDEN | The requested operation is not allowed. |
| CODE_AUTH_FAILED | Authentication failed. |
| CODE_ACCESS_TOKEN_INVALID | The provided access token is invalid. |
| CODE_UNAUTHORIZED | You not are allowed to perform this action. |
| CODE_NOT_EXISTS | The requested item does not exist. |
| CODE_USER_EMAIL_ERROR | The provided email format is invalid. |
| CODE_USER_ACCOUNT_NOT_EXIST | The requested user account does not exist. |
| CODE_USER_WORKER_NOT_EXISTS | The requested user does not exist. |
| CODE_USER_NAME_DUPLICATED | The provided name already exists. |
| CODE_USER_CSV_IMPORT_INCOMPLETE_PROP | Please provide both first name and last name. |
| CODE_ACCESS_POLICY_USER_TIMEZONE_NOT_FOUND | The requested workday schedule could not be found. |
| CODE_ACCESS_POLICY_HOLIDAY_TIMEZONE_NOT_FOUND | The requested holiday schedule could not be found. |
| CODE_ACCESS_POLICY_HOLIDAY_GROUP_NOT_FOUND | The requested holiday group could not be found. |
| CODE_ACCESS_POLICY_HOLIDAY_NOT_FOUND | The requested holiday could not be found. |
| CODE_ACCESS_POLICY_SCHEDULE_NOT_FOUND | The requested schedule could not be found. |
| CODE_ACCESS_POLICY_HOLIDAY_NAME_EXIST | The provided holiday name already exists. |
| CODE_ACCESS_POLICY_HOLIDAY_GROUP_NAME_EXIST | The provided holiday group name already exists. |
| CODE_ACCESS_POLICY_SCHEDULE_NAME_EXIST | The provided schedule name already exists. |
| CODE_ACCESS_POLICY_SCHEDULE_CAN_NOT_DELETE | The schedule could not be deleted. |
| CODE_ACCESS_POLICY_HOLIDAY_GROUP_CAN_NOT_DELETE | The holiday group could not be deleted. |
| CODE_CREDS_NFC_HAS_BIND_USER | The NFC card is already registered and assigned to another user. |
| CODE_CREDS_DISABLE_TRANSFER_UID_USER_NFC | The UniFi Identity Enterprise user's NFC card is not transferrable. |
| CODE_CREDS_NFC_READ_SESSION_NOT_FOUND | Failed to obtain the NFC read session. |
| CODE_CREDS_NFC_READ_POLL_TOKEN_EMPTY | The NFC token is empty. |
| CODE_CREDS_NFC_CARD_IS_PROVISION | The NFC card is already registered at another site. |
| CODE_CREDS_NFC_CARD_PROVISION_FAILED | Please hold the NFC card against the reader for more than 5 seconds. |
| CODE_CREDS_NFC_CARD_INVALID | The card type is not supported. Please use a UA Card. |
| CODE_CREDS_NFC_CARD_CANNOT_BE_DELETE | The NFC card could not be deleted. |
| CODE_CREDS_PIN_CODE_CREDS_ALREADY_EXIST | The PIN code already exists. |
| CODE_CREDS_PIN_CODE_CREDS_LENGTH_INVALID | The PIN code length does not meet the preset requirements. |
| CODE_SPACE_DEVICE_BOUND_LOCATION_NOT_FOUND | The device's location was not found. |
| CODE_DEVICE_DEVICE_VERSION_NOT_FOUND | The firmware version is up to date. |

| | |
|---|---|
| CODE_DEVICE_DEVICE_VERSION_TOO_OLD | The firmware version is too old. Please update to the latest version. |
| CODE_DEVICE_DEVICE_BUSY | The camera is currently in use. |
| CODE_DEVICE_DEVICE_NOT_FOUND | The device was not found. |
| CODE_DEVICE_DEVICE_OFFLINE | The device is currently offline. |
| CODE_OTHERS_UID_ADOPTED_NOT_SUPPORTED | The API is not available after upgrading to Identity Enterprise. |
| CODE_HOLIDAY_GROUP_CAN_NOT_DELETE | The holiday group could not be deleted. |
| CODE_HOLIDAY_GROUP_CAN_NOT_EDIT | The holiday group could not be edited. |
| CODE_DEVICE_WEBHOOK_ENDPOINT_DUPLICATED | The provided endpoint already exists. |
| CODE_DEVICE_API_NOT_SUPPORTED | The API is currently not available for this device. |

# 2.5 HTTP Verbs

| HTTP Method | Description |
|---|---|
| GET | Used for retrieving objects. |
| POST | Used for creating objects or performing custom actions. |
| PUT | Used for replacing objects or collections. |
| DELETE | Used for deleting objects. |

# 2.6 API Host

The Open API Server is hosted on port 12445 and can be accessed via HTTPS at https://console-ip:12445. The server certificate is self-generated and untrusted.

# 2.7 Request Header

The header for a request contains the following information:

| Parameter | Required | Type | Description | Example |
|---|---|---|---|---|
| Authorization | T | String | Token required for authentication and access control. | Authorization: Bearer wHFmHRuX4l7sB2oDkD6wHg |

# 2.8 Response Schema

The response for a request contains the following information:

```
{
  "code": "SUCCESS",
  "msg": "success",
  "data": {}
}
```

1. code: Represents the result of request handling and indicates success or failure.
2. msg: Represents the error description if the code is not equal to 1.
3. data: Represents the data of API requests.

# 3. User

> The APIs here are designed for managing users, including handling their basic information and assigning NFC cards, PIN codes, and access policies to them.

## 3.1 Schemas

| Parameter | Type | Description |
| --- | --- | --- |
| Id | String | Identity ID of the user. |
| first_name | String | First name of the user. |
| last_name | String | Last name of the user. |
| full_name | String | Full name of the user. |
| alias | String | Preferred name of the user. |
| user_email | String | Email of the user. **UniFi Access Requirement:** `1.22.16 or later` |
| email_status | String | The status of the user's email. |
| phone | String | Contact phone number of the user. |
| employee_number | String | Employee number of the user. |
| onboard_time | Integer | User onboarding date. |
| nfc_cards | Array[Object] | Token associated with the bound NFC card. |
| nfc_cards[].id | String | Display ID of the NFC card. |
| nfc_cards[].token | String | Unique NFC card token. |
| pin_code | Object | Token associated with the bound PIN code. |
| pin_code.token | String | The user's PIN hash code credential for unlocking a door. |
| access_policy_ids | Array[String] | Collection of the access policy ID. |
| access_policies | Array[Object] | All policies assigned to the user. |
| status | String | `enum status {ACTIVE,PENDING,DEACTIVATED}` **ACTIVE**: The user account is in active status. **PENDING**: A new admin account has been invited by the SSO account, but the invitation has not been accepted. **DEACTIVATED**: The account has been deactivated. |

## 3.2 User Registration

This API allows you to register a new user.

- Request URL: `/api/v1/developer/users`
- Permission Key: `edit:user`

- Method: `POST`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Body

| Parameter | Required | Type | Description | Example | How to Get It? |
| --- | --- | --- | --- | --- | --- |
| first_name | T | String | First name of the user. | | |
| last_name | T | String | Last name of the user. | | |
| user_email | F | String | Email of the user. **UniFi Access Requirement:** `1.22.16 or later` | | |
| employee_number | F | String | Employee number of the user. **Omit this parameter if it doesn't need to be updated.** | | |
| onboard_time | F | Integer | User onboarding date. | 1689150139 | |

## Response Body

> Schemas: Schemas

## Response Sample

```
{
    "code": "SUCCESS",
    "msg": "success",
    "data": {
         "first_name": "Fist Name",
         "last_name": "Last Name",
         "id": "37f2b996-c2c5-487b-aa22-8b453ff14a4b",
         "user_email": "example@*.com"
     }
}
```

## Request Sample

The request body should be a JSON object containing the following fields:

```
curl '{{host}}/api/v1/developer/users'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data-raw '{
        "first_name":"H",
        "last_name":"L",
        "employee_number":"100000",
        "onboard_time":1689150139,
        "user_email": "example@*.com"
    }'
    --insecure
```

# 3.3 Update User

This API allows you to update user details.

- Request URL: `/api/v1/developer/users/:id`
- Permission Key: `edit:user`
- Method: `PUT`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | Example |
| --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the user. | 348e868e-534a-4ace-ba77-ce80394e31e3 |

## Request Body

| Parameter | Required | Type | Description | Example | How to Get It? |
| --- | --- | --- | --- | --- | --- |
| first_name | F | String | First name of the user. | | |
| last_name | F | String | Last name of the user. | | |
| user_email | F | String | Email of the user. **UniFi Access Requirement:** `1.22.16 or later` | | |
| employee_number | F | String | Employee number of the user. | | |
| onboard_time | F | Integer | User onboarding date. | 1689150139 | |
| status | F | String | Status of the user. | "ACTIVE" "DEACTIVATED" | |

## Response Body

> Schemas: Schemas

## Response Sample

```
{
    "code": "SUCCESS",
    "data": null,
    "msg": "success"
}
```

## Request Sample

The request body should be a JSON object containing the following fields:

```
curl -XPUT '{{host}}/api/v1/developer/user'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data-raw '{
        "first_name":"H",
        "last_name":"L",
        "employee_number":"",
        "user_email":"example@*.com",
        "pin_code":"",
        "onboard_time":1689150139,
        "status": "ACTIVE"
    }'
    --insecure
```

# 3.4 Fetch User

This API allows you to fetch user details.

- Request URL: `/api/v1/developer/users/:id`
- Permission Key: `view:user`
- Method: `GET`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | Example |
| --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the user. | 348e868e-534a-4ace-ba77-ce80394e31e3 |

## Query Parameters

| Parameter | Required | Type | Description | Example |
| --- | --- | --- | --- | --- |
| expand[] | F | Boolean | Determine whether to return the access policies assigned to a user (Optional). | expand[]=access_policy |

## Response Body

> Schemas: Schemas

## Response Sample


```
{
    "code": "SUCCESS",
    "data": {
        "access_policies": [
            {
                "id": "edbc80df-3698-49fd-8b53-f1867f104947",
                "name": "test",
                "resources": [
                    {
                        "id": "d5573467-d6b3-4e8f-8e48-8a322b91664a",
                        "type": "door_group"
                    },
                    {
                        "id": "5c496423-6d25-4e4f-8cdf-95ad5135188a",
                        "type": "door_group"
                    },
                    {
                        "id": "6ff875d2-af87-470b-9cb5-774c6596afc8",
                        "type": "door"
                    }
                ],
                "schedule_id": "73facd6c-839e-4521-a4f4-c07e1d44e748"
            }
        ],
        "access_policy_ids": [
            "edbc80df-3698-49fd-8b53-f1867f104947"
        ],
        "employee_number": "",
        "first_name": "***",
        "id": "17d2f099-99df-429b-becb-1399a6937e5a",
```

```
        "last_name": "L",
        "user_email": "example@*.com",
        "nfc_cards": [
            {
                "id": "100001",
                "token":
"d27822fc682b478dc637c6db01813e465174df6e54ca515d8427db623cfda1d0",
                "type": "ua_card"
            }
        ],
        "onboard_time": 1689047588,
        "pin_code": {
            "token":
"5f742ee4424e5a7dd265de3461009b9ebafa1fb9d6b15018842055cc0466ac56"
        },
        "status": "ACTIVE"
    },
    "msg": "success"
}
```


## Request Sample

The request body should be a JSON object containing the following fields:

```
curl -XGET '{{host}}/api/v1/developer/users/348e868e-534a-4ace-ba77-ce80394e31e3?
expand[]=access_policy'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure
```

# 3.5 Fetch All Users

This API allows you to fetch all users.

- Request URL: `/api/v1/developer/users`
- Permission Key: `view:user`
- Method: `GET`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Query Parameters

| Parameter | Required | Type | Description | Example |
|---|---|---|---|---|
| expand[] | F | Boolean | Determine whether to return the access policies assigned to a user (Optional). | expand[]=access_policy |
| page_num | F | String | Current page number in the pagination. | 1 |
| page_size | F | String | Number of users per page. | 25 |

## Response Body

> Schemas: Schemas

## Response Sample


```
{
    "code": "SUCCESS",
    "data": [
        {
            "access_policies": [
                {
                    "id": "73f15cab-c725-4a76-a419-a4026d131e96",
                    "name": "Default Admin Policy",
                    "resources": [
                        {
                            "id": "d5573467-d6b3-4e8f-8e48-8a322b91664a",
                            "type": "door_group"
                        },
                        {
                            "id": "5c496423-6d25-4e4f-8cdf-95ad5135188a",
                            "type": "door_group"
                        }
                    ],
                    "schedule_id": "73facd6c-839e-4521-a4f4-c07e1d44e748"
                }
            ],
            "access_policy_ids": [
                "73f15cab-c725-4a76-a419-a4026d131e96"
            ],
            "employee_number": "",
            "first_name": "UniFi",
            "id": "83569f9b-0096-48ab-b2e4-5c9a598568a8",
            "last_name": "User",
            "user_email": "",
            "nfc_cards": [],
            "onboard_time": 0,
            "pin_code": null,
            "status": "ACTIVE"
```

```
        },
        {
            "access_policies": [
                {
                    "id": "c1682fb8-ef6e-4fe2-aa8a-b6f29df753ff",
                    "name": "policy_1690272668035",
                    "resources": [
                        {
                            "id": "6ff875d2-af87-470b-9cb5-774c6596afc8",
                            "type": "door"
                        }
                    ],
                    "schedule_id": "0616ef06-b807-4372-9ae0-7a87e12e4019"
                }
            ],
            "access_policy_ids": [
                "c1682fb8-ef6e-4fe2-aa8a-b6f29df753ff"
            ],
            "employee_number": "",
            "first_name": "Ttttt",
            "id": "3a3ba57a-796e-46e0-b8f3-478bb70a114d",
            "last_name": "Tttt",
            "nfc_cards": [],
            "onboard_time": 1689048000,
            "pin_code": null,
            "status": "ACTIVE"
        }
    ],
    "msg": "success",
    "pagination": {
        "page_num": 1,
        "page_size": 97,
        "total": 97
    }
}
```


## Request Sample

The request body should be a JSON object containing the following fields:

```
curl -XGET '{{host}}/api/v1/developer/users?
page_num=1&page_size=25&expand[]=access_policy'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure
```

# 3.6 Assign Access Policy to User

- Request URL: `/api/v1/developer/users/:id/access_policies`
- Permission Key: `edit:user`
- Method: `PUT`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the user. | Get it from the API `/api/v1/developer/users` |

## Request Body

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| access_policy_ids | T | Array[String] | Collection of multiple policy IDs. | Get it from the API `/api/v1/developer/access_policies` |

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
//assign access policies to user
curl -XPUT '{{host}}/api/v1/developer/users/38857332-7a5e-4bb6-8837-651b2a47cea5/access_policies'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data-raw '{
        "access_policy_ids":[
            "03895c7f-9f53-4334-812b-5db9c122c109",
            "3b6bcb0c-7498-44cf-8615-00a96d824cbe"
        ]
    }'
    --insecure

//remove all access policies from user
curl -XPUT '{{host}}/api/v1/developer/users/38857332-7a5e-4bb6-8837-651b2a47cea5/access_policies'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
```

```
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data-raw '{
        "access_policy_ids":[]
    }'
    --insecure
```

## Response Sample

```
{"code":"SUCCESS","msg":"success"}
```

# 3.7 Assign NFC Card to User

- Request URL: `/api/v1/developer/users/:id/nfc_cards`
- Permission Key: `edit:user`
- Method: `PUT`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the user. | Get it from the API `/api/v1/developer/users` |

## Request Body

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| token | T | String | Token of the NFC card. | Get it from the API `/api/v1/developer/nfc_cards/sessions/{session_id}` |
| force_add | F | Boolean | Determine whether to overwrite an NFC card that has already been assigned. | true or false |

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl -XPUT '{{host}}/api/v1/developer/users/17d2f099-99df-429b-becb-1399a6937e5a/nfc_cards'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data-raw '{
        "token": "d27822fc682b478dc637c6db01813e465174df6e54ca515d8427db623cfda1d0",
        "force_add": true
    }'

```

## Response Sample

```
{"code":"SUCCESS","msg":"success"}
```

# 3.8 Unassign NFC Card from User

- Request URL: `/api/v1/developer/users/:id/nfc_cards/delete`
- Permission Key: `edit:user`
- Method: `PUT`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the user. | Get it from the API `/api/v1/developer/users` |

## Request Body

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| token | T | String | Token of the NFC card. | Get it from the API `/api/v1/developer/users` |

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl -XDELETE '{{host}}/api/v1/developer/users/17d2f099-99df-429b-becb-1399a6937e5a/nfc_cards/delete'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    -d '{
        "token": "d27822fc682b478dc637c6db01813e465174df6e54ca515d8427db623cfda1d0"
    }'
    --insecure

```

## Response Sample

```
{"code":"SUCCESS","msg":"success"}
```

# 3.9 Assign PIN Code to User

- Request URL: `/api/v1/developer/users/:id/pin_codes`
- Permission Key: `edit:user`
- Method: `PUT`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the user. | Get it from the API `/api/v1/developer/users` |

## Request Body

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| pin_code | T | String | Generate a PIN code for the user to unlock doors. | Get it from the API `/api/v1/developer/pin_codes` |

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl -XPUT '{{host}}/api/v1/developer/users/17d2f099-99df-429b-becb-1399a6937e5a/pin_codes'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data-raw '{
        "pin_code": "57301208"
    }'
    --insecure

```

## Response Sample

```
{"code":"SUCCESS","msg":"success"}
```

# 3.10 Unassign PIN Code from User

- Request URL: `/api/v1/developer/users/:id/pin_codes`
- Permission Key: `edit:user`
- Method: `DELETE`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the user. | Get it from the API `/api/v1/developer/users` |

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl -XDELETE '{{host}}/api/v1/developer/users/17d2f099-99df-429b-becb-1399a6937e5a/pin_codes'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure

```

## Response Sample

```
{"code":"SUCCESS","msg":"success"}
```

# 3.11 Create User Group

- Request URL: `/api/v1/developer/user_groups`
- Permission Key: `edit:user`
- Method: `POST`
- UniFi Access Requirement: `2.2.6 or later`
- API version: `v1`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Body

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| name | T | String | Name of the group. | |
| up_id | F | String | Parent group ID **(Optional)** | Get it from the API `/api/v1/developer/user_groups` |

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl -XPOST '{{host}}/api/v1/developer/user_groups'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data-raw '{
        "name": "Group Name",
        "up_id": "013d05d3-7262-4908-ba69-badbbbf8f5a6"
    }'
    --insecure

```

## Response Sample

```
{"code":"SUCCESS","msg":"success"}
```

# 3.12 Fetch All User Groups

- Request URL: `/api/v1/developer/user_groups`
- Permission Key: `view:user`
- Method: `GET`
- UniFi Access Requirement: `2.2.6 or later`
- API version: `v1`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl -XGET '{{host}}/api/v1/developer/developer/user_groups'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure

```

## Response Sample

```
{
    "code": "SUCCESS",
    "data": [
        {
            "full_name": "Group Name",
            "id": "75011ee6-b7ab-4927-9d9f-dd08ef0a3199",
            "name": "Group Name",
            "up_id": "a27899fc-a2d1-4797-8d4d-86118f8555f3",
            "up_ids": [
                "a27899fc-a2d1-4797-8d4d-86118f8555f3"
            ]
        }
    ],
    "msg": "success"
}
```

# 3.13 Fetch User Group

- Request URL: `/api/v1/developer/user_groups/:id`
- Permission Key: `view:user`
- Method: `GET`
- UniFi Access Requirement: `2.2.6 or later`
- API version: `v1`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the user group. | Get it from the API `/api/v1/developer/user_groups` |

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl -XGET '{{host}}/api/v1/developer/developer/user_groups/75011ee6-b7ab-4927-9d9f-dd08ef0a3199'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure

```

## Response Sample


```
{
    "code": "SUCCESS",
    "data": {
        "full_name": "Group Name",
        "id": "75011ee6-b7ab-4927-9d9f-dd08ef0a3199",
        "name": "Group Name",
        "up_id": "a27899fc-a2d1-4797-8d4d-86118f8555f3",
        "up_ids": [
            "a27899fc-a2d1-4797-8d4d-86118f8555f3"
        ]
    },
    "msg": "success"
}
```


# 3.14 Update User Group

- Request URL: `/api/v1/developer/user_groups/:id`
- Permission Key: `edit:user`
- Method: `PUT`
- UniFi Access Requirement: `2.2.6 or later`
- API version: `v1`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the user group. | Get it from the API `/api/v1/developer/user_groups` |

## Request Body

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| name | T | String | Name of the group. | |
| up_id | F | String | Parent group ID | Get it from the API `/api/v1/developer/user_groups` |

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl -XPUT '{{host}}/api/v1/developer/user_groups/75011ee6-b7ab-4927-9d9f-dd08ef0a3199'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data-raw '{
        "name": "Group Name",
        "up_id": "013d05d3-7262-4908-ba69-badbbbf8f5a6"
    }'
    --insecure

```

## Response Sample

```
{"code":"SUCCESS","msg":"success"}
```

# 3.15 Delete User Group

- Request URL: `/api/v1/developer/user_groups/:id`
- Permission Key: `edit:user`
- Method: `DELETE`
- UniFi Access Requirement: `2.2.6 or later`
- API version: `v1`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the user group. | Get it from the API `/api/v1/developer/user_groups` |

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl -XDELETE '{{host}}/api/v1/developer/developer/user_groups/75011ee6-b7ab-4927-9d9f-dd08ef0a3199'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure

```

## Response Sample

```
{"code":"SUCCESS","msg":"success"}
```

# 3.16 Assign User to User Group

- Request URL: `/api/v1/developer/user_groups/:id/users`
- Permission Key: `edit:user`
- Method: `POST`
- UniFi Access Requirement: `2.2.6 or later`
- API version: `v1`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the user group. | Get it from the API `/api/v1/developer/user_groups` |

## Request Body

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| | T | Array[String] | Identity ID of the user. | Get it from the API `/api/v1/developer/users` |

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl -XPOST '{{host}}/api/v1/developer/user_groups/75011ee6-b7ab-4927-9d9f-dd08ef0a3199/users'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data-raw '[
        "7c6e9102-acb7-4b89-8ed4-7561e6fb706c",
        "fd63bc4c-52e0-4dbf-a699-e1233339c73b"
    ]'
    --insecure

```

## Response Sample

```
{"code":"SUCCESS","msg":"success"}
```

# 3.17 Unassign User from User Group

- Request URL: `/api/v1/developer/user_groups/:id/users/delete`
- Permission Key: `edit:user`
- Method: `POST`
- UniFi Access Requirement: `2.2.6 or later`
- API version: `v1`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the user group. | Get it from the API `/api/v1/developer/user_groups` |

## Request Body

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| | T | Array[String] | Identity ID of the user. | Get it from the API `/api/v1/developer/users` |

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl -XPOST '{{host}}/api/v1/developer/user_groups/75011ee6-b7ab-4927-9d9f-dd08ef0a3199/users/delete'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data-raw '[
        "7c6e9102-acb7-4b89-8ed4-7561e6fb706c",
        "fd63bc4c-52e0-4dbf-a699-e1233339c73b"
    ]'
    --insecure

```

## Response Sample

```
{"code":"SUCCESS","msg":"success"}
```

# 3.18 Fetch Users in a User Group

> This API allows you to fetch only the users in a user group, excluding any subgroups.

- Request URL: `/api/v1/developer/user_groups/:id/users`
- Permission Key: `view:user`
- Method: `GET`

- UniFi Access Requirement: `2.2.6 or later`
- API version: `v1`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the user group. | Get it from the API `/api/v1/developer/user_groups` |

## Response Body

> Schemas: Schemas

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl -XGET '{{host}}/api/v1/developer/user_groups/23676a54-382e-4121-aa80-878d2d9bacaa/users'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure

```

## Response Sample


```
{
    "code": "SUCCESS",
    "data": [
        {
            "alias": "",
            "avatar_relative_path": "",
            "email": "*@*.com",
            "email_status": "UNVERIFIED",
            "employee_number": "1000000",
            "first_name": "",
            "full_name": "",
            "id": "27aa91ac-2924-43d4-82e1-24b6a570d29e",
            "last_name": "Chen",
            "onboard_time": 1689150139,
            "phone": "",
```

```
            "status": "ACTIVE",
            "user_email": "",
            "username": ""
        }
    ],
    "msg": "success"
}
```

# 3.19 Fetch All Users in a User Group

> This API allows you to fetch all users in a user group, including those in subgroups.

- Request URL: `/api/v1/developer/user_groups/:id/users/all`
- Permission Key: `view:user`
- Method: `GET`
- UniFi Access Requirement: `2.2.6 or later`
- API version: `v1`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the user group. | Get it from the API `/api/v1/developer/user_groups` |

## Response Body

> Schemas: Schemas

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl -XGET '{{host}}/api/v1/developer/user_groups/23676a54-382e-4121-aa80-878d2d9bacaa/users/all'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure

```

## Response Sample


```
{
    "code": "SUCCESS",
    "data": [
        {
            "alias": "",
            "avatar_relative_path": "",
            "email": "*@*.com",
            "email_status": "UNVERIFIED",
            "employee_number": "1000000",
            "first_name": "",
            "full_name": "",
            "id": "27aa91ac-2924-43d4-82e1-24b6a570d29e",
            "last_name": "Chen",
            "onboard_time": 1689150139,
            "phone": "",
            "status": "ACTIVE",
            "user_email": "",
            "username": ""
        }
    ],
    "msg": "success"
}
```


# 3.20 Fetch the Access Policies Assigned to a User

> This API allows you to fetch the access policies assigned to a user.

- Request URL: `/api/v1/developer/users/:id/access_policies`
- Permission Key: `view:user`
- Method: `GET`
- UniFi Access Requirement: `2.2.6 or later`
- API version: `v1`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the user. | Get it from the API `/api/v1/developer/users` |

## Query Parameters

| Parameter | Required | Type | Description | Example |
| --- | --- | --- | --- | --- |
| only_user_policies | F | Boolean | If 'only_user_policies' is set to false, all policies of the group the user belongs to are displayed. If set to true, only the policies assigned directly to the user are shown. | only_user_policies=true |

## Response Body

> Schemas: Schemas

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl -XGET '{{host}}/users/27aa91ac-2924-43d4-82e1-24b6a570d29e/access_policies?only_user_policies=false'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure

```

## Response Sample


```
{
    "code": "SUCCESS",
    "data": [
        {
            "id": "89a4ca95-1502-4ae7-954f-d986b67afe5c",
            "name": "Default Site Policy",
            "resources": [
                {
                    "id": "fd2a06e2-81af-4cf4-9bd5-8bceb5e7b7d7",
                    "type": "door_group"
                }
            ],
            "schedule_id": "6b79d12a-2a6e-4463-949c-f1a98fff40d2"
        },
        {
            "id": "bbe48a65-2ac1-4bf6-bd65-bc8f9ee7fb75",
            "name": "Access Policy Name",
```

```
            "resources": [],
            "schedule_id": "f7414bcd-f0cc-4d3e-811a-b5ac75f7ddb8"
        }
    ],
    "msg": "success"
}
```

# 3.21 Assign Access Policy to User Group

> This API is used to assign access policies to a user group.

- Request URL: `/api/v1/developer/user_groups/:id/access_policies`
- Permission Key: `edit:user`
- Method: `PUT`
- UniFi Access Requirement: `2.2.6 or later`
- API version: `v1`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the user group. | Get it from the API `/api/v1/developer/user_groups` |

## Request Body

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| access_policy_ids | T | Array[String] | Identity ID of the access policy. | Get it from the API `/api/v1/developer/access_policies` |

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl -XPUT '{{host}}/user_groups/23676a54-382e-4121-aa80-878d2d9bacaa/access_policies'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data '{
        "access_policy_ids": [
            "bbe48a65-2ac1-4bf6-bd65-bc8f9ee7fb75"
         ]
    }'
    --insecure

```

## Response Sample

```
{
    "code": "SUCCESS",
    "data": null,
    "msg": "success"
}
```

# 3.22 Fetch the Access Policies Assigned to a User Group

> This API allows you to fetch the access policies assigned to a user group.

- Request URL: `/api/v1/developer/user_groups/:id/access_policies`
- Permission Key: `view:user`
- Method: `GET`
- UniFi Access Requirement: `2.2.6 or later`
- API version: `v1`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the user group. | Get it from the API `/api/v1/developer/user_groups` |

## Response Body

> Schemas: Schemas

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl -XGET '{{host}}/user_groups/23676a54-382e-4121-aa80-878d2d9bacaa/access_policies'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure

```

## Response Sample


```
{
    "code": "SUCCESS",
    "data": [
        {
            "id": "89a4ca95-1502-4ae7-954f-d986b67afe5c",
            "name": "Default Site Policy",
            "resources": [
                {
                    "id": "fd2a06e2-81af-4cf4-9bd5-8bceb5e7b7d7",
                    "type": "door_group"
                }
            ],
            "schedule_id": "6b79d12a-2a6e-4463-949c-f1a98fff40d2"
        },
        {
            "id": "bbe48a65-2ac1-4bf6-bd65-bc8f9ee7fb75",
            "name": "Access Policy Name",
            "resources": [],
            "schedule_id": "f7414bcd-f0cc-4d3e-811a-b5ac75f7ddb8"
        }
    ],
    "msg": "success"
}
```


# 3.23 Delete User

> This API allows you to delete a user whose status is disabled.

- Request URL: `/api/v1/developer/users/:id`
- Permission Key: `edit:user`
- Method: `DELETE`
- UniFi Access Requirement: `3.1.30 or later`
- API version: `v1`
- **Note**: Only users with a **disabled** can be deleted.

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the user. | Get it from the API `/api/v1/developer/users` |

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl -XDELETE '{{host}}/api/v1/developer/developer/users/348e868e-534a-4ace-ba77-ce80394e31e3'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure

```

## Response Sample

```
{"code":"SUCCESS","msg":"success"}
```

# 3.24 Search Users

This API allows you to fetch all users.

- Request URL: `/api/v1/developer/users/search`
- Permission Key: `view:user`
- Method: `GET`

- UniFi Access Requirement: `3.1.30 or later`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Query Parameters

| Parameter | Required | Type | Description | Example |
| --- | --- | --- | --- | --- |
| keyword | F | String | Determine whether to return the access policies assigned to a user (Optional). | keyword=Name |
| user_id | F | String | Support filtering by multiple user Identity IDs. | user_id=472cabd2-0634-4e85-9e8d-5a73b500516a&user_id=21472b1d-aa3a-4f2c-855e-0ec3dcaaeb5a |
| only_admin | F | Boolean | Filter to display only admin users. | only_admin=false |
| status | F | String | Support filtering by multiple user statuses. | status=ACTIVE&status=DEACTIVATED&status=PENDING |
| page_num | F | String | Current page number in the pagination. | 1 |
| page_size | F | String | Number of users per page. | 25 |

## Response Body

> Schemas: Schemas

## Response Sample


```
[
    {
        "alias": "",
        "avatar_relative_path": "/avatar/186b07b1-fa13-49b5-8954-399d1b9c5285",
        "email": "User1@*.com",
        "email_status": "VERIFIED",
        "employee_number": "",
        "first_name": "Name",
        "full_name": "Full Name",
        "id": "472cabd2-0634-4e85-9e8d-5a73b500516a",
        "last_name": "Last Name",
        "nfc_cards": [],
        "onboard_time": 0,
        "phone": "",
        "pin_code": null,
        "status": "ACTIVE",
        "user_email": "User1@*.com,
        "username": ""
    }
]
```

## Request Sample

The request body should be a JSON object containing the following fields:

```
curl -XGET '{{host}}/api/v1/developer/users/search?keyword=Name&user_id=472cabd2-0634-4e85-9e8d-5a73b500516a&user_id=21472b1d-aa3a-4f2c-855e-0ec3dcaaeb5a&only_admin=false&status=ACTIVE&status=DEACTIVATED&page_size=10&page_num=1'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure
```

# 4. Visitor

> The APIs here are designed for managing visitors, including creating, viewing, and deleting visitors. They also enable the assigning of NFC cards, schedules, PIN codes, locations, and other access resources to visitors.

## 4.1 Schemas

| Parameter | Type | Description |
|---|---|---|
| Id | String | Identity ID of the visitor. |
| first_name | String | First name of the visitor. |
| last_name | String | Last name of the visitor. |
| status | String | The visitor's status. `enum status {UPCOMING=1,VISITED=2,VISITING=3,CANCELLED=4,NO_VISIT=5,ACTIVE=6}` |
| inviter_id | String | Identity ID of the inviter. |
| inviter_name | String | Name of the inviter. |
| mobile_phone | String | Contact phone number of the visitor. |
| remarks | String | Remarks of the visitor. |
| email | String | Email of the visitor. |
| visitor_company | String | Company of the visitor. |
| visit_reason | String | Visit reason: `enum reason {Interview,Business,Cooperation,Others}` |
| start_time | Integer | Start time of the visit. |
| end_time | Integer | End time of the visit. |
| nfc_cards | Array[Object] | Token associated with the bound NFC card. |
| nfc_cards[].id | String | Display ID of the NFC card. |
| nfc_cards[].token | String | Unique NFC card token. |
| pin_code | Object | Token associated with the bound PIN code. |
| pin_code.token | String | The user's PIN hash code credential for unlocking a door. |
| schedule_id | String | Identity ID of the schedule. |
| schedule | Object | The schedule assigned to the visitor. **If the `schedule` information is present, it indicates that the visit schedule is recurring. If the `schedule` information is not included, it indicates a one-time visit schedule.** |
| schedule.name | String | Name of the schedule. |
| schedule.type | String | Type of the schedule. |
| schedule.week_schedule | Object | The customizable scheduling strategy for each day from Sunday to Saturday. If not specified, it means access is allowed every day. |
| schedule.week_schedule.monday | Array[Object] | Specify the daily access time period from Sunday to Saturday. |
| schedule.week_schedule.monday[0].start_time | String | Start time of the access time period. |
| schedule.week_schedule.monday[0].end_time | String | End time of the access time period. |
| resources | Array[Object] | Specify the locations that the visitor can access. |
| resources[0].id | String | Identity ID of the door group. |
| resources[0].name | String | Name of the door group. |
| resources[0].type | String | Type of the door group. |

# 4.2 Create Visitor

This API enables you to create a new visitor.

- Request URL: `/api/v1/developer/visitors`
- Permission Key: `edit:visitor`
- Method: `POST`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Body

| Parameter | Required | Type | Description | Example | How to Get It? |
| --- | --- | --- | --- | --- | --- |
| first_name | T | String | First name of the visitor. | | |
| last_name | T | String | Last name of the visitor. | | |
| remarks | F | String | Remarks about the visitor. | | |
| mobile_phone | F | String | Mobile phone number of the visitor. | | |
| email | F | String | Email address of the visitor. | | |
| visitor_company | F | String | Company name of the visitor. | | |
| start_time | T | Integer | Start time of the visit. | 1688546460 | |
| end_time | T | Integer | End time of the visit. | 1688572799 | |
| visit_reason | T | String | Visit reason: `enum reason {Interview,Business,Cooperation,Others}` | Interview | |

> Assign Available Locations to Visitor

| Parameter | Required | Type | Description | Example | How to Get It? |
| --- | --- | --- | --- | --- | --- |
| resources | F | Array[Object] | Assign available locations to the visitor. | | Get it from the API `/api/v1/developer/door_groups/topology` |
| resources[0].id | F | String | When resource type is `door_group`, it refers to the group ID of a building or a customized door group. When type is `door`, it refers to the ID of a door. | 9bee6e0e-108d-4c52-9107-76f2c7dea4f1 | |
| resources[0].type | F | String | Support both `door` and `door_group`. | door_group | |

> Assigned Schedules To Visitor
>
> If the `week_schedule` information is present, it indicates that the visit schedule is recurring. If the `week_schedule` information is not included, it indicates a one-time visit schedule.

| Parameter | Required | Type | Description | Example |
| --- | --- | --- | --- | --- |
| week_schedule | F | Object | The customizable scheduling strategy for each day from Sunday to Saturday. If not specified, it means access is allowed every day. | |
| week_schedule.monday | F | Array[Object] | Specify the daily access time period from Sunday to Saturday. | |
| week_schedule.monday[0].start_time | F | String | Start time of the access time period. | 00:00:00 |
| week_schedule.monday[0].end_time | F | String | End time of the access time period. | 23:59:59 |

# Response Body

> Schemas: Schemas

# Response Sample


```
{
    "code": "SUCCESS",
    "data": {
        "first_name": "H",
        "id": "fbe8d920-47d3-4cfd-bda7-bf4b0e26f73c",
        "last_name": "L",
        "nfc_cards": [],
        "resources": [
            {
                "id": "5c496423-6d25-4e4f-8cdf-95ad5135188a",
                "name": "Test Group",
                "type": "door_group"
            },
            {
                "id": "d5573467-d6b3-4e8f-8e48-8a322b91664a",
                "name": "UNVR",
                "type": "door_group"
            },
            {
                "id": "369215b0-cabe-49b6-aeaa-e0b7ec6424d5",
                "name": "visitor-1691671529285",
                "type": "door_group"
            }
        ],
        "schedule": {
            "id": "1fb849bb-e7e5-4516-8dd9-b78094a6708a",
            "is_default": false,
            "name": "schedule-1691671529237",
            "type": "access",
            "weekly": {
                "friday": [
                    {
                        "end_time": "17:00:59",
                        "start_time": "10:00:00"
                    }
```

```
                ],
                "monday": [],
                "saturday": [],
                "sunday": [],
                "thursday": [
                    {
                        "end_time": "17:00:59",
                        "start_time": "11:00:00"
                    }
                ],
                "tuesday": [
                    {
                        "end_time": "17:00:59",
                        "start_time": "10:00:00"
                    }
                ],
                "wednesday": [
                    {
                        "end_time": "17:00:59",
                        "start_time": "10:00:00"
                    }
                ]
            }
        },
        "schedule_id": "1fb849bb-e7e5-4516-8dd9-b78094a6708a",
        "status": "ACTIVE"
    },
    "msg": "success"
}
```


## Request Sample

The request body should be a JSON object containing the following fields:

```
curl '{{host}}/api/v1/developer/visitors'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'   --data-raw
    '{
        "first_name": "H",
        "last_name": "L",
        "remarks": "",
        "mobile_phone": "",
        "email": "",
        "visitor_company": "",
        "start_time": 1688546460,
        "end_time": 1788572799,
        "visit_reason": "Interviemw",
        "week_schedule": {
```

```
            "sunday": [],
            "monday": [],
            "tuesday": [
                {
                    "start_time": "10:00:00",
                    "end_time": "17:00:59"
                }
            ],
            "wednesday": [
                {
                    "start_time": "10:00:00",
                    "end_time": "17:00:59"
                }
            ],
            "thursday": [
                {
                    "start_time": "11:00:00",
                    "end_time": "17:00:59"
                }
            ],
            "friday": [
                {
                    "start_time": "10:00:00",
                    "end_time": "17:00:59"
                }
            ],
            "saturday": []
        },
        "resources": [
            {
                "id": "6ff875d2-af87-470b-9cb5-774c6596afc8",
                "type": "door"
            },
            {
                "id": "5c496423-6d25-4e4f-8cdf-95ad5135188a",
                "type": "door_group"
            },
            {
                "id": "d5573467-d6b3-4e8f-8e48-8a322b91664a",
                "type": "door_group"
            }
        ]
    }'
    --insecure
```

## 4.3 Fetch Visitor

This API enables you to fetch visitor details.

- Request URL: `/api/v1/developer/visitors/:id`
- Permission Key: `view:visitor`
- Method: `GET`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | |
| --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the visitor. | |

## Response Body

> Schemas: Schemas

## Response Sample


```
{
    "code": "SUCCESS",
    "data": {
        "first_name": "User1",
        "id": "3566867c-fa04-4752-98f6-43cf9a342d4a",
        "last_name": "Last Name",
        "nfc_cards": [
            {
                "id": "100001",
                "token": "d27822fc682b478dc637c6db01813e465174df6e54ca515d8427db623cfda1d0",
                "type": "ua_card"
            }
        ],
        "pin_code": {
            "token": "bc3e3135013e2dcae119390b7897166e8cec3bcf5becb6b05578ab67634559ed"
        },
        "resources": [
            {
                "id": "fd293ecb-98d2-425b-a020-cb9365ea48b3",
                "name": "visitor-1690337152955",
```

```
                "type": "door_group"
            }
        ],
        "schedule": {
            "id": "6ccf9e1e-b174-476d-b2fe-96817c780fbf",
            "is_default": false,
            "name": "visitor-1690337152955",
            "type": "access",
            "weekly": null
        },
        "schedule_id": "6ccf9e1e-b174-476d-b2fe-96817c780fbf",
        "status": "VISITED"
    },
    "msg": "success"
}
```


## Request Sample

The request body should be a JSON object containing the following fields:

```
curl -XGET '{{host}}/api/v1/developer/visitors/76794bd8-98c0-49b6-9230-ba8c5812cf29'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure
```

# 4.4 Fetch All Visitors

This API enables you to fetch the list of all visitors.

- Request URL: `/api/v1/developer/visitors`
- Permission Key: `view:visitor`
- Method: `GET`

## Request Header

| Parameter | Required | Type | Description |
|---|---|---|---|
| Authorization | T | String | Token required for authentication and access control. |

## Query Parameters

| Parameter | Required | Type | Description | Example |
| --- | --- | --- | --- | --- |
| status | F | Integer | The visitor's status. `enum status {UPCOMING=1,VISITED=2,VISITING=3,CANCELLED=4,NO_VISIT=5,ACTIVE=6}` | 1 |
| keyword | F | String | Support prefix matching for first name and last name. **NOTE**: The status filtering is disabled when searching with keyword. | H |
| page_num | F | String | Current page number in the pagination. | 1 |
| page_size | F | String | Number of visitors per page. | 25 |
| expand[] | F | string | Determine whether to return the objects (Optional). currently supports the following objects: `enum objects {none,access_policy,resource,schedule,nfc_card,pin_code}`. For the "none" option, it means that no object will be returned. **UniFi Access Requirement:** `1.22.16 or later` | expand[]=access_policy &expand[]=resource &expand[]=schedule &expand[]=nfc_card &expand[]=pin_code |

## Response Body

> Schemas: Schemas

## Request Sample

The request body should be a JSON object containing the following fields:

```
curl -XGET '{{host}}/api/v1/developer/visitors?page_num=1&page_size=25'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure

# fetch users through keyword
curl -XGET '{{host}}/api/v1/developer/visitors?keyword=H'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure

# using the "expands" options
curl -XGET '{{host}}/api/v1/developer/visitors?
expand[]=access_policy&expand[]=resource&expand[]=schedule&expand[]=nfc_card&expand
[]=pin_code'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure
```

# 4.5 Update Visitor

This API enables you to update a visitor.

- Request URL: `/api/v1/developer/visitors/:id`
- Permission Key: `edit:visitor`
- Method: `PUT`

# Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

# Request Body

| Parameter | Required | Type | Description | Example | How to Get It? |
| --- | --- | --- | --- | --- | --- |
| first_name | F | String | First name of the visitor. | | |
| last_name | F | String | Last name of the visitor. | | |
| remarks | F | String | Remarks about the visitor. | | |
| mobile_phone | F | String | Mobile phone number of the visitor. | | |
| email | F | String | Email address of the visitor. | | |
| visitor_company | F | String | Company name of the visitor. | | |
| start_time | F | Integer | Start time of the visit. | 1688546460 | |
| end_time | F | Integer | End time of the visit. | 1688572799 | |
| visit_reason | F | String | Visit reason: `enum reason {Interview,Business,Cooperation,Others}` | Interview | |

> Assign Available Locations to Visitor

| Parameter | Required | Type | Description | Example | How to Get It? |
| --- | --- | --- | --- | --- | --- |
| resources | F | Array[Object] | Assign available locations to the visitor. **Omit this parameter if it doesn't need to be updated.** | | Get it from the API `/api/v1/developer/door_groups/topology` |
| resources[0].id | F | String | When resource type is `door_group`, it refers to the group ID of a building or a customized door group. When type is `door`, it refers to the ID of a door. | 9bee6e0e-108d-4c52-9107-76f2c7dea4f1 | |
| resources[0].type | F | String | Support both `door` and `door_group`. | | |

> Assigned Schedules To Visitor

| Parameter | Required | Type | Description | Example |
| --- | --- | --- | --- | --- |
| week_schedule | F | Object | The customizable scheduling strategy for each day from Sunday to Saturday. If not specified, it means access is allowed every day. **Omit this parameter if it doesn't need to be updated.** | |
| week_schedule.monday | F | Array[Object] | Specify the daily access time period from Sunday to Saturday. | |
| week_schedule.monday[0].start_time | F | String | Start time of the access time period. | 00:00:00 |
| week_schedule.monday[0].end_time | F | String | End time of the access time period. | 23:59:59 |

## Request Path

| Parameter | Required | Type | Description | |
|---|---|---|---|---|
| id | T | String | Identity ID of the visitor. | |

## Response Body

> Schemas: [Schemas]
>
> **Note**: Status change is not supported.

## Response Sample


```
{
    "code": "SUCCESS",
    "data": {
        "first_name": "H",
        "id": "8564ce90-76ba-445f-b78b-6cca39af0130",
        "last_name": "L",
        "nfc_cards": [],
        "pin_code": null,
        "resources": [
            {
                "id": "5c496423-6d25-4e4f-8cdf-95ad5135188a",
                "name": "Door-Group-1",
                "type": "door_group"
            },
            {
                "id": "d5573467-d6b3-4e8f-8e48-8a322b91664a",
                "name": "UNVR",
                "type": "door_group"
            },
            {
                "id": "e311ca94-172c-49fe-9c91-49bd8ecef845",
                "name": "visitor-1691646856144",
                "type": "door_group"
            }
        ],
        "schedule": {
            "id": "c03bf601-0b90-4341-bce4-6061931e9f4e",
            "is_default": false,
            "name": "visitor-1691646856097",
            "type": "access",
            "weekly": {
                "friday": [
                    {
                        "end_time": "17:00:59",
```

```
                        "start_time": "10:00:00"
                    }
                ],
                "monday": [
                    {
                        "end_time": "17:00:59",
                        "start_time": "10:00:00"
                    }
                ],
                "saturday": [],
                "sunday": [],
                "thursday": [
                    {
                        "end_time": "18:00:59",
                        "start_time": "11:00:00"
                    }
                ],
                "tuesday": [],
                "wednesday": [
                    {
                        "end_time": "17:00:59",
                        "start_time": "10:00:00"
                    }
                ]
            }
        },
        "schedule_id": "c03bf601-0b90-4341-bce4-6061931e9f4e",
        "status": "ACTIVE"
    },
    "msg": "success"
}
```


## Request Sample

The request body should be a JSON object containing the following fields:

```
curl -XPUT '{{host}}/api/v1/developer/visitors/c81dfee6-5970-4938-bd30-40820e16ea01'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data-raw '
    {
        "first_name": "Test",
        "last_name": "L",
        "remarks": "",
        "mobile_phone": "",
        "email": "",
        "visitor_company": "",
```

```
        "start_time": 1688546460,
        "end_time": 1788572799,
        "visit_reason": "Interviemw",
        "week_schedule": {
            "sunday": [],
            "monday": [
                {
                    "start_time": "10:00:00",
                    "end_time": "17:00:59"
                }
            ],
            "tuesday": [],
            "wednesday": [
                {
                    "start_time": "10:00:00",
                    "end_time": "17:00:59"
                }
            ],
            "thursday": [
                {
                    "start_time": "11:00:00",
                    "end_time": "18:00:59"
                }
            ],
            "friday": [
                {
                    "start_time": "10:00:00",
                    "end_time": "17:00:59"
                }
            ],
            "saturday": []
        },
        "resources": [
            {
                "id": "6ff875d2-af87-470b-9cb5-774c6596afc8",
                "type": "door"
            },
            {
                "id": "5c496423-6d25-4e4f-8cdf-95ad5135188a",
                "type": "door_group"
            },
            {
                "id": "d5573467-d6b3-4e8f-8e48-8a322b91664a",
                "type": "door_group"
            }
        ]
    }'
    --insecure
```

# 4.6 Delete Visitor

This API enables you to delete a visitor.

- Request URL: `/api/v1/developer/visitors/:id`
- Permission Key: `edit:visitor`
- Method: `DELETE`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | |
| --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the visitor. | |

## Query Parameters

| Parameter | Required | Type | Description | Example |
| --- | --- | --- | --- | --- |
| is_force | F | Boolean | If is_force is passed, physically delete this visitor; otherwise, update to canceled status. | is_force=true |

## Response Sample

```
{
    "code": "SUCCESS",
    "data": null,
    "msg": "success"
}
```

## Request Sample

The request body should be a JSON object containing the following fields:

```
curl -XDELETE '{{host}}/api/v1/developer/visitors/c81dfee6-5970-4938-bd30-40820e16ea01?is_force=true'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure
```

# 4.7 Assign NFC Card To Visitor

- Request URL: `/api/v1/developer/visitors/:id/nfc_cards`
- Permission Key: `edit:visitor`
- Method: `PUT`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the visitor. | Get it from the API `/api/v1/developer/visitors` |

## Request Body

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| token | T | String | Token of the NFC card. | Get it from the API `/api/v1/developer/nfc_cards/sessions/{session_id}` |
| force_add | F | Boolean | Determine whether to overwrite an NFC card that has already been assigned. | true or false |

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl -XPUT '{{host}}/api/v1/developer/visitors/60b5c15e-9aff-4fc8-9547-d21d2e39c1ff/nfc_cards
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data-raw '{
        "token": "d27822fc682b478dc637c6db01813e465174df6e54ca515d8427db623cfda1d0",
        "force_add": true
    }'
    --insecure
```

## Response Sample

```
{"code":"SUCCESS","msg":"success"}
```

# 4.8 Unassign NFC Card From Visitor

- Request URL: `/api/v1/developer/visitors/:id/nfc_cards/delete`
- Permission Key: `edit:visitor`
- Method: `PUT`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the visitor. | Get it from the API `/api/v1/developer/visitors` |

## Request Body

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| token | T | String | Token of the NFC card. | Get it from the API `/api/v1/developer/visitors` |

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl -XPUT '{{host}}/api/v1/developer/visitors/60b5c15e-9aff-4fc8-9547-d21d2e39c1ff/nfc_cards/delete
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    -d '{
        "token": "d27822fc682b478dc637c6db01813e465174df6e54ca515d8427db623cfda1d0"
    }'
    --insecure
```

## Response Sample

```
{"code":"SUCCESS","msg":"success"}
```

# 4.9 Assign PIN Code To Visitor

- Request URL: `/api/v1/developer/visitors/:id/pin_codes`
- Permission Key: `edit:visitor`
- Method: `PUT`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the visitor. | Get it from the API `/api/v1/developer/visitors` |

## Request Body

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| pin_code | T | String | Generate a PIN code for the visitor to unlock doors. | Get it from the API `/api/v1/developer/pin_codes` |

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl -XPUT '{{host}}/api/v1/developer/visitors/17d2f099-99df-429b-becb-1399a6937e5a/pin_codes
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data-raw '{
        "pin_code": "57301208"
    }'
    --insecure
```

## Response Sample

```
{"code":"SUCCESS","msg":"success"}
```

# 4.10 Unassign PIN Code From Visitor

- Request URL: `/api/v1/developer/visitors/:id/pin_codes`
- Permission Key: `edit:visitor`
- Method: `DELETE`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the visitor. | Get it from the API `/api/v1/developer/visitors` |

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl -XDELETE '{{host}}/api/v1/developer/visitors/17d2f099-99df-429b-becb-1399a6937e5a/pin_codes
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure

```

## Response Sample

```
{"code":"SUCCESS","msg":"success"}
```

# 5. Access Policy

> The APIs here are designed for managing door access policies. These policies can be bound with access schedules, floors, and other door access resources.

## 5.1 Access Policy Schemas

| Parameter | Type | Description |
| --- | --- | --- |
| Id | | Identity ID of the access policy. |
| name | String | Name of the access policy. |
| resources | Array[Object] | Specify the locations that can be accessed. |
| resources[].type | String | Include door and door_group resources. `enum type {door,door_group}` |
| resources[].id | String | When resource type is `door_group`, it refers to the group ID of a building or a customized door group. When type is `door`, it refers to the ID of a door. |
| schedule_id | String | Identity ID of the schedule. |

## 5.2 Create Access Policy

This API allows you to create an access policy.

- Request URL: `/api/v1/developer/access_policies`
- Permission Key: `edit:policy`
- Method: `POST`

### Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

### Request Body

| Parameter | Required | Type | Description | How to Get It? | Example |
|---|---|---|---|---|---|
| name | T | String | Name of the access policy. | | |
| resource | F | Array[Object] | Used to assign accessible locations to the access policy. | Get it from the API `/api/v1/developer/door_groups/topology;` `/api/v1/developer/door_groups` | |
| resources[].type | F | String | `enum type {door,door_group}` | | |
| resources[].id | F | String | When resource type is `door_group`, it refers to the group ID of a building or a customized door group. When type is `door`, it refers to the ID of a door. | | |
| schedule_id | T | String | Identity ID of the schedule. Assign additional schedules. | Get it from the API `/api/v1/developer/access_policies/schedules` | |

# Response Body

| Parameter | Required | Type | Description | Related API | Purpose |
|---|---|---|---|---|---|
| id | T | String | Identity ID of the policy. | `/api/v1/developer/users/:user_id/access_policies` | Used to both assign an access policy to a user or unassign a policy from them. |

> Schemas: Schemas

# Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl '{{host}}/api/v1/developer/access_policies'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'   --data-raw
    '{
        "name": "test",
        "resource": [
            {
                "id": "6ff875d2-af87-470b-9cb5-774c6596afc8",
                "type": "door"
            },
            {
                "id": "5c496423-6d25-4e4f-8cdf-95ad5135188a",
                "type": "door_group"
            },
             {
                "id": "d5573467-d6b3-4e8f-8e48-8a322b91664a",
                "type": "door_group"
            }
        ],
        "schedule_id": "4e108aeb-ec9a-4822-bf86-170ea986f934"
    }'
```

```
    --insecure
```

## Response Sample

```
{
    "code": "SUCCESS",
    "data": {
        "id": "bb5eb965-42dc-4206-9654-88a2d1c3aaa5",
        "name": "test13",
        "resources": [
            {
                "id": "6ff875d2-af87-470b-9cb5-774c6596afc8",
                "type": "door"
            },
            {
                "id": "5c496423-6d25-4e4f-8cdf-95ad5135188a",
                "type": "door_group"
            },
            {
                "id": "d5573467-d6b3-4e8f-8e48-8a322b91664a",
                "type": "door_group"
            }
        ],
        "schedule_id": "4e108aeb-ec9a-4822-bf86-170ea986f934"
    },
    "msg": "success"
}
```

# 5.3 Update Access Policy

This API allows you to update a policy.

- Request URL: `/api/v1/developer/access_policies/:id`
- Permission Key: `edit:policy`
- Method: `PUT`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | Example |
| --- | --- | --- | --- | --- |
| Id | T | String | Identity ID of the access policy. | |

## Request Body

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| name | F | String | Name of the access policy. **Omit this parameter if it doesn't need to be updated.** | |
| resource | F | Array[Object] | Used to assign accessible locations to the access policy. **Omit this parameter if it doesn't need to be updated.** | Get it from the API `/api/v1/developer/door_groups/topology`; `/api/v1/developer/door_groups` |
| resources[].type | F | String | `enum type {door,door_group}` | |
| resources[].id | F | String | When resource type is `door_group`, it refers to the group ID of a building or a customized door group. When type is `door`, it refers to the ID of a door. | |
| schedule_id | F | String | Identity ID of the schedule. Assign additional schedules. **Omit this parameter if it doesn't need to be updated.** | Get it from the API `/api/v1/developer/access_policies/schedules` |

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl '{{host}}/api/v1/developer/access_policies/242c88e3-0524-42de-8447-45891c5df714'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'   --data-raw
    '{
        "name": "test",
        "resource": [
            {
                "id": "6ff875d2-af87-470b-9cb5-774c6596afc8",
                "type": "door"
            },
            {
                "id": "5c496423-6d25-4e4f-8cdf-95ad5135188a",
                "type": "door_group"
            },
             {
                "id": "d5573467-d6b3-4e8f-8e48-8a322b91664a",
                "type": "door_group"
            }
        ],
        "schedule_id": "4e108aeb-ec9a-4822-bf86-170ea986f934"
```

```
    }'
    --insecure
```

## Response Body

| Parameter | Required | Type | Description | Related api | Purpose |
| --- | --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the policy. | `/api/v1/developer/users/:user_id/access_policies` | Used to both assign an access policy to a user or unassign a policy from them. |

> Schemas: Schemas

# 5.4 Delete Access Policy

This API allows you to delete an access policy.

- Request URL: `/api/v1/developer/access_policies/:id`
- Permission Key: `edit:policy`
- Method: `DELETE`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Body

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Id | T | String | Identity ID of the access policy. |

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl -XDELETE '{{host}}/api/v1/developer/access_policies/460d0bcc-5d4f-4e7b-8a3c-8d4502765e11'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    --insecure
```

## Response Sample

```
{"code":"SUCCESS","msg":"success","data":"success"}
```

# 5.5 Fetch Access Policy

This API allows you to fetch a  policy details.

- Request URL: `/api/v1/developer/access_policies/:id`
- Permission Key: `view:policy`
- Method: `GET`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Body

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Id | T | String | Identity ID of the access policy. |

## Response Body

| Parameter | Required | Type | Description | Example | |
| --- | --- | --- | --- | --- | --- |
| Data | T | Object | | | |

> Schemas: Schemas

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl '{{host}}/api/v1/developer/ccess_policy'
   -H 'Authorization: Bearer wHFmHR******kD6wHg'
   -H 'accept: application/json'
   -H 'content-type: application/json'
   --insecure
```

## Response Sample


```
{
    "code": "SUCCESS",
    "data": {
        "id": "ed09985f-cf52-486e-bc33-377b6ed7bbf2",
        "name": "test11",
        "resources": [
            {
                "id": "6ff875d2-af87-470b-9cb5-774c6596afc8",
                "type": "door"
            },
            {
                "id": "5c496423-6d25-4e4f-8cdf-95ad5135188a",
                "type": "door_group"
            },
            {
                "id": "d5573467-d6b3-4e8f-8e48-8a322b91664a",
                "type": "door_group"
            }
        ],
        "schedule_id": "4e108aeb-ec9a-4822-bf86-170ea986f934"
    },
    "msg": "success"
}
```


# 5.6 Fetch All Access Policies

This API allows you to fetch all access policies.

- Request URL: `/api/v1/developer/access_policies`
- Permission Key: `view:policy`
- Method: `GET`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Response Body

| Parameter | Required | Type | Description | Example | |
|---|---|---|---|---|---|
| Data | T | Array[Object] | | | |

> Schemas: Schemas

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl '{{host}}/api/v1/developer/access_policies'
   -H 'Authorization: Bearer wHFmHR******kD6wHg'
   -H 'accept: application/json'
   -H 'content-type: application/json'
   --insecure
```

## Response Sample


```
{
    "code": "SUCCESS",
    "data": [
        {
            "id": "73f15cab-c725-4a76-a419-a4026d131e96",
            "name": "Default Admin Policy",
            "resources": [
                {
                    "id": "d5573467-d6b3-4e8f-8e48-8a322b91664a",
                    "type": "door_group"
                },
                {
                    "id": "5c496423-6d25-4e4f-8cdf-95ad5135188a",
                    "type": "door_group"
                }
            ],
            "schedule_id": "73facd6c-839e-4521-a4f4-c07e1d44e748"
        },
        {
            "id": "b96948a4-fed9-40a3-9c4a-e473822a3db7",
            "name": "Default UNVR Policy",
            "resources": [
                {
                    "id": "d5573467-d6b3-4e8f-8e48-8a322b91664a",
                    "type": "door_group"
                },
                {
                    "id": "6ff875d2-af87-470b-9cb5-774c6596afc8",
```

```
                    "type": "door"
                }
            ],
            "schedule_id": "58c0f89b-f35c-4d2c-af7b-8b8918df2c31"
        },
        {
            "id": "edbc80df-3698-49fd-8b53-f1867f104947",
            "name": "TEST",
            "resources": [
                {
                    "id": "d5573467-d6b3-4e8f-8e48-8a322b91664a",
                    "type": "door_group"
                },
                {
                    "id": "5c496423-6d25-4e4f-8cdf-95ad5135188a",
                    "type": "door_group"
                },
                {
                    "id": "6ff875d2-af87-470b-9cb5-774c6596afc8",
                    "type": "door"
                }
            ],
            "schedule_id": "73facd6c-839e-4521-a4f4-c07e1d44e748"
        }
    ],
    "msg": "success"
}
```


## 5.7 Holiday Group Schemas

> A holiday group refers to a collection of holidays.

| Parameter | Type | Description |
| --- | --- | --- |
| id | String | Identity ID of the holiday group. |
| name | String | Name of the holiday group. |
| is_default | Boolean | Indicate whether the holiday group is the system default. |
| description | String | Description of the holiday group. |
| holidays | Array[Object] | Show a list of the holidays within the holiday group. |
| holidays[].description | String | Description of the holiday. |
| holidays[].id | String | Identity ID of the holiday. |
| holidays[].name | String | Name of the holiday. |
| holidays[].repeat | Boolean | Indicate whether the holiday repeats annually. |
| holidays[].start_time | String | Start time of the holiday, provided in UTC format according to RFC3339. |
| holidays[].end_time | String | End time of the holiday, provided in UTC format according to RFC3339. |

# 5.8 Create Holiday Group

This API allows you to create a holiday group.

- Request URL: `/api/v1/developer/access_policies/holiday_groups`
- Permission Key: `edit:policy`
- Method: `POST`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Body

| Parameter | Required | Type | Description | Example |
| --- | --- | --- | --- | --- |
| name | T | String | Name of the holiday group. | |
| description | F | String | Description of the holiday group. | |
| holidays | F | Array[Object] | Show a list of the holidays within the holiday group. | |
| holidays[].description | F | String | Description of the holiday. | |
| holidays[].name | F | String | Name of the holiday. | |
| holidays[].repeat | F | Boolean | Indicate whether the holiday repeats annually. | |
| holidays[].is_template | F | Boolean | Indicate whether the holiday is created using a holiday group template. | |
| holidays[].start_time | F | String | Start time of the holiday, provided in UTC format according to RFC3339. | 2023-08-25T00:00:00Z |
| holidays[].end_time | F | String | End time of the holiday, provided in UTC format according to RFC3339. | 2023-08-26T00:00:00Z |

# Response Body

| Parameter | Required | Type | Description | Related API | Purpose |
| --- | --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the holiday group. | `/api/v1/developer/access_policies/schedules` | Used to add a holiday group to the schedule. |

> Schemas: Schemas

# Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```shell
curl '{{host}}/api/v1/developer/access_policies/holiday_groups'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'   --data-raw
    '{
        "name": "Holiday Group-169286791557142",
        "holidays": [
            {
                "name": "Holiday Name 1",
                "description": "",
                "repeat": false,
                "start_time": "2023-08-25T00:00:00Z",
                "end_time": "2023-08-26T00:00:00Z"
            },
            {
                "name": "Holiday Name 2",
                "description": "",
                "repeat": false,
                "start_time": "2023-08-26T00:00:00Z",
                "end_time": "2023-08-27T00:00:00Z"
            }
        ]
```

```
    }'
    --insecure
```

## Response Sample

```
{
    "code": "SUCCESS",
    "data": {
        "description": "",
        "holidays": [
            {
                "description": "",
                "end_time": "2023-08-26 00:00:00Z",
                "id": "8900533d-03be-4f84-832d-54ff59905759",
                "name": "Holiday Name 1",
                "repeat": false,
                "start_time": "2023-08-25 00:00:00Z"
            },
            {
                "name": "holiday-2023-08-26",
                "end_time": "2023-08-27 00:00:00Z",
                "id": "9fff81cc-d476-40c4-80f9-d510451ce2cd",
                "name": "Holiday Name 2",
                "repeat": false,
                "start_time": "2023-08-26 00:00:00Z"
            }
        ],
        "id": "7be7a7a0-818f-4f76-98c3-1c38957f4dca",
        "is_default": false,
        "name": "Holiday Group-169286791557142",
        "template_name": ""
    },
    "msg": "success"
}
```

# 5.9 Update Holiday Group

This API allows you to update a holiday group.

- Request URL: `/api/v1/developer/access_policies/holiday_groups/:id`
- Permission Key: `edit:policy`
- Method: `PUT`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | Example | |
| --- | --- | --- | --- | --- | --- |
| Id | T | String | Identity ID of the holiday group. | | |

## Request Body

| Parameter | Required | Type | Description | Example |
| --- | --- | --- | --- | --- |
| name | T | String | Name of the holiday group. | |
| description | F | String | Description of the holiday group. | |
| holidays | F | Array[Object] | Show a list of the holidays within the holiday group. **Omit this parameter if it doesn't need to be updated.** | |
| holidays[].id | F | String | Identity ID of the holiday.**Omit this parameter if it doesn't need to be updated.** | Get it from the API `/api/v1/developer/access_policies/holiday_groups` |
| holidays[].description | F | String | Description of the holiday. | |
| holidays[].name | F | String | Name of the holiday. | |
| holidays[].repeat | F | Boolean | Indicate whether the holiday repeats annually. | |
| holidays[].start_time | F | String | Start time of the holiday, provided in UTC format according to RFC3339. | 2023-08-25T00:00:00Z |
| holidays[].end_time | F | String | End time of the holiday, provided in UTC format according to RFC3339. | 2023-08-26T00:00:00Z |

## Response Body

| Parameter | Required | Type | Description | Related API | Purpose |
| --- | --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the holiday group. | `/api/v1/developer/access_policies/schedules` | Used to add a holiday group to the schedule. |

> Schemas: Schemas

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl -XPUT '{{host}}/api/v1/developer/access_policies/holiday_groups/7be7a7a0-818f-4f76-98c3-1c38957f4dca'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
```

```
    -H 'accept: application/json'
    -H 'content-type: application/json'   --data-raw
    '{
        "name": "Holiday Group-169286791557142",
        "holidays": [
            {
                "name": "Holiday Name 1",
                "description": "",
                "repeat": false,
                "start_time": "2023-08-25T00:00:00Z",
                "end_time": "2023-08-26T00:00:00Z"
            },# add a new holiday
            {
                "id":"d23a4226-765f-4967-b84f-6dfd53f33c89", # update an existing holiday
                "name": "Holiday Name 2",
                "description": "",
                "repeat": false,
                "start_time": "2023-08-26T00:00:00Z",
                "end_time": "2023-08-27T00:00:00Z"
            }
        ]
    }'
    --insecure
```

## Response Sample

```
{
    "code": "SUCCESS",
    "data": {
        "description": "",
        "holidays": [
            {
                "description": "",
                "end_time": "2023-08-26 00:00:00Z",
                "id": "8900533d-03be-4f84-832d-54ff59905759",
                "name": "Holiday Name 1",
                "repeat": false,
                "start_time": "2023-08-25 00:00:00Z"
            },
            {
                "description": "",
                "end_time": "2023-08-27 00:00:00Z",
                "id": "9fff81cc-d476-40c4-80f9-d510451ce2cd",
                "name": "Holiday Name 2",
                "repeat": false,
                "start_time": "2023-08-26 00:00:00Z"
            }
```

```
        ],
        "id": "7be7a7a0-818f-4f76-98c3-1c38957f4dca",
        "is_default": false,
        "name": "Holiday Group-169286791557142",
        "template_name": ""
    },
    "msg": "success"
}
```


# 5.10 Delete Holiday Group

This API allows you to delete a holiday group.

- Request URL: `/api/v1/developer/access_policies/holiday_groups/:id`
- Permission Key: `edit:policy`
- Method: `DELETE`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | Related API |
| --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the holiday group. | `/api/v1/developer/access_policies/holiday_groups` |

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl -XDELETE '{{host}}/api/v1/developer/access_policies/holiday_groups/7be7a7a0-818f-4f76-98c3-1c38957f4dca'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure
```

## Response Sample

```
{
    "code": "SUCCESS",
    "data": "success",
    "msg": "success"
}
```

# 5.11 Fetch Holiday Group

This API allows you to fetch a holiday group.

- Request URL: `/api/v1/developer/access_policies/holiday_groups/:id`
- Permission Key: `view:policy`
- Method: `GET`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Body

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Id | T | String | Identity ID of the holiday group. |

## Response Body

| Parameter | Required | Type | Description | Related API | Purpose |
| --- | --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the holiday group. | `/api/v1/developer/access_policies/schedules` | Used to add a holiday group to the schedule. |

> Schemas: Schemas

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl -XGET '{{host}}/api/v1/developer/access_policies/holiday_groups/7be7a7a0-818f-4f76-98c3-1c38957f4dca'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure
```

## Response Sample


```
{
    "code": "SUCCESS",
    "data": {
        "description": "",
        "holidays": [
            {
                "description": "",
                "end_time": "2023-08-26 00:00:00Z",
                "id": "8900533d-03be-4f84-832d-54ff59905759",
                "name": "Holiday Name 1",
                "repeat": false,
                "start_time": "2023-08-25 00:00:00Z"
            },
            {
                "description": "",
                "end_time": "2023-08-27 00:00:00Z",
                "id": "9fff81cc-d476-40c4-80f9-d510451ce2cd",
                "name": "Holiday Name 2",
                "repeat": false,
                "start_time": "2023-08-26 00:00:00Z"
            }
        ],
        "id": "7be7a7a0-818f-4f76-98c3-1c38957f4dca",
        "is_default": false,
        "name": "Holiday Group-169286791557142",
        "template_name": ""
    },
    "msg": "success"
}
```


# 5.12 Fetch All Holiday Groups

This API allows you to fetch the list of all holiday groups.

- Request URL: `/api/v1/developer/access_policies/holiday_groups`
- Permission Key: `view:policy`

- Method: `GET`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Body

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Id | T | String | Identity ID of the holiday group. |

## Response Body

| Parameter | Required | Type | Description | Related API | Purpose |
| --- | --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the holiday group. | `/api/v1/developer/access_policies/schedules` | Used to add a holiday group to the schedule. |
| name | T | String | Name of the holiday group. | | |
| description | T | String | Description of the holiday group. | | |
| count | T | Integer | Total number of holidays. | | |

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl -XGET '{{host}}/api/v1/developer/access_policies/holiday_groups'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure
```

## Response Sample

```
{
    "code": "SUCCESS",
    "data": [
        {
            "count": 0,
            "description": "",
            "id": "8cc22b49-a7f4-49a6-9f04-044444992d6c",
            "is_default": true,
            "name": "No Holidays"
        },
```

```
        {
            "count": 2,
            "description": "",
            "id": "86c634da-7b2c-411c-a2c1-1495d089c719",
            "is_default": false,
            "name": "Holiday Group-1692867312225"
        }
    ],
    "msg": "success"
}
```


## 5.13 Schedule Schemas

> These schemas are utilized for creating time periods for daily visits from Sunday through Saturday. The primary purpose of these schemas is to facilitate the assignment of access policies to users.

| Parameter | Type | Description |
| --- | --- | --- |
| id | String | Identity ID of the schedule. |
| name | String | Name of the schedule. |
| is_default | Boolean | Indicate whether the schedule is the system default. |
| type | String | Contains the `access` type, which is assigned to a user along with an access policy. |
| weekly | Object | The customizable scheduling strategy for each day from Sunday to Saturday. If not specified, it means access is allowed every day. |
| weekly.monday | Array[Object] | Specify the daily access time period from Sunday to Saturday. |
| weekly.monday[].start_time | String | Start time of the access time period. |
| weekly.monday[].end_time | String | End time of the access time period. |
| holiday_schedule | Array[Object] | Specify the accessible period during holidays. **UniFi Access Requirement:** `1.20.11 or later` |
| holiday_schedule[0].start_time | String | Start time of the access time period. |
| holiday_schedule[0].end_time | String | End time of the access time period. |
| holiday_group_id | String | Identity ID of the holiday group. |
| holiday_group.id | String | Identity ID of the holiday group. |
| holiday_group.name | String | Name of the holiday group. |
| holiday_group.is_default | Boolean | Indicate whether the holiday group is the system default. |
| holiday_group.description | String | Description of the holiday group. |
| holiday_group.holidays | Array[Object] | Show a list of the holidays within the holiday group. |
| holiday_group.holidays[].description | String | Description of the holiday. |
| holiday_group.holidays[].id | String | Identity ID of the holiday. |
| holiday_group.holidays[].name | String | Name of the holiday. |
| holiday_group.holidays[].repeat | Boolean | Indicate whether the holiday repeats annually. |
| holiday_group.holidays[].start_time | String | Start time of the holiday, provided in UTC format according to RFC3339. |
| holiday_group.holidays[].end_time | String | End time of the holiday, provided in UTC format according to RFC3339. |

# 5.14 Create Schedule

This API allows you to create a door access schedule.

- Request URL: `/api/v1/developer/access_policies/schedules`
- Permission Key: `edit:policy`

- Method: `POST`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Body

| Parameter | Required | Type | Description | Example |
| --- | --- | --- | --- | --- |
| name | T | String | The name of the schedule. This needs to be globally unique. | schedule-1789067565323 |
| week_schedule | T | Object | The customizable scheduling strategy for each day from Sunday to Saturday. If not specified, it means access is allowed every day. | |
| week_schedule.monday | T | Array[Object] | Specify the daily access time period from Sunday to Saturday. | |
| week_schedule.monday[0].start_time | T | String | Start time of the access time period. | 00:00:00 |
| week_schedule.monday[0].end_time | T | String | End time of the access time period. | 23:59:59 |
| holiday_group_id | F | String | Identity ID of the holiday group. The default is `No holidays` system group. | 75660081-431b-4dbe-9b98-e0257877118e |
| holiday_schedule | F | Array[Object] | Specify the accessible period during holidays. | |
| holiday_schedule[].start_time | F | String | Start time of the access time period. | 03:15:00 |
| holiday_schedule[].end_time | F | String | End time of the access time period. | 11:45:59 |

## Response Body

| Parameter | Required | Type | Description | Related API | Purpose |
| --- | --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the schedule. | `/api/v1/developer/access_policies` | Used to add a schedule to the access policy. |

> Schemas: Schemas

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl '{{host}}/api/v1/developer/access_policies/schedules'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'   --data-raw
'{
    "name": "schedule-1688977094169",
    "week_schedule": {
        "sunday": [],
        "monday": [
```

```
            {
                "start_time": "10:00:00",
                "end_time": "17:00:59"
            }
        ],
        "tuesday": [
            {
                "start_time": "10:00:00",
                "end_time": "17:00:59"
            }
        ],
        "wednesday": [
            {
                "start_time": "10:00:00",
                "end_time": "17:00:59"
            }
        ],
        "thursday": [],
        "friday": [
            {
                "start_time": "10:00:00",
                "end_time": "17:00:59"
            }
        ],
        "saturday": []
    },
    "holiday_group_id": "75660081-431b-4dbe-9b98-e0257877118e",
    "holiday_schedule": [
        {
            "start_time": "03:15:00",
            "end_time": "11:45:59"
        },
        {
            "start_time": "15:00:00",
            "end_time": "19:00:59"
        }
    ]
}'
```

## Response Sample

```
{
    "code": "SUCCESS",
    "data": {
        "id": "1d31b648-b8ff-4bd1-b742-60dbd70592cd",
        "is_default": false,
        "name": "schedule-1688977094169",
        "type": "access",
```

```
        "weekly": {
            "friday": [
                {
                    "end_time": "17:00:59",
                    "start_time": "10:00:00"
                }
            ],
            "monday": [
                {
                    "end_time": "17:00:59",
                    "start_time": "10:00:00"
                }
            ],
            "saturday": [],
            "sunday": [],
            "thursday": [],
            "tuesday": [
                {
                    "end_time": "17:00:59",
                    "start_time": "10:00:00"
                }
            ],
            "wednesday": [
                {
                    "end_time": "17:00:59",
                    "start_time": "10:00:00"
                }
            ]
        },
        "holiday_group_id": "75660081-431b-4dbe-9b98-e0257877118e",
        "holiday_group": {
            "description": "",
            "holidays": [
                {
                    "description": "",
                    "end_time": "2023-08-26 00:00:00Z",
                    "id": "d51777c4-9559-45aa-8e23-434995d9d2a1",
                    "is_template": false,
                    "name": "Holiday Name 1",
                    "repeat": false,
                    "start_time": "2023-08-25 00:00:00Z"
                },
                {
                    "description": "",
                    "end_time": "2023-08-27 00:00:00Z",
                    "id": "d23a4226-765f-4967-b84f-6dfd53f33c89",
                    "is_template": false,
                    "name": "Holiday Name 2",
                    "repeat": false,
```

```
                    "start_time": "2023-08-26 00:00:00Z"
                }
            ],
            "id": "75660081-431b-4dbe-9b98-e0257877118e",
            "is_default": false,
            "name": "Holiday Group-1692867915571423",
            "template_name": ""
        },
        "holiday_schedule": [
            {
                "start_time": "03:15:00",
                "end_time": "11:45:59"
            },
            {
                "start_time": "15:00:00",
                "end_time": "19:00:59"
            }
        ]
    },
    "msg": "success"
}
```


## 5.15 Update Schedule

This API allows you to update a door access schedule.

- Request URL: `/api/v1/developer/access_policies/schedules/:id`
- Permission Key: `edit:policy`
- Method: `PUT`

### Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

### Request Path

| Parameter | Required | Type | Description | Example | |
| --- | --- | --- | --- | --- | --- |
| Id | T | String | Identity ID of the schedule. | | |

# Request Body

| Parameter | Required | Type | Description | Example |
| --- | --- | --- | --- | --- |
| name | F | String | The name of the schedule. This needs to be globally unique. **Omit this parameter if it doesn't need to be updated.** | schedule-1789067565323 |
| week_schedule | F | Object | The customizable scheduling strategy for each day from Sunday to Saturday. If not specified, it means access is allowed every day. **Omit this parameter if it doesn't need to be updated.** | |
| week_schedule.monday | F | Array[Object] | Specify the daily access time period from Sunday to Saturday. | |
| week_schedule.monday[0].start_time | F | String | Start time of the access time period. | 00:00:00 |
| week_schedule.monday[0].end_time | F | String | End time of the access time period. | 23:59:59 |
| holiday_group_id | F | String | Identity ID of the holiday group. The default is `No holidays` system group. | 75660081-431b-4dbe-9b98-e0257877118e |
| holiday_schedule | F | Array[Object] | Specify the accessible period during holidays. | |
| holiday_schedule[].start_time | F | String | Start time of the access time period. | 03:15:00 |
| holiday_schedule[].end_time | F | String | End time of the access time period. | 11:45:59 |

# Response Body

| Parameter | Required | Type | Description | Related API | Purpose |
| --- | --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the schedule. | `/api/v1/developer/access_policies` | Used to add a schedule to the access policy. |

> Schemas: Schemas

# Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```shell
curl -XPUT '{{host}}/api/v1/developer/access_policies/schedules/1d31b648-b8ff-4bd1-b742-60dbd70592cd'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'   --data-raw
    '{
        "name": "schedule-1688977094169",
        "holiday_group_id": "75660081-431b-4dbe-9b98-e0257877118e",
        "week_schedule": {
            "sunday": [],
            "monday": [
                {
                    "start_time": "10:00:00",
                    "end_time": "17:00:59"
                }
            ],
```

```
            "tuesday": [
                {
                    "start_time": "10:00:00",
                    "end_time": "17:00:59"
                }
            ],
            "wednesday": [
                {
                    "start_time": "10:00:00",
                    "end_time": "17:00:59"
                }
            ],
            "thursday": [
                {
                    "start_time": "10:00:00",
                    "end_time": "17:01:59"
                }
            ],
            "friday": [
                {
                    "start_time": "10:00:00",
                    "end_time": "17:00:59"
                }
            ],
            "saturday": []
        },
        "holiday_schedule": [
            {
                "start_time": "03:15:00",
                "end_time": "11:45:59"
            }
        ]
    }'
    --insecure
```

## Response Sample

```
{
    "code": "SUCCESS",
    "data": {},
    "msg": "success"
}
```

# 5.16 Fetch Schedule

This API allows you to fetch a door access schedule.

- Request URL: `/api/v1/developer/access_policies/schedules/:id`
- Permission Key: `view:policy`
- Method: `GET`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | Example |
| --- | --- | --- | --- | --- |
| Id | T | String | Identity ID of the schedule. | |

## Response Body

| Parameter | Required | Type | Description | Example |
| --- | --- | --- | --- | --- |
| name | T | String | Name of the schedule. | |
| week_schedule | T | Object | The customizable scheduling strategy for each day from Sunday to Saturday. If not specified, it means access is allowed every day. | |
| holiday_group | F | Object | Show the assigned holiday group. | |
| holiday_schedule | F | Array[Object] | Show the accessible period during holidays. | |

> Schemas: Schemas

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
 curl --location '{{host}}/api/v1/developer/access_policies/schedules/908079e7-e26b-4951-9073-d775446d3584' --header 'Authorization: Bearer wHFmHR******kD6wHg'

```

## Response Sample

```
{
    "code": "SUCCESS",
    "data": {
        "id": "1d31b648-b8ff-4bd1-b742-60dbd70592cd",
        "is_default": false,
        "name": "schedule-1688977094169",
        "type": "access",
        "weekly": {
```

```
            "friday": [
                {
                    "end_time": "17:00:59",
                    "start_time": "10:00:00"
                }
            ],
            "monday": [
                {
                    "end_time": "17:00:59",
                    "start_time": "10:00:00"
                }
            ],
            "saturday": [],
            "sunday": [],
            "thursday": [
                {
                    "end_time": "17:01:59",
                    "start_time": "10:00:00"
                }
            ],
            "tuesday": [
                {
                    "end_time": "17:00:59",
                    "start_time": "10:00:00"
                }
            ],
            "wednesday": [
                {
                    "end_time": "17:00:59",
                    "start_time": "10:00:00"
                }
            ]
        },
        "holiday_group_id": "75660081-431b-4dbe-9b98-e0257877118e",
        "holiday_group": {
            "description": "",
            "holidays": [
                {
                    "description": "",
                    "end_time": "2023-08-26 00:00:00Z",
                    "id": "d51777c4-9559-45aa-8e23-434995d9d2a1",
                    "is_template": false,
                    "name": "Holiday Name 1",
                    "repeat": false,
                    "start_time": "2023-08-25 00:00:00Z"
                },
                {
                    "description": "",
                    "end_time": "2023-08-27 00:00:00Z",
```

```
                    "id": "d23a4226-765f-4967-b84f-6dfd53f33c89",
                    "is_template": false,
                    "name": "Holiday Name 2",
                    "repeat": false,
                    "start_time": "2023-08-26 00:00:00Z"
                }
            ],
            "id": "75660081-431b-4dbe-9b98-e0257877118e",
            "is_default": false,
            "name": "Holiday Group-16928679155714",
            "template_name": ""
        },
        "holiday_schedule": [
            {
                "end_time": "11:45:59",
                "start_time": "09:15:00"
            }
        ]
    },
    "msg": "success"
}
```


## 5.17 Fetch All Schedules

This API allows you to fetch all door access schedules.

- Request URL: `/api/v1/developer/access_policies/schedules`
- Permission Key: `view:policy`
- Method: `GET`

### Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

### Response Body

| Parameter | Required | Type | Description | Example |
| --- | --- | --- | --- | --- |
| Data | T | Array[Object] | | |

> Schemas: Schemas

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl --location '{{host}}/api/v1/developer/access_policies/schedules'
   -H 'Authorization: Bearer wHFmHR******kD6wHg'
   --insecure

```

## Response Sample


```
{
    "code": "SUCCESS",
    "data": [
        {
            "id": "73facd6c-839e-4521-a4f4-c07e1d44e748",
            "holiday_group_id": "75660081-431b-4dbe-9b98-e0257877118e",
            "is_default": true,
            "name": "Always Access",
            "status": 1,
            "type": "access"
        },
        {
            "id": "58c0f89b-f35c-4d2c-af7b-8b8918df2c31",
            "holiday_group_id": "75660081-431b-4dbe-9b98-e0257877118e",
            "is_default": false,
            "name": "UNVR Schedule",
            "status": 1,
            "type": "access"
        }
    ],
    "msg": "success"
}
```


# 5.18 Delete Schedule

This API allows you to delete a door access schedule.

- Request URL: `/api/v1/developer/access_policies/schedules/:id`
- Permission Key: `edit:policy`
- Method: `DELETE`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | Example | |
| --- | --- | --- | --- | --- | --- |
| Id | T | String | Identity ID of the schedule. | | |

## Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
 curl -XDELETE '{{host}}/api/v1/developer/access_policies/schedules/f5355214-3a45-4e0b-9245-12df7075df37'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure
```

## Response Sample

```
{"code":"SUCCESS","msg":"success","data":"success"}
```

# 6. Credential

> The APIs here are designed for managing PIN codes, NFC cards, and other related credentials.

## 6.1 Generate PIN Code

This API enables you to generate a PIN code. A PIN code can be assigned to a visitor or user, and once assigned, they can use it to unlock doors.

- Request URL: `/api/v1/developer/credentials/pin_codes`
- Permission Key: `view:credential`
- Method: `POST`

### Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

### Response Body

| Parameter | Required | Type | Description | Related API | Purpose |
| --- | --- | --- | --- | --- | --- |
| data | T | String | PIN code | `/api/v1/developer/users;` `/api/v1/developer/visitors` | Assign the PIN code to the created user or visitor. |

### Request Sample: Shell/cURL

The request body should be a JSON object containing the following fields:

```
curl --location --request POST
'https://{{host}}/api/v1/developer/credentials/pin_codes'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'Content-Type: application/json' \
    --data ''
    --insecure
```

### Response Sample

```
{
    "code": "SUCCESS",
    "data": "67203419",
    "msg": "success"
}
```

# 6.2 Enroll NFC Card

Wake up a UA reader and create a session to enroll an NFC card.

- Request URL: `/api/v1/developer/credentials/nfc_cards/sessions`
- Permission Key: `edit:credential`
- Method: `POST`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Body

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| device_id | T | String | Identity ID of the device. | Get it from the API `/api/v1/developer/devices`. |
| reset_ua_card | F | Boolean | | This option allows you to reset an NFC card already enrolled at another site. |

## Response Body

| Parameter | Required | Type | Description | Related API | Purpose |
| --- | --- | --- | --- | --- | --- |
| session_id | T | String | The session for enrolling an NFC card. | **#6.3** `/api/v1/developer/credentials/nfc_cards/sessions/{session_id}` | NFC card polling result. |

## Request Sample

```
curl '{{host}}/api/v1/developer/credentials/nfc_cards/sessions'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data-raw '{"device_id": "0418d6a2bb7a", "reset_ua_card": false}'
    --insecure
```

## Response Sample

```
{
    "code": "SUCCESS",
    "msg": "success",
    "data": {
        "session_id": "e8a97c52-6676-4c48-8589-bd518afc4094"
    }
}
```


# 6.3 Fetch NFC Card Enrollment Status

This API enables you to poll a UA reader to fetch the NFC card enrollment status and the generated card tokens.

- Request URL: `/api/v1/developer/credentials/nfc_cards/sessions/:id`
- Permission Key: `edit:credential`
- Method: `GET`

## Request Header

| Parameter | Required | Type | Description |
|---|---|---|---|
| Authorization | T | String | Token required for authentication and access control. |

## Request Body

| Parameter | Required | Type | Description | Related API | How to Get It? |
|---|---|---|---|---|---|
| id | T | String | The ID of the session **#6.2**. | `/api/v1/developer/users`; `/api/v1/developer/visitors` | Get session id from the API **#6.2** |

## Response Body

| Parameter | Required | Type | Description | Related API | Purpose |
|---|---|---|---|---|---|
| token | T | String | Unique NFC card token. | `/api/v1/developer/users`; `/api/v1/developer/visitors` | The generated card token is used to bind to a user or visitor. |
| card_id | T | String | Display ID of the NFC card. | | |

## Request Sample

```
GET /api/v1/developer/credentials/nfc_cards/sessions/e8a97c52-6676-4c48-8589-bd518afc4094
```

## Response Sample

```
{
    "code": "SUCCESS",
    "msg": "success",
    "data": {
        "card_id": "014A3151",
        "token": "821f90b262e90c5c0fbcddf3d6d2f3b94cc015d6e8104ab4fb96e4c8b8e90cb7"
    }
}
```

# 6.4 Remove a Session Created for NFC Card Enrollment

This API enables you to remove a session created for enrolling an NFC card.

- Request URL: `/api/v1/developer/credentials/nfc_cards/sessions/:id`
- Permission Key: `edit:credential`
- Method: `DELETE`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Body

| Parameter | Required | Type | Description | Related API | How to Get It? |
| --- | --- | --- | --- | --- | --- |
| id | T | String | The ID of the session **#6.2**. | `/api/v1/developer/users;` `/api/v1/developer/visitors` | Get session id from the API **#6.2** |

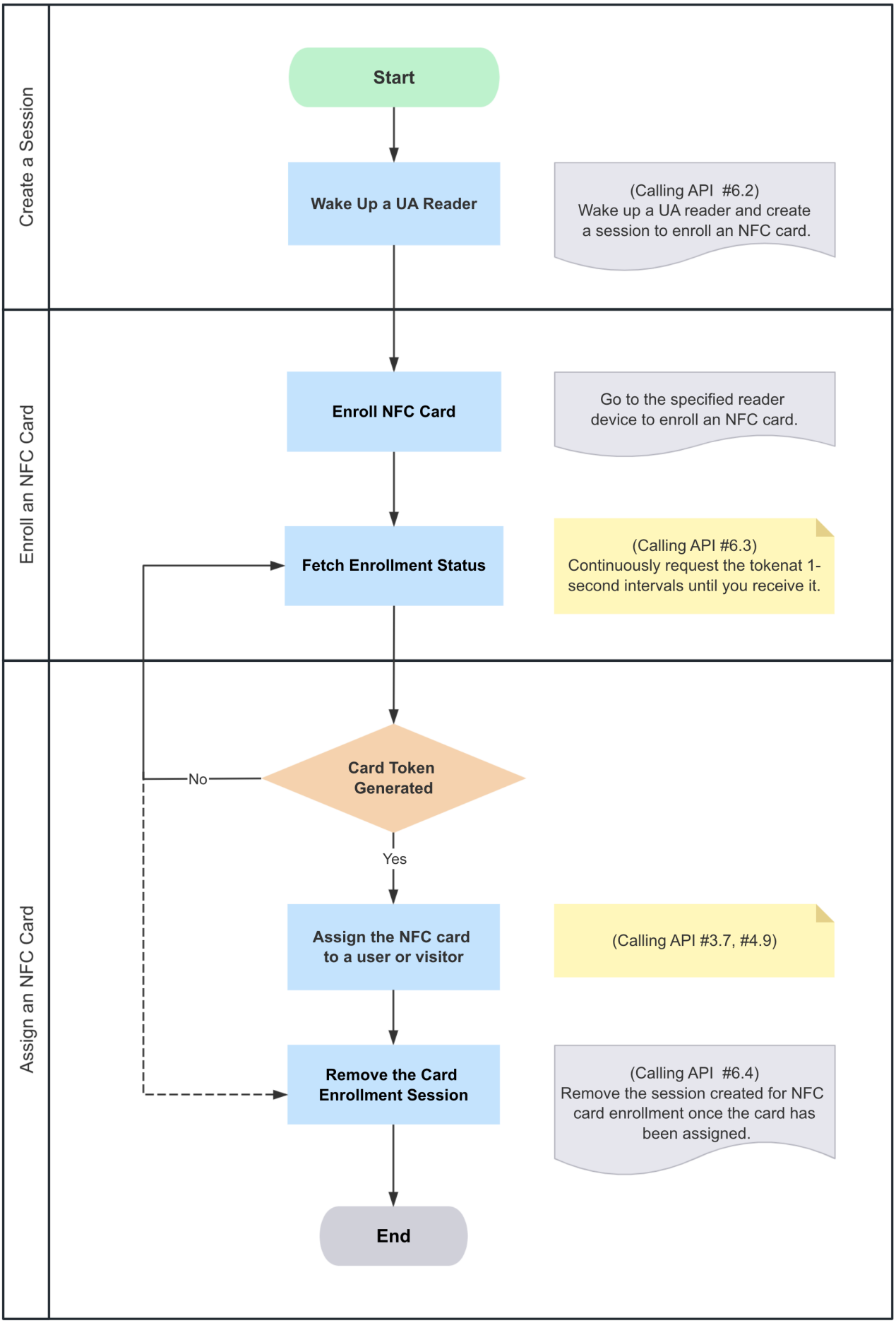
## Request Sample

```
DELETE /api/v1/developer/credentials/nfc_cards/sessions/e8a97c52-6676-4c48-8589-bd518afc4094
```

## Response Sample

```
{"code":"SUCCESS","msg":"success","data":"success"}
```

## 6.5 Flowchart for NFC Card Enrollment


Create a Session
Start
Wake Up a UA Reader
(Calling API #6.2)
Wake up a UA reader and create a session to enroll an NFC card.
Enroll an NFC Card
Enroll NFC Card
Go to the specified reader device to enroll an NFC card.
Fetch Enrollment Status
(Calling API #6.3)
Continuously request the tokenat 1-second intervals until you receive it.
Assign an NFC Card
Card Token Generated
No
Yes
Assign the NFC card to a user or visitor
(Calling API #3.7, #4.9)
Remove the Card Enrollment Session
(Calling API #6.4)
Remove the session created for NFC card enrollment once the card has been assigned.
End

# 6.6 NFC Card Schemas

| Parameter | Type | Description |
| --- | --- | --- |
| token | String | Identity token of the NFC card. |
| display_id | String | Display ID of the NFC card. |
| status | String | Status of the NFC card. `enum status {assigned,pending,disable,deleted,loss}` |
| alias | String | Preferred name of the NFC card. |
| card_type | String | Type of the NFC card. |
| user_id | String | Owner ID of the NFC card. |
| user_type | String | Type of the owner. `enum user_type {USER,VISITOR}` |
| user | Object | Owner of the NFC card. |
| user.id | String | Identity ID of the user. |
| user.first_name | String | First name of the user. |
| user.last_name | String | Last name of the user. |
| user.name | String | Full name of the user. |

# 6.7 Fetch NFC Card

This API allows you to fetch NFC card details.

- Request URL: `/api/v1/developer/credentials/nfc_cards/tokens/:token`
- Permission Key: `view:credential`
- Method: `GET`
- UniFi Access Requirement: `Version 1.22.16 or later`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | Example | How to Get It? |
| --- | --- | --- | --- | --- | --- |
| token | T | String | Token of the NFC card. | 821f90b262e90c5c0fbcddf3d6d2f3b94cc015d6e8104ab4fb96e4c8b8e90cb7 | Get it from the API **#6.3 #6.8** |

## Response Body

Schemas: NFC Card Schemas

## Response Sample


```
{
    "code": "SUCCESS",
    "data": {
        "alias": "",
        "card_type": "ua_card",
        "display_id": "100005",
        "note": "100005",
        "status": "assigned",
        "token":
"f77d69b08eaf5eb5d647ac1a0a73580f1b27494b345f40f54fa022a8741fa15c",
        "user": {
            "avatar": "",
            "first_name": "H",
            "id": "e0051e08-c4d5-43db-87c8-a9b19cb66513",
            "last_name": "L",
            "name": "H L"
        },
        "user_id": "e0051e08-c4d5-43db-87c8-a9b19cb66513",
        "user_type": "USER"
    },
    "msg": "success"
}
```


## Request Sample

The request body should be a JSON object containing the following fields:

```
curl -XGET
'{{host}}/api/v1/developer/credentials/nfc_cards/tokens/f77d69b08eaf5eb5d647ac1a0a73
580f1b27494b345f40f54fa022a8741fa15c'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure
```

# 6.8 Fetch All NFC Cards

This API allows you to fetch all NFC cards.

- Request URL: `/api/v1/developer/credentials/nfc_cards/tokens`

- Permission Key: `view:credential`
- Method: `GET`
- UniFi Access Requirement: `Version 1.22.16 or later`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Query Parameters

| Parameter | Required | Type | Description | Example |
| --- | --- | --- | --- | --- |
| page_num | F | String | Current page number in the pagination. | 1 |
| page_size | F | String | Number of users per page. | 25 |

## Response Body

Schemas: NFC Card Schemas

## Response Sample


```
{
    "code": "SUCCESS",
    "data": [
        {
            "alias": "",
            "card_type": "ua_card",
            "display_id": "100004",
            "note": "100004",
            "status": "assigned",
            "token":
"9e24cdfafebf63e58fd02c5f67732b478948e5793d31124239597d9a86b30dc4",
            "user": {
                "avatar": "",
                "first_name": "H",
                "id": "e0051e08-c4d5-43db-87c8-a9b19cb66513",
                "last_name": "L",
                "name": "H L"
            },
            "user_id": "e0051e08-c4d5-43db-87c8-a9b19cb66513",
            "user_type": "USER"
        },
        {
            "alias": "F77D69B03",
```

```
            "card_type": "ua_card",
            "display_id": "100005",
            "note": "100005",
            "status": "assigned",
            "token":
"f77d69b08eaf5eb5d647ac1a0a73580f1b27494b345f40f54fa022a8741fa15c",
            "user": {
                "avatar": "",
                "first_name": "H2",
                "id": "34dc90a7-409f-4bf8-a5a8-1c59535a21b9",
                "last_name": "L",
                "name": "H2 L"
            },
            "user_id": "34dc90a7-409f-4bf8-a5a8-1c59535a21b9",
            "user_type": "VISITOR"
        }
    ],
    "msg": "succ",
    "pagination": {
        "page_num": 1,
        "page_size": 2,
        "total": 2
    }
}
```


## Request Sample

The request body should be a JSON object containing the following fields:

```
curl -XGET '{{host}}/api/v1/developer/credentials/nfc_cards/tokens?
page_num=1&page_size=12'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure
```

# 6.9 Delete NFC Card

This API allows you to delete an NFC card.

- Request URL: `/api/v1/developer/credentials/nfc_cards/tokens/:token`
- Permission Key: `edit:credential`
- Method: `DELETE`
- UniFi Access Requirement: `Version 1.22.16 or later`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | Example | How to Get It? |
| --- | --- | --- | --- | --- | --- |
| token | T | String | Token of the NFC card. | 821f90b262e90c5c0fbcddf3d6d2f3b94cc015d6e8104ab4fb96e4c8b8e90cb7 | Get it from the API **#6.3 #6.8** |

## Response Sample

```
{
    "code": "SUCCESS",
    "data": "success",
    "msg": "success"
}
```

## Request Sample

The request body should be a JSON object containing the following fields:

```
curl -XDELETE
'{{host}}/api/v1/developer/credentials/nfc_cards/tokens/f77d69b08eaf5eb5d647ac1a0a73
580f1b27494b345f40f54fa022a8741fa15c'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure
```

# 6.10 Update NFC Card

This API allows you to update an NFC card.

- Request URL: `/api/v1/developer/credentials/nfc_cards/tokens/:token`
- Permission Key: `edit:credential`
- Method: `PUT`
- UniFi Access Requirement: `Version 3.1.30 or later`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | Example | How to Get It? |
| --- | --- | --- | --- | --- | --- |
| token | T | String | Token of the NFC card. | 821f90b262e90c5c0fbcddf3d6d2f3b94cc015d6e8104ab4fb96e4c8b8e90cb7 | Get it from the API **#6.3 #6.8** |

## Request Body

| Parameter | Required | Type | Description | Related API | How to Get It? |
| --- | --- | --- | --- | --- | --- |
| alias | F | String | Alias of the NFC card. | | |

## Response Sample

```
{
    "code": "SUCCESS",
    "data": "success",
    "msg": "success"
}
```

## Request Sample

The request body should be a JSON object containing the following fields:

```
curl -XPUT
'{{host}}/api/v1/developer/credentials/nfc_cards/tokens/f77d69b08eaf5eb5d647ac1a0a73
580f1b27494b345f40f54fa022a8741fa15c'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data '{
        "alias": "New Alias"
    }'
    --insecure
```

# 7. Space

> The APIs here are designed for managing spaces, including doors, door groups, and floors.

## 7.1 Fetch Door Group Topology

Fetch all current floor and door resources for access policy and visitor assignment purposes.

- Request URL: `/api/v1/developer/door_groups/topology`
- Permission Key: `view:space`
- Method: `GET`

### Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

### Response Body

| Parameter | Required | Type | Description | Purpose | Related API |
| --- | --- | --- | --- | --- | --- |
| id | F | String | Identity ID of the door group. | Used when creating an access policy. | |
| type | T | String | `enum group_type {building,access}` The `building` type contains all the doors; the `access` type represents all the customized door groups. | | `access: /api/v1/developer/door_groups` |
| resource_topologies | T | Array[Object] | Contain information about the floor and all its associated doors. | Bind policies or specify available locations for visitors to access. | `/api/v1/developer/access_policies` `/api/v1/developer/visitors` |
| resource_topologies[].id | T | String | Identity ID of the floor. | | |
| resource_topologies[].type | T | String | Type of the floor. | | |
| resource_topologies[].name | T | String | Name of the floor. | | |
| resource_topologies[].resources | T | Array[Object] | Contain all the doors on the floor. | | |
| resource_topologies[].is_bind_hub | F | Boolean | Indicate whether the door has bound to a hub device. It can only be used for remote opening if it's bound. | | |
| resource_topologies[].resources.id | T | String | Identity ID of the door. | | |
| resource_topologies[].resources.name | T | String | Name of the door. | | |
| resource_topologies[].resources.type | T | String | Type of the door. | | |

# Request Sample

The request body should be a JSON object containing the following fields:

```
curl -X GET '{{host}}/api/v1/developer/door_groups/topology'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure
```

# Response Sample

```
{
    "code": "SUCCESS",
    "data": [
        {
            "id": "d5573467-d6b3-4e8f-8e48-8a322b91664a",
            "name": "All Doors",
```

```
            "resource_topologies": [
                {
                    "id": "9bee6e0e-108d-4c52-9107-76f2c7dea4f1",
                    "name": "Main Floor",
                    "resources": [
                        {
                            "id": "6ff875d2-af87-470b-9cb5-774c6596afc8",
                            "name": "Door 3855",
                            "type": "door",
                            "is_bind_hub": true
                        }
                    ],
                    "type": "floor"
                }
            ],
            "type": "building"
        },
        {
            "id": "5c496423-6d25-4e4f-8cdf-95ad5135188a",
            "name": "customized group",
            "resource_topologies": [
                {
                    "id": "9bee6e0e-108d-4c52-9107-76f2c7dea4f1",
                    "name": "Main Floor",
                    "resources": [
                        {
                            "id": "6ff875d2-af87-470b-9cb5-774c6596afc8",
                            "name": "Door 3855",
                            "type": "door",
                            "is_bind_hub": true
                        }
                    ],
                    "type": "floor"
                }
            ],
            "type": "access"
        }
    ],
    "msg": "success"
}
```


## 7.2 Create Door Group

This API allows you to create a door group.

- Request URL: `/api/v1/developer/door_groups`
- Permission Key: `edit:space`

- Method: `POST`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Body

| Parameter | Required | Type | Description | Example | How to Get It? |
| --- | --- | --- | --- | --- | --- |
| group_name | T | String | Name of the door group. This needs to be globally unique. | | |
| resources | T | Array[String] | Collection of door ID. | | `/api/v1/developer/door_groups/topology` |

## Response Body

| Parameter | Required | Type | Description | Purpose | Related API |
| --- | --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the door group. | Used to assign access group to visitor and access policy. | `/api/v1/developer/visitors;` `/api/v1/developer/access_policies;` |
| name | T | String | Name of the door group. | | |
| resources | T | String | All doors contained under the group. | | |
| resources[].id | T | String | Identity ID of the door. | | |
| resources[].type | T | String | Type of the door. | | |

## Request Sample

The request body should be a JSON object containing the following fields:

```
curl '{{host}}/api/v1/developer/door_groups'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data-raw '{
        "group_name": "Test",
        "resources": [
            "6ff875d2-af87-470b-9cb5-774c6596afc8"
        ]
    }'
    --insecure
```

## Response Sample

```
{
    "code": "SUCCESS",
    "data": {
        "id": "0140fa3d-8973-4305-a0ce-5306ae277878",
        "name": "Customized Door Group",
        "resources": [
            {
                "id": "6ff875d2-af87-470b-9cb5-774c6596afc8",
                "type": "door"
            }
        ],
        "type": "access"
    },
    "msg": "success"
}
```

# 7.3 Fetch Door Group

This API allows you to fetch door group details.

- Request URL: `/api/v1/developer/door_groups/:id`
- Permission Key: `view:space`
- Method: `GET`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Id | T | String | Identity ID of the door group. |

## Response Body

| Parameter | Required | Type | Description | Purpose | Related API |
| --- | --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the door group. | Used to assign access group to visitor and access policy. | `/api/v1/developer/visitors;` `/api/v1/developer/access_policies;` |
| name | T | String | Name of the door group. | | |
| type | t | String | Include door access and building resources. `enum type {access,building}` The `building` type stands for all doors, which is a special group obtained from the topology API. The `access` type represents all custom door groups. | | |
| resources | T | String | All doors contained under the group. | | |
| resources[].id | T | String | Identity ID of the door. | | |
| resources[].type | T | String | The access type represents all custom door group types. | | |
| resources[].name | T | String | Name of the door. | | |

# Request Sample

The request body should be a JSON object containing the following fields:

```
curl '{{host}}/api/v1/developer/door_groups/d5573467-d6b3-4e8f-8e48-8a322b91664a'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure
```

# Response Sample

```
# Group type is building
{
    "code": "SUCCESS",
    "data": {
        "id": "d5573467-d6b3-4e8f-8e48-8a322b91664a",
        "name": "All Doors",
        "resources": [
            {
                "id": "6ff875d2-af87-470b-9cb5-774c6596afc8",
                "name": "Door 3855",
                "type": "door"
            },
            {
                "id": "7cc1823f-9cdb-447b-b01b-4cb2abc661c0",
                "name": "A2 Door",
                "type": "door"
            },
```

```
            {
                "id": "ececa68e-239f-4b82-adc4-0c9ce70c60ff",
                "name": "A3",
                "type": "door"
            }
        ],
        "type": "building"
    },
    "msg": "success"
}

# Customized groups
{
    "code": "SUCCESS",
    "data": {
        "id": "1be0c995-0347-4cb2-93b3-66a9624af568",
        "name": "Door Group 01",
        "resources": [
            {
                "id": "6ff875d2-af87-470b-9cb5-774c6596afc8",
                "type": "door",
                "name": "Door 385"
            }
        ],
        "type": "access"
    },
    "msg": "success"
}
```

# 7.4 Update Door Group

This API allows you to update door group details.

- Request URL: `/api/v1/developer/door_groups/:id`
- Permission Key: `edit:space`
- Method: `PUT`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| id | T | String | Identity ID of the door group. |

## Request Body

| Parameter | Required | Type | Description | Example | How to Get It? |
| --- | --- | --- | --- | --- | --- |
| group_name | F | String | Name of door group. **Omit this parameter if it doesn't need to be updated.** | | |
| resources | F | Array[String] | Collection of door identifier ID. **Omit this parameter if it doesn't need to be updated.** | | `/api/v1/developer/door_groups/topology` |

## Response Body

| Parameter | Required | Type | Description | Purpose | Related API |
| --- | --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the door group. | Used to assign access group to visitor and access policy. | `/api/v1/developer/visitors;` `/api/v1/developer/access_policies;` |
| name | T | String | Name of the door group. | | |
| resources | T | String | All doors contained under the group. | | |
| resources[].id | T | String | Identity ID of the door. | | |
| resources[].type | T | String | Type of the door. | | |

## Request Sample

The request body should be a JSON object containing the following fields:

```
curl -X PUT'{{host}}/api/v1/developer/door_groups/0140fa3d-8973-4305-a0ce-5306ae277878'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data-raw '{
        "resources": [
            "6ff875d2-af87-470b-9cb5-774c6596afc8",
            "5a2c3d4e-1f6b-4c8d-9e0f-2a3b4c5d6e7f",
            "2p3q4r5s-6t7u-8v9w-x0y1-z2a3b4c5d6e"
        ]
    }'
    --insecure

# Delete all door resources
```

```
curl -X PUT'{{host}}/api/v1/developer/door_groups/0140fa3d-8973-4305-a0ce-5306ae277878'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data-raw '{
        "resources": []
    }'
    --insecure
```

## Response Sample

```
{
    "code": "SUCCESS",
    "data": {
        "id": "0140fa3d-8973-4305-a0ce-5306ae277878",
        "name": "test",
        "resources": [
            {
                "id": "6ff875d2-af87-470b-9cb5-774c6596afc8",
                "type": "door"
            }
        ],
        "type": "access"
    },
    "msg": "success"
}
```

# 7.5 Fetch All Door Groups

This API allows you to fetch the list of all door groups.

- Request URL: `/api/v1/developer/door_groups`
- Permission Key: `view:space`
- Method: `GET`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Response Body

| Parameter | Required | Type | Description | Purpose | Related API |
| --- | --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the door group. | Used to create a door groups and to open doors remotely, if a device is bound. | `/api/v1/developer/door_groups;` `/api/v1/developer/doors/:id/remote_unlock;` |
| name | T | String | Name of the door group. | | |
| type | t | String | Include door access and building resources. `enum type {access,building}` The `building` type represents all doors, which is a special group obtained from the topology API. But the list does not contain the group type `building`. The `access` type represents all custom door groups. | | |
| resources | T | String | All doors contained under the group. | | |
| resources[].id | T | String | Identity ID of the door. | | |
| resources[].type | T | String | Type of the door. | | |

## Response Sample

The request body should be a JSON object containing the following fields:

```
curl '{{host}}/api/v1/developer/door_groups'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure
```

## Response Sample


```
{
    "code": "SUCCESS",
    "data": [
        {
            "id": "5c496423-6d25-4e4f-8cdf-95ad5135188a",
            "name": "Test",
            "resources": [
                {
                    "id": "6ff875d2-af87-470b-9cb5-774c6596afc8",
                    "type": "door"
                }
            ],
            "type": "access"
```

```
        },
        {
            "id": "1907cc46-0a73-4077-94c1-95b625bdb0f8",
            "name": "Test2",
            "resources": [
                {
                    "id": "6ff875d2-af87-470b-9cb5-774c6596afc8",
                    "type": "door"
                }
            ],
            "type": "access"
        }
    ],
    "msg": "success"
}
```


# 7.6 Delete Door Group

This API allows you to delete a door group.

- Request URL: `/api/v1/developer/door_groups/:id`
- Permission Key: `edit:space`
- Method: `DELETE`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| id | T | String | Identity ID of the door group. |

## Request Sample

The request body should be a JSON object containing the following fields:

```
curl -X DELETE'{{host}}/api/v1/developer/door_groups/0140fa3d-8973-4305-a0ce-5306ae277878'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure
```

## Response Sample

```
{
    "code": "SUCCESS",
    "data": "success",
    "msg": "success"
}
```

# 7.7 Fetch Door

This API allows you to fetch door details.

- Request URL: `/api/v1/developer/doors/:id`
- Permission Key: `view:space`
- Method: `GET`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| id | T | String | Identity ID of the door. |

## Response Body

| Parameter | Required | Type | Description | Purpose | Related API |
| --- | --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the door group. | 1. Used to create a door group. 2. Used to unlock a door remotely, if the door is bound to a hub device. | `/api/v1/developer/door_groups;` `/api/v1/developer/doors/:id/remote_unlock;` |
| name | T | String | Name of the door. | | |
| full_name | T | String | Full name of the door. | | |
| floor_id | T | String | Identity ID of the floor. | | |
| type | T | String | Type of the door. | | |
| is_bind_hub | T | String | Indicate whether the door has bound to a hub device. It can only be used for remote opening if it's bound. | | |
| door_lock_relay_status | T | String | Door lock status. `enum door_lock_relay_status {lock,unlock}` | | |
| door_position_status | T | String | DPS: Door position status, including both `open` and `close`. A null value means that no device is connected. | | |

# Request Sample

The request body should be a JSON object containing the following fields:

```
curl -X GET '{{host}}/api/v1/developer/doors/0ed545f8-2fcd-4839-9021-b39e707f6aa9'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure
```

# Response Sample

```
{
    "code": "SUCCESS",
    "data": {
        "door_lock_relay_status": "lock",
        "door_position_status": "",
        "floor_id": "3275af8d-3fa7-4902-a11b-011e41c8464a",
        "full_name": "UNVR - 1F - Main Door",
        "id": "0ed545f8-2fcd-4839-9021-b39e707f6aa9",
        "is_bind_hub": true,
        "name": "Main Door",
        "type": "door"
    },
    "msg": "success"
}
```

# 7.8 Fetch All Doors

This API allows you to fetch the list of all doors.

- Request URL: `/api/v1/developer/doors`
- Permission Key: `view:space`
- Method: `GET`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Response Body

| Parameter | Required | Type | Description | Purpose | Related API |
| --- | --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the door group. | 1. Used to create a door group. 2. Used to unlock a door remotely, if the door is bound to a hub device. | `/api/v1/developer/door_groups;` `/api/v1/developer/doors/:id/remote_unlock;` |
| name | T | String | Name of the door. | | |
| full_name | T | String | Full name of the door. | | |
| floor_id | T | String | Identity ID of the floor. | | |
| type | T | String | Type of the door. | | |
| is_bind_hub | T | String | Indicate whether the door has bound to a hub device. It can only be used for remote opening if it's bound. | | |
| door_lock_relay_status | T | String | Door lock status. `enum door_lock_relay_status {lock,unlock}` | | |
| door_position_status | T | String | DPS: Door position status, including both `open` and `close`. A null value means that no device is connected. | | |

## Request Sample

The request body should be a JSON object containing the following fields:

```
curl -X GET '{{host}}/api/v1/developer/doors'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure
```

## Response Sample


```
{
    "code": "SUCCESS",
    "data": [
        {
            "door_lock_relay_status": "unlock",
            "door_position_status": "open",
            "floor_id": "23c5db06-b59b-494d-94f1-23e88fbe4909",
            "full_name": "UNVR - 2F - A2 Door",
            "id": "0ed545f8-2fcd-4839-9021-b39e707f6aa9",
            "is_bind_hub": true,
            "name": "A2 Door",
            "type": "door"
        },
        {
            "door_lock_relay_status": "lock",
            "door_position_status": "close",
            "floor_id": "7c62b4b3-692f-44ea-8eb8-e212833b4e0f",
            "full_name": "UNVR - 1F - Door 3855",
            "id": "5785e97b-6123-4596-ba49-b6e51164db9b",
            "is_bind_hub": true,
            "name": "Door 3855",
            "type": "door"
        }
    ],
    "msg": "success"
}
```


# 7.9 Remote Door Unlocking

This API allows you to remotely unlock a door.

- Request URL: `/api/v1/developer/doors/:id/unlock`
- Permission Key: `edit:space`
- Method: `PUT`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| id | T | String | Identity ID of the door. |

## Request Sample

The request body should be a JSON object containing the following fields:

```
curl -X PUT '{{host}}/api/v1/developer/doors/5785e97b-6123-4596-ba49-b6e51164db9b/unlock'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure
```

## Response Sample

```
{
    "code": "SUCCESS",
    "data": "success",
    "msg": "success"
}
```

# 7.10 Set Temporary Door Locking Rule

This API allows you to temporarily set the locking rules for doors.

- Request URL: `/api/v1/developer/doors/:id/lock_rule`
- Permission Key: `edit:space`
- Method: `PUT`
- UniFi Access Requirement: `1.24.6 or later`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| id | T | String | Identity ID of the door. |

## Request Body

| Parameter | Required | Type | Description | Example |
| --- | --- | --- | --- | --- |
| type | T | String | `enum type {keep_lock,keep_unlock,custom,reset,lock_early}` `keep_lock` is used to set the door to the "keep locked" state, while `keep_unlock` is used to set it to the "keep unlocked" state. `custom` allows customization of the unlock time duration, and `reset` is used to restore the door to its initial state (not applicable to the "lock_early" state). **NOTE:** If the door is currently on an unlock schedule ( `schedule` ), you can use `lock_early` to lock the door early. | |
| interval | F | Integer | Set the lock time duration (**minutes**) when type is **custom**. | 10 |

## Request Sample

The request body should be a JSON object containing the following fields:

```
# Customized 10-minute unlocked
curl -X PUT '{{host}}/api/v1/developer/doors/e4978b83-203d-4015-97df-b86efc91cb0c/lock_rule'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data '{
        "type": "custom",
        "interval": 10
    }'
    --insecure

# Keep it unlocked
curl -X PUT '{{host}}/api/v1/developer/doors/e4978b83-203d-4015-97df-b86efc91cb0c/lock_rule'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data '{
        "type": "keep_unlock"
    }'
    --insecure
```

```
# Keep it locked
curl -X PUT '{{host}}/api/v1/developer/doors/e4978b83-203d-4015-97df-b86efc91cb0c/lock_rule'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data '{
        "type": "keep_lock"
    }'
    --insecure

# Use reset to restore the temporary unlock schedule (e.g., “Unlock for 1 Hour”) to its original locked state. This parameter is intended solely for resetting a temporary unlock schedule. If you wish to lock a door before its scheduled unlock time ends, use lock_early below instead.
curl -X PUT '{{host}}/api/v1/developer/doors/e4978b83-203d-4015-97df-b86efc91cb0c/lock_rule'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data '{
        "type": "reset"
    }'
    --insecure

# If the door is currently on an unlock schedule, you can use lock_early to lock the door before the scheduled time ends. For instance, if the unlock schedule is set from 9:00 AM to 6:00 PM, you can use this parameter at 3:00 PM to end the schedule and lock the door earlier.
curl -X PUT '{{host}}/api/v1/developer/doors/e4978b83-203d-4015-97df-b86efc91cb0c/lock_rule'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data '{
        "type": "lock_early"
    }'
    --insecure
```

## Response Sample

```
{
    "code": "SUCCESS",
    "data": "success",
    "msg": "success"
}
```

# 7.11 Fetch Door Locking Rule

This API allows you to fetch the locking rules for doors.

- Request URL: `/api/v1/developer/doors/:id/lock_rule`
- Permission Key: `view:space`
- Method: `GET`
- UniFi Access Requirement: `1.24.6 or later`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| id | T | String | Identity ID of the door. |

## Response Body

| Parameter | Required | Type | Description | Example |
| --- | --- | --- | --- | --- |
| type | T | String | `enum type {schedule,keep_lock,keep_unlock,custom,lock_early}` `keep_lock` is used to set the "keep locked" state, while `keep_unlock` is used to set the "keep unlocked" state. `custom` is used to customize the unlock time duration. `schedule` indicates that the door is currently in the unlock schedule state. The `lock_early` is used to terminate doors in an unlock schedule early. | |
| ended_time | T | Integer | End time of the set rule. | 1708672257 |

## Request Sample

The request body should be a JSON object containing the following fields:

```
curl -X '{{host}}/api/v1/developer/doors/e4978b83-203d-4015-97df-b86efc91cb0c/lock_rule'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure
```

## Response Sample

```
# Keep it locked
{
    "code": "SUCCESS",
    "data": {
        "ended_time": 3602128309,
        "type": "keep_lock"
    },
    "msg": "success"
}

# Keep it unlocked
{
    "code": "SUCCESS",
    "data": {
        "ended_time": 3602128562,
        "type": "keep_unlock"
    },
    "msg": "success"
}

# Customized unlock duration
{
    "code": "SUCCESS",
    "data": {
        "ended_time": 1708673342,
        "type": "custom"
    },
    "msg": "success"
}

#The `lock_early` is used to terminate doors in an unlock schedule early.
{
    "code": "SUCCESS",
    "data": {
          "type": "lock_early",
           "ended_time": 1708673342,
    },
    "msg": "success"
}
```

# 7.12 Set Door Emergency Status

This API allows you to set the emergency status for all doors.

- Request URL: `/api/v1/developer/doors/settings/emergency`

- Permission Key: `edit:space`
- Method: `PUT`
- UniFi Access Requirement: `1.24.6 or later`

## Request Header

| Parameter | Required | Type | Description |
|---|---|---|---|
| Authorization | T | String | Token required for authentication and access control. |

## Request Body

| Parameter | Required | Type | Description | Example |
|---|---|---|---|---|
| lockdown | F | Boolean | `True` will keep the door locked. | true |
| evacuation | F | Boolean | `True` will keep the door unlocked. | false |

## Request Sample

The request body should be a JSON object containing the following fields:

```
# Keep it locked
curl -X PUT '{{host}}/api/v1/developer/doors/settings/emergency'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data '{
       "lockdown": true,
       "evacuation": false
    }'
    --insecure

# Keep it unlocked
curl -X PUT '{{host}}/api/v1/developer/doors/settings/emergency'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data '{
       "lockdown": false,
       "evacuation": true
    }'
    --insecure

# Restore the initial state or release the setting
curl -X PUT '{{host}}/api/v1/developer/doors/settings/emergency'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
```

```
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data '{
       "lockdown": false,
       "evacuation": false
    }'
    --insecure
```

## Response Sample

```
{
    "code": "SUCCESS",
    "data": "success",
    "msg": "success"
}
```

# 7.13 Fetch Door Emergency Status

This API allows you to fetch the emergency status for all doors.

- Request URL: `/api/v1/developer/doors/settings/emergency`
- Permission Key: `view:space`
- Method: `GET`
- UniFi Access Requirement: `1.24.6 or later`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Response Body

| Parameter | Required | Type | Description | Example |
| --- | --- | --- | --- | --- |
| lockdown | F | Boolean | `True` will keep the door locked. | true |
| evacuation | F | Boolean | `True` will keep the door unlocked. | false |

## Request Sample

The request body should be a JSON object containing the following fields:

```
curl -X '{{host}}/api/v1/developer/doors/settings/emergency'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure

```

## Response Sample

```
{
    "code": "SUCCESS",
    "data": {
        "evacuation": true,
        "lockdown": false
    },
    "msg": "success"
}
```

# 8. Device

> The APIs here are designed for device management, including obtaining device lists, device information, device statuses, device configurations, and more.

## 8.1 Fetch Devices

Obtain a list of all current devices. The device ID is required for enrolling an NFC card.

- Request URL: `/api/v1/developer/devices`
- Permission Key: `view:device`
- Method: `GET`

### Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

### Response Body

| Parameter | Required | Type | Description | Purpose | Related API |
| --- | --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the device. | Used for enrolling an NFC card. | `/api/v1/developer/credentials/nfc_cards/sessions` |
| name | T | String | Name of the device. | | |
| type | T | String | Type of the device. | | |
| alias | T | String | Alias of the device. | | |

### Request Sample: Shell/cURL

```shell
curl '{{host}}/api/v1/developer/devices'
   -H 'Authorization: Bearer wHFmHR******kD6wHg'
   -H 'accept: application/json'
   --insecure
```

### Response Body

```
{
    "code": "SUCCESS",
    "data": [
        [
            {
                "alias": "UNVR - Main Floor",
                "id": "7483c2773855",
```

```
                "name": "UA-HUB-3855",
                "type": "UAH"
            },
            {
                "alias": "UNVR - Main Floor",
                "id": "f492bfd28ced",
                "name": "UA-LITE-8CED",
                "type": "UDA-LITE"
            },
            {
                "alias": "UNVR - Main Floor",
                "id": "0418d6a2bb7a",
                "name": "UA-G2-PRO-BB7A",
                "type": "UA-G2-PRO"
            }
        ]
    ],
    "msg": "success"
}
```

# 9. System Log

> The APIs here are designed for system log management.

## 9.1 Topic Reference

| Parameter | Description |
| --- | --- |
| all | Fetch all logs. |
| door_openings | Fetch door opening logs. |
| critical | Fetch logs for device restart, deletion, offline status, and detection. |
| updates | Fetch device update logs. |
| device_events | Fetch logs for device online status, device updates, access policy synchronization, and active and inactive door unlock schedules. |
| admin_activity | Fetch logs for admin activity, such as access policy updates, settings changes, and user management. |
| visitor | Fetch logs of of visitor-related operations. |

### 9.1.1 Event Structure

**Event:** Basic information about the event.

| Key | Value (Example) |
| --- | --- |
| Type | access.door.unlock |
| Display Message | Access Granted (Remote) |
| Result | ACCESS |
| Published | 1701087091000 |
| Tag | access |

**Actor:** Information about the event operator.

| Key | Value |
| --- | --- |
| ID | [Actor ID] |
| Type | user |
| Display Name | [Display Name] |
| Alternate ID | [Alternate ID] |
| Alternate Name | [Alternate Name] |
| Avatar | [Avatar] |
| SSO Picture | [SSO Picture] |

**Authentication:** Certification information.

| Key | Value |
| --- | --- |
| Credential Provider | REMOTE_THROUGH_UAH |
| Issuer | [Issuer] |

**Target(s):** Additional information associated with the event. **Note that each event contains different attributes.**

| Type | ID | Display Name | Alternate ID | Alternate Name |
| --- | --- | --- | --- | --- |
| UAH | 7483c2773855 | UA-HUB-3855 | [Alternate ID] | [Alternate Name] |
| device_config | door_entry_method | entry/exit | [Alternate ID] | [Alternate Name] |
| door | e4978b83-203d-4015-97df-b86efc91cb0c | Door 3855 | [Alternate ID] | [Alternate Name] |
| floor | 04ddb371-457f-44ae-b8cc-8e96bcee5de4 | 1F | [Alternate ID] | [Alternate Name] |
| building | e622671e-89a5-11ee-8897-76acb95e28d5 | UDM Pro | [Alternate ID] | [Alternate Name] |

# 9.2 Fetch System Logs

This API enables you to fetch system logs.

- Request URL: `/api/v1/developer/system/logs`
- Permission Key: `view:system_log`
- Method: `POST`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Query Parameters

| Parameter | Required | Type | Description | Example |
| --- | --- | --- | --- | --- |
| page_num | F | String | Current page number in the pagination. | 1 |
| page_size | F | String | Number of logs per page. | 25 |

## Request Body

| Parameter | Required | Type | Description | Example |
| --- | --- | --- | --- | --- |
| topic | T | String | Fetch different system logs by topic. `enum topic {critical,door_openings,updates,device_events,admin_activity,visitor}` | door_openings |
| since | F | Integer | Start time for log fetching. | 1689304925 |
| until | F | Integer | End time for log fetching. | 1689804925 |
| actor_id | F | String | Identity ID of the actor (user, visitor, and device). | 3e1f196e-c97b-4748-aecb-eab5e9c251b2 |

## Response Body

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| actor | T | Object | Information about the event operator. |
| event | T | Object | Basic information about the event. |
| authentication | F | Object | Certification information. |
| target | F | Array[Object] | Additional information associated with the event, such as updated information. |

## Request Sample: Shell/cURL

```
curl '{{host}}/api/v1/developer/system/logs?page_size=1&page_num=25'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data '{
        "topic": "door_openings",
        "since": 1690770546,
        "until": 1690771546,
        "actor_id":"3e1f196e-c97b-4748-aecb-eab5e9c251b2"
    }'
    --insecure
```

## Response Body


```
{
    "code": "SUCCESS",
    "data": {
        "hits": [
            {
                "@timestamp": "2023-07-11T12:11:27Z",
                "_id": "",
                "_source": {
                    "actor": {
                        "alternate_id": "",
                        "alternate_name": "",
                        "display_name": "N/A",
                        "id": "",
                        "type": "user"
                    },
                    "authentication": {
                        "credential_provider": "NFC",
                        "issuer": "6FC02554"
                    },
                    "event": {
                        "display_message": "Access Denied / Unknown (NFC)",
                        "published": 1689077487000,
                        "reason": "",
                        "result": "BLOCKED",
                        "type": "access.door.unlock"
                    },
                    "target": [
                        {
                            "alternate_id": "",
                            "alternate_name": "",
                            "display_name": "UA-HUB-3855",
                            "id": "7483c2773855",
                            "type": "UAH"
                        }
```

```
                    ]
                },
                "tag": "access"
            }
        ]
    },
    "page": 1,
    "total": 4
}
```

# 9.3 Export System Logs

This API enables you to export system logs to a CSV file.

- Request URL: `/api/v1/developer/system/logs/export`
- Permission Key: `view:system_log`
- Method: `POST`
- UniFi Access  Requirement: `1.20.11 or later`

## Request Header

| Parameter | Required | Type | Description |
|---|---|---|---|
| Authorization | T | String | Token required for authentication and access control. |

## Request Body

| Parameter | Required | Type | Description | Example |
|---|---|---|---|---|
| topic | T | String | Fetch different system logs by topic. `enum topic {critical,door_openings,updates,device_events,admin_activity,visitor}` | door_openings |
| since | T | Integer | Start time for log fetching. | 1689304925 |
| until | T | Integer | End time for log fetching. **Note that the since and until periods cannot exceed one month.** | 1689804925 |
| timezone | T | String | Timezone for formatting time. | America/New_York |
| actor_id | F | String | Identity ID of the actor (user, visitor, and device). | 3e1f196e-c97b-4748-aecb-eab5e9c251b2 |

## Request Sample: Shell/cURL

```
curl '{{host}}/api/v1/developer/system/logs/export'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data '{
        "topic": "door_openings",
        "since": 1690770546,
        "until": 1690771546,
        "timezone": "America/New_York",
        "actor_id":"3e1f196e-c97b-4748-aecb-eab5e9c251b2"
    }'
    --insecure
```

# 9.4 Fetch Resources in System Logs

This API enables you to fetch the resources in system logs.

- Request URL: `api/v1/developer/system/logs/resource/:id`
- Permission Key: `view:system_log`
- Method: `GET`
- UniFi Access Requirement: `1.24.6 or later`

## Request Header

| Parameter | Required | Type | Description |
|---|---|---|---|
| Authorization | T | String | Token required for authentication and access control. |

## Request Body

| Parameter | Required | Type | Description | Example |
|---|---|---|---|---|
| Id | T | String | Resource ID is obtained from targets categorized as the 'activities_resource' type in system logs. | "target": [{"type": "activities_resource","id": "0418d6a38f00-b6906057-2a90-4426-835c-b5b172381fec","display_name": "Resource","alternate_id": "","alternate_name": ""}] |

## Response Sample

```
{
    "code": "SUCCESS",
    "msg": "success",
    "data": {
        "video_record": "/activities_resource/video/1708424638_f24a7b67-c584-4e22-a7b8-074f0fa93da0.mp4",
        "video_record_thumbnail": "/activities_resource/thumbnail/1708424638_a6d1fa60-d895-4d4c-a40c-447f97c8824f.jpg",
        "created_at": "2024-02-20T18:23:58+08:00",
        "updated_at": "2024-02-20T18:23:58+08:00"
    }
}
```


## Request Sample: Shell/cURL

```
curl '{{host}}/api/v1/developer/system/logs/resource/0418d6a38f00-b6906057-2a90-4426-835c-b5b172381fec'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure
```

# 9.5 Fetch Static Resources in System Logs

This API enables you to fetch static resources in system logs.

- Request URL: `api/v1/developer/system/static/:path`
- Permission Key: `view:system_log`
- Method: `GET`
- UniFi Access Requirement: `1.24.6 or later`

## Request Header

| Parameter | Required | Type | Description |
|---|---|---|---|
| Authorization | T | String | Token required for authentication and access control. |

## Request Body

| Parameter | Required | Type | Description | Example |
|---|---|---|---|---|
| path | T | String | Resource paths, currently supporting `/avatar`, `/capture`, and `activities_resource`. | |

## Request Sample: Shell/cURL

```shell
# avatar resource
curl '{{host}}/api/v1/developer/system/static/avatar/da1aceb6-20ba-4285-a6b1-c4f2bf7161f8'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure

# preview resource
curl '{{host}}/api/v1/developer/system/static/preview/1700707333_9660da3a-06c8-459d-8cc9-24889d13fba5.png'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure

# capture video resource
curl '{{host}}/api/v1/developer/system/static/activities_resource/video/1708402379_92746868-5c69-4a11-9d4c-33f03785d741.jpg'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure

# capture thumbnail resource
curl '{{host}}/api/v1/developer/system/static/activities_resource/thumbnail/1708402379_92746868-5c69-4a11-9d4c-33f03785d741.jpg'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure
```

# 10. UniFi Identity

> The APIs here are designed for the UniFi Identity app.

## 10.1 Send UniFi Identity Invitations

This API enables you to send invitations and invite users to join UniFi Identity.

- Request URL: `/api/v1/developer/users/identity/invitations`
- Permission Key: `edit:user`
- Method: `POST`
- UniFi Access Requirement: `1.24.6 or later`

### Request Header

| Parameter | Required | Type | Description |
|---|---|---|---|
| Authorization | T | String | Token required for authentication and access control. |

### Request Body

| Parameter | Required | Type | Description |
|---|---|---|---|
| Array[Object].user_id | T | String | Identity ID of the user. |
| Array[Object].email | F | String | Email of the user. **If filled in, it will update the user's existing email address.** |

### Request Sample: Shell/cURL

```
curl -XPOST '{{host}}/api/v1/developer/users/identity/invitations'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data-raw '[
    {
        "user_id": "e0051e08-c4d5-43db-87c8-a9b19cb66513",
        "email": "example@*.com"
    },

]'
    --insecure
```

## Response Body

```
# Success
{
    "code": "SUCCESS",
    "data": [],
    "msg": "success"
}

# If there is a failure to send an email
{
    "code": "SUCCESS",
    "data": [
        {
            "error_code": "",
            "error_msg": "invalid email",
            "user_email": "example@*.com",
            "user_id": "e0051e08-c4d5-43db-87c8-a9b19cb66513"
        }
    ],
    "msg": "success"
}
```

# 10.2 Fetch Available Resources

This API enables you to fetch the available UniFi Identity resources.

- Request URL: `/api/v1/developer/users/identity/assignments`
- Permission Key: `view:user`
- Method: `GET`
- UniFi Access Requirement: `1.24.6 or later`

## Request Header

| Parameter | Required | Type | Description |
|---|---|---|---|
| Authorization | T | String | Token required for authentication and access control. |

## Query Parameters

| Parameter | Required | Type | Description | Example |
|---|---|---|---|---|
| resource_type | F | String | Display the type of resources; show all when left blank. `enum resource_type {ev_station,vpn,wifi}` | resource_type=ev_station,wifi,vpn |

## Response Body

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| data.[resource_type] | T | String | Type of the resources. `enum resource_type {ev_station,vpn,wifi,camera}` |
| id | T | String | Identity ID of the resources. |
| name | T | String | Name of the resources. |
| deleted | T | Boolean | Indicate whether the resource is disabled. |

## Request Sample: Shell/cURL

```
curl '{{host}}/api/v1/developer/users/identity/assignments?resource_type=ev_station,wifi,vpn'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure
```

## Response Sample


```
{
    "code": "SUCCESS",
    "data": {
        "ev_station": [],
        "vpn": [
            {
                "deleted": false,
                "id": "65dff9a9c188cb71cfac8e9d",
                "metadata": null,
                "name": "UDM Pro",
                "short_name": ""
            }
        ],
        "wifi": [
            {
                "deleted": false,
                "id": "65dff9a8c188cb71cfac8e9a",
                "metadata": null,
                "name": "UniFi Identity",
                "short_name": ""
            }
        ]
    },
    "msg": "success"
}
```

# 10.3 Assign Resources to Users

This API enables you to assign UniFi Identity resources to users.

- Request URL: `/api/v1/developer/users/:id/identity/assignments`
- Permission Key: `edit:user`
- Method: `POST`
- UniFi Access Requirement: `1.24.6 or later`

## Request Header

| Parameter | Required | Type | Description |
|---|---|---|---|
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | How to Get It? |
|---|---|---|---|---|
| id | T | String | Identity ID of the user. | Get it from the API `/api/v1/developer/users` |

## Request Body

| Parameter | Required | Type | Description | How to Get It? |
|---|---|---|---|---|
| resource_type | T | String | `enum resource_type {ev_station,vpn,wifi}` | Get it from the API 10.2 |
| resource_ids | T | Array[String] | Identity ID of the resources. | Get it from the API 10.2 |

## Request Sample: Shell/cURL

```
curl -XPOST '{{host}}/api/v1/developer/users/b602879b-b857-400b-970b-336d4cb881ad/identity/assignments'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data '{
        "resource_type": "wifi",
        "resource_ids": [
            "65dff9a8c188cb71cfac8e9a"
        ]
    }'
    --insecure
```

## Response Sample

```
{
    "code": "SUCCESS",
    "data": null,
    "msg": "success"
}
```

# 10.4 Fetch Resources Assigned to Users

This API enables you to fetch the UniFi Identity resources assigned to users.

- Request URL: `/api/v1/developer/users/:id/identity/assignments`
- Permission Key: `view:user`
- Method: `GET`
- UniFi Access Requirement: `1.24.6 or later`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the user. | Get it from the API `/api/v1/developer/users` |

## Response Body

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| data.[resource_type] | T | Array[Object] | Type of the resources. `enum resource_type {ev_station,vpn,wifi,camera}` |
| id | T | String | Identity ID of the resources. |
| name | T | String | Name of the resources. |
| deleted | T | Boolean | Indicate whether a resource is disabled. |

## Request Sample: Shell/cURL

```
curl -XGET '{{host}}/api/v1/developer/users/b602879b-b857-400b-970b-336d4cb881ad/identity/assignments'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure
```

## Response Sample


```
{
    "code": "SUCCESS",
    "data": {
        "ev_station": [],
        "vpn": [
            {
                "deleted": false,
                "id": "65dff9a9c188cb71cfac8e9d",
                "metadata": {
                    "has_ip": true
                },
                "name": "UDM Pro",
                "short_name": ""
            }
        ],
        "wifi": [
            {
                "deleted": false,
                "id": "65dff9a8c188cb71cfac8e9a",
                "metadata": null,
                "name": "UniFi Identity",
                "short_name": ""
            }
        ]
    },
    "msg": "success"
}
```


# 10.5 Assign Resources to User Groups

This API enables you to assign UniFi Identity resources to user groups.

- Request URL: `/api/v1/developer/user_groups/:id/identity/assignments`
- Permission Key: `edit:user`
- Method: `POST`
- UniFi Access Requirement: `2.2.0 or later`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the user group. | Get it from the API `/api/v1/developer/user_groups` |

## Request Body

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| resource_type | T | String | `enum resource_type {ev_station,vpn,wifi}` | Get it from the API 10.2 |
| resource_ids | T | Array[String] | Identity ID of the resources. | Get it from the API 10.2 |

## Request Sample: Shell/cURL

```
curl -XPOST '{{host}}/api/v1/developer/user_groups/7476c839-8e10-472e-894f-c5b8254c35b5/identity/assignments'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data '{
        "resource_type": "wifi",
        "resource_ids": [
            "65dff9a8c188cb71cfac8e9a"
        ]
    }'
    --insecure
```

## Response Sample

```
{
    "code": "SUCCESS",
    "data": null,
    "msg": "success"
}
```

# 10.6 Fetch the Resources Assigned to User Groups

This API enables you to fetch the UniFi Identity resources assigned to user groups.

- Request URL: `/api/v1/developer/user_groups/:id/identity/assignments`
- Permission Key: `view:user`
- Method: `GET`
- UniFi Access Requirement: `2.2.0 or later`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | How to Get It? |
| --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the user group. | Get it from the API `/api/v1/developer/user_groups` |

## Response Body

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| data.[resource_type] | T | Array[Object] | Type of the resources. `enum resource_type {ev_station,vpn,wifi,camera}` |
| id | T | String | Identity ID of the resources. |
| name | T | String | Name of the resources. |
| deleted | T | Boolean | Indicate whether a resource is disabled. |

## Request Sample: Shell/cURL

```shell
curl -XGET '{{host}}/api/v1/developer/user_groups/7476c839-8e10-472e-894f-c5b8254c35b5/identity/assignments'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure
```

## Response Sample

```
{
    "code": "SUCCESS",
    "data": {
        "ev_station": [],
        "vpn": [
            {
                "deleted": false,
                "id": "65dff9a9c188cb71cfac8e9d",
                "metadata": {
```

```
                    "has_ip": true
                },
                "name": "UDM Pro",
                "short_name": ""
            }
        ],
        "wifi": [
            {
                "deleted": false,
                "id": "65dff9a8c188cb71cfac8e9a",
                "metadata": null,
                "name": "UniFi Identity",
                "short_name": ""
            }
        ]
    },
    "msg": "success"
}
```

# 11. Notification

> The APIs here are designed for Webhooks and WebSockets.

## 11.1 Fetch Notifications [WebSocket]

This API enables you to fetch notifications, such as doorbell notifications.

- Request URL: `/api/v1/developer/devices/notifications`
- Permission Key: `view:device`
- Protocol: `WebSocket`
- Method: `GET`
- UniFi Access Requirement: `1.20.11 or later`

### Request Sample: wscat

```
wscat
    --no-check
    -c wss://192.168.1.1:12445/api/v1/developer/devices/notifications
    -H "Authorization: Bearer qoFJM******9YQX0+g+g"
    -H "Upgrade: websocket"
    -H "Connection: Upgrade"
```

### WebSocket Payload

> When a doorbell rings: [access.remote_view]


```
{
    "event": "access.remote_view",
    "receiver_id": "",
    "event_object_id": "535b6125-860c-489a-b0a1-0ba01906afa9",
    "save_to_history": false,
    "data": {
        "channel": "4513899f-0370-4116-9731-63a6b0feea23",
        "token": "6dff120f-2688-497d-856f-0ca08b383d1d",
        "device_id": "e4388386be1d",
        "device_type": "UA-G2-PRO",
        "device_name": "UA-G2-PRO-BE1D",
        "door_name": "Door 236b",
        "controller_id": "68d79a1f494f",
        "floor_name": "1F",
        "request_id": "J0FeDJc8ZNzHjxr1SUIP6GLQDjAkdZFp",
        "clear_request_id": "",
        "in_or_out": "in",
        "create_time": 1694771479,
```

```
        "reason_code": 0,
        "door_guard_ids": [
            "daa10301-7744-4623-a90e-372718af1d41",
            "9d526114-70ce-49ec-8655-81767ffb3eb4",
            "c003514f-60bb-4aa3-9150-cd361b1458a0",
            "0d6273e4-9b54-4b91-b7f8-40d0d41780c1",
            "f395f473-e6ea-4232-a45f-9ec20c813e96",
            "c52f9920-be17-4357-936c-47d94dad753a",
            "8ed10bf7-8f5e-4b68-98da-76d2f315d515",
            "e4de9c92-e385-4b31-8cfc-8e9d192c0e10",
            "2159ac55-a1d9-42f3-ba2b-6c87cf5ea383"
        ],
        "connected_uah_id": "e4388384236b",
        "room_id": "WR-e4388386be1d-3YSCjtuV5VuyfLT46mUXnnY2q5KQfKxX",
        "host_device_mac": "68D79A1F494B"
    }
}
```


> Doorbell status change: [access.remote_view.change]

| reason_code | Description |
| --- | --- |
| 105 | Doorbell timed out. |
| 106 | An admin rejected to unlock a door. |
| 107 | An admin successfully unlocked a door. |
| 108 | A visitor canceled a doorbell. |
| 400 | Doorbell was answered by another admin. |


```
{
    "event": "access.remote_view.change",
    "receiver_id": "",
    "event_object_id": "450a6c0f-28f7-47ca-99c3-5a35475fece4",
    "save_to_history": false,
    "data": {
        "reason_code": 108,
        "remote_call_request_id": "J0FeDJc8ZNzHjxr1SUIP6GLQDjAkdZFp"
    }
}
```


> Remote door unlock by admin: [access.data.device.remote_unlock]


```
{
    "event": "access.data.device.remote_unlock",
    "receiver_id": "",
    "event_object_id": "e4388384236b",
```

```
    "save_to_history": false,
    "data": {
        "unique_id": "5d600936-8618-4f2d-a1b7-d786865b70ba",
        "name": "Door 236b",
        "up_id": "913a35bc-66c9-4293-a617-846dd2e285ed",
        "timezone": "",
        "location_type": "door",
        "extra_type": "",
        "full_name": "UDR-ML - 1F - Door 236b",
        "level": 0,
        "work_time": "[]",
        "work_time_id": "",
        "extras": {
            "uah-input_state_dps": "off",
            "uah-wiring_state_dps-neg": "off",
            "uah-wiring_state_dps-pos": "off"
        }
    }
}
```


## 11.2 List of Supported Webhook Events [Webhook]

> To enable webhook events, you must register webhook endpoints. Once registered, UniFi Access pushes real-time event data to your application's webhook endpoint as events occur. UniFi Access uses HTTPS to send these webhook events to your application, delivering a JSON payload containing a JSON object.

| Event/Operation | Description |
| --- | --- |
| **access.doorbell.incoming** | Incoming doorbell notifications. |
| **access.doorbell.completed** | Accept, decline, or cancel doorbell events. |
| **access.doorbell.incoming.REN** | Doorbells triggered via request-to-enter (REN) buttons. |
| **access.device.dps_status** | Changes in door position sensor (DPS) status. |
| **access.door.unlock** | All door unlock events. |
| **access.device.emergency_status** | Changes in emergency mode. |

## 11.3 Fetch Webhook Endpoints List [Webhook]

This API enables you to fetch the available webhook endpoints.

- Request URL: `/api/v1/developer/webhooks/endpoints`

- Permission Key: `view:webhook`
- Method: `GET`
- UniFi Access Requirement: `2.2.10 or later`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Response Body

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| id | T | String | Identity ID of the endpoint. |
| endpoint | T | String | The HTTPS URL where webhook events are sent. |
| name | T | String | Name of the webhook subscription. |
| secret | T | String | The secret used for verifying events coming from UniFi Access. |
| events | T | Array[String] | List of events to subscribe to. |
| headers | F | Object<String, String> | Custom headers for forwarding requests to the endpoint service. |

## Request Sample: Shell/cURL

```
curl '{{host}}/api/v1/developer/webhooks/endpoints'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure
```

## Response Sample

```
{
    "code": "SUCCESS",
    "data": [
        {
            "endpoint": "http://192.168.1.1:8080",
            "events": [
                "access.doorbell.incoming",
                "access.doorbell.completed",
                "access.doorbell.incoming.REN",
```

```
                "access.device.dps_status",
                "access.door.unlock"
            ],
            "id": "1a639160-a7c8-45cb-8789-50dfd255a0fe",
            "name": "subscription events",
            "secret": "6601f1243d2ff70f",
            "headers": {
                "key": "value"
            }
        }
    ],
    "msg": "success"
}
```

# 11.4 Add Webhook Endpoints [Webhook]

This API enables you to add a webhook endpoint.

- Request URL: `/api/v1/developer/webhooks/endpoints`
- Permission Key: `edit:webhook`
- Method: `POST`
- UniFi Access Requirement: `2.2.10 or later`

## Request Header

| Parameter | Required | Type | Description |
|---|---|---|---|
| Authorization | T | String | Token required for authentication and access control. |

## Request Body

| Parameter | Required | Type | Description |
|---|---|---|---|
| endpoint | T | String | The HTTPS URL where webhook events are sent. **Note**: HTTPS is recommended for security. |
| name | T | String | Name of the webhook subscription. |
| events | T | Array[String] | List of events to subscribe to. |
| headers | F | Object<String, String> | Custom headers for forwarding requests to the endpoint service. |

## Request Sample: Shell/cURL

```shell
curl -XPOST '{{host}}/api/v1/developer/webhooks/endpoints'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data '{
        "name": "subscription events",
        "endpoint": "http://192.168.1.1:8080",
        "events": [
            "access.doorbell.incoming",
            "access.doorbell.completed",
            "access.doorbell.incoming.REN",
            "access.device.dps_status",
            "access.door.unlock"
        ],
        "headers": {
            "key": "value"
         }
    }
    --insecure
```

## Response Sample

```shell
{
    "code": "SUCCESS",
    "data": {
        "endpoint": "http://192.168.1.1:8080",
        "events": [
            "access.doorbell.incoming",
            "access.doorbell.completed",
            "access.doorbell.incoming.REN",
            "access.device.dps_status",
            "access.door.unlock"
        ],
        "id": "a22ee283-c91f-432b-9d0f-e89bcccad4be",
        "name": "subscription events",
        "secret": "1a7c9c6a69bb5a1e",
         "headers": {
            "key": "value"
         }
    },
    "msg": "success"
}
```

# 11.5 Update Webhook Endpoints [Webhook]

This API enables you to update the available webhook endpoints.

- Request URL: `/api/v1/developer/webhooks/endpoints/:id`
- Permission Key: `edit:webhook`
- Method: `PUT`
- UniFi Access Requirement: `2.2.10 or later`

## Request Header

| Parameter | Required | Type | Description |
|---|---|---|---|
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | Example |
|---|---|---|---|---|
| id | T | String | Identity ID of the endpoint. | a22ee283-c91f-432b-9d0f-e89bcccad4be |

## Request Body

| Parameter | Required | Type | Description |
|---|---|---|---|
| endpoint | T | String | The HTTPS URL where webhook events are sent. |
| name | T | String | Name of the webhook subscription. |
| events | T | Array[String] | List of events to subscribe to. |
| headers | F | Object<String, String> | Custom headers for forwarding requests to the endpoint service. |

## Request Sample: Shell/cURL

```
curl -XPUT '{{host}}/api/v1/developer/webhooks/endpoints/a22ee283-c91f-432b-9d0f-e89bcccad4be'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --data '{
        "name": "subscription events",
        "endpoint": "http://192.168.1.1:8080",
        "events": [
            "access.doorbell.incoming",
            "access.doorbell.completed",
```

```
            "access.doorbell.incoming.REN",
            "access.device.dps_status",
            "access.door.unlock",
            "access.device.emergency_status"
        ],
         "headers": {
            "key": "value"
         }
    }
    --insecure
```

## Response Sample

```
{
    "code": "SUCCESS",
    "data": {
        "endpoint": "http://192.168.1.1:8080",
        "events": [
            "access.doorbell.incoming",
            "access.doorbell.completed",
            "access.doorbell.incoming.REN",
            "access.device.dps_status",
            "access.door.unlock"
        ],
        "id": "a22ee283-c91f-432b-9d0f-e89bcccad4be",
        "name": "subscription events",
        "secret": "1a7c9c6a69bb5a1e",
         "headers": {
            "key": "value"
         }
    },
    "msg": "success"
}
```

# 11.6 Delete Webhook Endpoints [Webhook]

This API enables you to delete the available webhook endpoints.

- Request URL: `/api/v1/developer/webhooks/endpoints/:id`
- Permission Key: `edit:webhook`
- Method: `DELETE`
- UniFi Access Requirement: `2.2.10 or later`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Request Path

| Parameter | Required | Type | Description | Example |
| --- | --- | --- | --- | --- |
| id | T | String | Identity ID of the endpoint. | a22ee283-c91f-432b-9d0f-e89bcccad4be |

## Request Sample: Shell/cURL

```
curl -XDELETE '{{host}}/api/v1/developer/webhooks/endpoints/a22ee283-c91f-432b-9d0f-e89bcccad4be'
    -H 'Authorization: Bearer wHFmHR******kD6wHg'
    -H 'accept: application/json'
    -H 'content-type: application/json'
    --insecure
```

## Response Sample

```
{
    "code": "SUCCESS",
    "data": null,
    "msg": "success"
}
```

# 11.7 Allow Webhook Endpoint Owner to Receive Webhook Events [Webhook]

The following samples demonstrate how to receive webhook messages. Note that the secret needs to be adjusted to the assigned secret. These are just examples; in actual use, HTTPS and a custom URL should be used.

## Request from UniFi Access

- Request URL: `Your webhook endpoint`
- Method: `POST`
- UniFi Access Requirement: `2.2.10 or later`

| Parameter | Required | Type | Description | Example |
| --- | --- | --- | --- | --- |
| Signature | T | String | Contains request time**(t)** and signature**(v1)** | t=1695902233, v1=a7ea8840af212767d7795481bed914a9f2ea7241d35212b597bec13aa4bfa06b |

# Response Payload Sample

```
#access.door.unlock
{
    "event": "access.door.unlock",
    "event_object_id": "4a98adf6-dbb8-4312-9b8b-593f6eba8c8e",
    "data": {
        "location": {
            "id": "d2b87427-7efa-43c1-aa52-b00d40d99ecf",
            "location_type": "door",
            "name": "Door 3855",
            "up_id": "62ff3aa1-ae96-4b6b-8eb5-44aadfd4aabd",
            "extras": {
                "door_thumbnail": "/preview/reader_0418d6a2bb7a_d2b87427-7efa-
43c1-aa52-b00d40d99ecf_1722913291.jpg",
                "door_thumbnail_last_update": 1722913291,
                "uah-input_state_dps": "on",
                "uah-wiring_state_dps-neg": "on",
                "uah-wiring_state_dps-pos": "on"
            },
            "device_ids": null
        },
        "device": {
            "name": "UA-HUB-3855",
            "alias": "Door 3855",
            "id": "7483c2773855",
            "ip": "192.168.1.132",
            "mac": "",
            "online": false,
            "device_type": "UAH",
            "connected_hub_id": "",
            "location_id": "d2b87427-7efa-43c1-aa52-b00d40d99ecf",
            "firmware": "v4.6.1.0",
            "version": "v4.6.129",
            "guid": "4a5e238f-4bae-48d5-84d7-dd2b0e919ab5",
            "start_time": 1721988528,
            "hw_type": "",
            "revision": "1722912520784126005",
            "cap": null
        },
        "actor": {
            "id": "d62e92fd-91aa-44c2-9b36-6d674a4b74d0",
            "name": "Hon***",
            "type": "user"
        },
```

```
        "object": {
            "authentication_type": "CALL", //Door opening method,
NFC/PIN_CODE/Call For a DoorBell
            "authentication_value": "",
            "policy_id": "",
            "policy_name": "",
            "reader_id": "",
            "result": "Access Granted"
        }
    }
}

#access.doorbell.incoming
{
    "event": "access.doorbell.incoming",
    "event_object_id": "641cee06-89b5-4fef-9ce5-f9a2ae33ffd8",
    "data": {
        "location": {
            "id": "d2b87427-7efa-43c1-aa52-b00d40d99ecf",
            "location_type": "door",
            "name": "Door 3855",
            "up_id": "62ff3aa1-ae96-4b6b-8eb5-44aadfd4aabd",
            "extras": {
                "door_thumbnail": "/preview/reader_0418d6a2bb7a_d2b87427-7efa-
43c1-aa52-b00d40d99ecf_1722912625.jpg",
                "door_thumbnail_last_update": 1722912625,
                "uah-input_state_dps": "on",
                "uah-wiring_state_dps-neg": "on",
                "uah-wiring_state_dps-pos": "on"
            },
            "device_ids": null
        },
        "device": {
            "name": "UA G2 Pro bb7a",
            "alias": "Door 3855 - Entry2",
            "id": "0418d6a2bb7a",
            "ip": "192.168.1.37",
            "mac": "04:18:d6:a2:bb:7a",
            "online": false,
            "device_type": "UA-G2-PRO",
            "connected_hub_id": "7483c2773855",
            "location_id": "d2b87427-7efa-43c1-aa52-b00d40d99ecf",
            "firmware": "v1.5.33",
            "version": "v1.5.3543",
            "guid": "06d4277c-3dd3-4691-8aac-3e140abb8699",
            "start_time": 1722483954,
            "hw_type": "GA",
            "revision": "1722912520784126005",
            "cap": null
```

```
        },
        "actor": null,
        "object": {
            "request_id": "Ed8XSsi4fLm6St7JOLLa26UcmzAZlGzn"
        }
    }
}

#access.doorbell.completed
>  Doorbell status changes:
| reason_code | Description                              |
| ----------- | ---------------------------------------- |
| 105         | Doorbell timed out.                      |
| 106         | An admin declined the door unlock request.   |
| 107         | An admin successfully unlocked the door.  |
| 108         | A visitor canceled the doorbell.       |
| 400         | Doorbell was answered by another admin. |

{
    "event": "access.doorbell.completed",
    "event_object_id": "c4bf20b7-1bd3-4b4c-a0b9-bca3cb273bcf",
    "data": {
        "location": {
            "id": "d2b87427-7efa-43c1-aa52-b00d40d99ecf",
            "location_type": "door",
            "name": "Door 3855",
            "up_id": "62ff3aa1-ae96-4b6b-8eb5-44aadfd4aabd",
            "extras": {
                "door_thumbnail": "/preview/reader_0418d6a2bb7a_d2b87427-7efa-
43c1-aa52-b00d40d99ecf_1722913291.jpg",
                "door_thumbnail_last_update": 1722913291,
                "uah-input_state_dps": "on",
                "uah-wiring_state_dps-neg": "on",
                "uah-wiring_state_dps-pos": "on"
            },
            "device_ids": null
        },
        "device": {
            "name": "UA G2 Pro bb7a",
            "alias": "Door 3855 - Entry2",
            "id": "0418d6a2bb7a",
            "ip": "192.168.1.37",
            "mac": "04:18:d6:a2:bb:7a",
            "online": false,
            "device_type": "UA-G2-PRO",
            "connected_hub_id": "7483c2773855",
            "location_id": "d2b87427-7efa-43c1-aa52-b00d40d99ecf",
            "firmware": "v1.5.33",
            "version": "v1.5.3543",
```

```
            "guid": "06d4277c-3dd3-4691-8aac-3e140abb8699",
            "start_time": 1722483954,
            "hw_type": "GA",
            "revision": "1722912520784126005",
            "cap": null
        },
        "actor": null,
        "object": {
            "host_device_mac": "",
            "reason_code": 400,
            "request_id": "Ed8XSsi4fLm6St7JOLLa26UcmzAZlGzn" //Request ID for doorbell
        }
    }
}

#access.device.dps_status
{
    "event": "access.device.dps_status",
    "event_object_id": "229899f2-ba52-461a-a3af-e011fb5078d1",
    "data": {
        "location": {
            "id": "d2b87427-7efa-43c1-aa52-b00d40d99ecf",
            "location_type": "door",
            "name": "Door 3855",
            "up_id": "62ff3aa1-ae96-4b6b-8eb5-44aadfd4aabd",
            "extras": {
                "door_thumbnail": "/preview/reader_0418d6a2bb7a_d2b87427-7efa-43c1-aa52-b00d40d99ecf_1722912625.jpg",
                "door_thumbnail_last_update": 1722912625,
                "uah-input_state_dps": "on",
                "uah-wiring_state_dps-neg": "on",
                "uah-wiring_state_dps-pos": "on"
            },
            "device_ids": null
        },
        "device": {
            "name": "UA-HUB-3855",
            "alias": "Door 3855",
            "id": "7483c2773855",
            "ip": "192.168.1.132",
            "mac": "74:83:c2:77:38:55",
            "online": false,
            "device_type": "UAH",
            "connected_hub_id": "",
            "location_id": "d2b87427-7efa-43c1-aa52-b00d40d99ecf",
            "firmware": "v4.6.1.0",
            "version": "v4.6.129",
            "guid": "4a5e238f-4bae-48d5-84d7-dd2b0e919ab5",
```

```
            "start_time": 1721988528,
            "hw_type": "",
            "revision": "1722912520784126005",
            "cap": null
        },
        "actor": null,
        "object": {
            "event_type": "dps_change", //DPS change type
            "status": "close" //DPS change status
        }
    }
}

#access.doorbell.incoming.REN
{
    "event": "access.doorbell.incoming.REN",
    "event_object_id": "a819573f-a273-4909-b13b-b1477b9e4a19",
    "data": {
        "location": {
            "id": "d2b87427-7efa-43c1-aa52-b00d40d99ecf",
            "location_type": "door",
            "name": "Door 3855",
            "up_id": "62ff3aa1-ae96-4b6b-8eb5-44aadfd4aabd",
            "extras": {
                "door_thumbnail": "/preview/reader_0418d6a2bb7a_d2b87427-7efa-
43c1-aa52-b00d40d99ecf_1722912625.jpg",
                "door_thumbnail_last_update": 1722912625,
                "uah-input_state_dps": "off",
                "uah-wiring_state_dps-neg": "off",
                "uah-wiring_state_dps-pos": "off"
            },
            "device_ids": null
        },
        "device": null,
        "actor": null,
        "object": {
            "host_device_mac": "74ACB95E28D5",
            "reason_code": 0,
            "request_id": "65d22fd2-29d7-43a8-b0f5-30a6722fcf1e" //Request ID for
doorbell
        }
    }
}

#access.device.emergency_status
{
    "event": "access.device.emergency_status",
    "event_object_id": "6db3c53a-3c7f-4794-967b-78d4489b4ff0",
    "data": {
```

```
        "device": {
            "name": "UA Hub Gate",
            "alias": "UA Hub Gate",
            "id": "f4e2c61fd2cf",
            "ip": "192.168.1.238",
            "mac": "f4:e2:c6:1f:d2:cf",
            "online": false,
            "adopting": false,
            "device_type": "UGT",
            "connected_hub_id": "",
            "location_id": "05b37c0f-f917-4cbb-b63c-3b4d8e5e35f7",
            "firmware": "v2.1.7.0",
            "version": "v2.1.76",
            "guid": "2fe09035-e671-4804-aa44-575d1cd6f0c6",
            "start_time": 1737690291,
            "hw_type": "",
            "revision": "1737692257198917966",
            "cap": null,
            "category": null
        },
        "object": {
            "glocal": false, //All locations will be unlocked.
            "mode": "lockdown", // lockdown or evacuation
            "value": true // true is open, false is close
        }
    }
}
```


## Handle Request（Using GO as a Sample）

> You can get the secret here. It is recommended to handle asynchronous processing logic, as there is currently a 5-second timeout set.

```go
package main

import (
    "crypto/hmac"
    "crypto/sha256"
    "encoding/hex"
    "encoding/json"
    "errors"
    "fmt"
    "io/ioutil"
    "log"
    "net/http"
    "strconv"
    "strings"
    "time"
)
```

```go

const (
    port   = ":8080"
    secret = "6601f1243d2ff70f"
)

func handlePostRequest(w http.ResponseWriter, r *http.Request) {
    signature := r.Header.Get("Signature")

    for name, values := range r.Header {
        for _, value := range values {
            fmt.Printf("headers: %s: %s\n", name, value)
        }
    }

    body, err := ioutil.ReadAll(r.Body)
    if err != nil {
        http.Error(w, "Failed to read request body", http.StatusBadRequest)
        return
    }

    _, err = ConstructEvent(body, signature, secret)
    if err != nil {
        w.WriteHeader(http.StatusInternalServerError)
        w.Write([]byte(err.Error()))
        return
    }

    w.WriteHeader(http.StatusOK)
    w.Write([]byte("OK"))
}

func main() {
    http.HandleFunc("/", handlePostRequest)

    fmt.Printf("Server listening on port %s...\n", port)
    log.Fatal(http.ListenAndServe(port, nil))
}

var (
    ErrInvalidHeader    = errors.New("webhook has invalid Signature header")
    ErrNoValidSignature = errors.New("webhook had no valid signature")
    ErrNotSigned        = errors.New("webhook has no Signature header")
)

func ConstructEvent(payload []byte, header string, secret string)
(json.RawMessage, error) {
    return constructEvent(payload, header, secret)
}
```

```go

type signedHeader struct {
    timestamp time.Time
    signature []byte
}

func constructEvent(payload []byte, sigHeader string, secret string)
(json.RawMessage, error) {
    var e json.RawMessage

    if err := validatePayload(payload, sigHeader, secret); err != nil {
        return e, err
    }

    if err := json.Unmarshal(payload, &e); err != nil {
        return e, fmt.Errorf("Failed to parse webhook body json: %s", err.Error())
    }

    return e, nil
}

var signingVersion = "v1"

func parseSignatureHeader(header string) (*signedHeader, error) {
    sh := &signedHeader{}

    if header == "" {
        return sh, ErrNotSigned
    }

    pairs := strings.Split(header, ",")
    for _, pair := range pairs {
        parts := strings.Split(pair, "=")
        if len(parts) != 2 {
            return sh, ErrInvalidHeader
        }

        switch parts[0] {
        case "t":
            timestamp, err := strconv.ParseInt(parts[1], 10, 64)
            if err != nil {
                return sh, ErrInvalidHeader
            }
            sh.timestamp = time.Unix(timestamp, 0)

        case signingVersion:
            sig, err := hex.DecodeString(parts[1])
            if err != nil {
                continue
```

```
            }

            sh.signature = sig
        }
    }

    if len(sh.signature) == 0 {
        return sh, ErrNoValidSignature
    }

    return sh, nil
}

func validatePayload(payload []byte, sigHeader string, secret string) error {
    header, err := parseSignatureHeader(sigHeader)
    if err != nil {
        return err
    }

    expectedSignature := computeSignature(header.timestamp, payload, secret)
    if hmac.Equal(expectedSignature, header.signature) {
        return nil
    }

    return ErrNoValidSignature
}

func computeSignature(t time.Time, payload []byte, secret string) []byte {
    mac := hmac.New(sha256.New, []byte(secret))
    mac.Write([]byte(fmt.Sprintf("%d", t.Unix())))
    mac.Write([]byte("."))
    mac.Write(payload)
    return mac.Sum(nil)
}

func hexSignature(signature []byte) string {
    return hex.EncodeToString(signature)
}
```

## Handle Request（Using Rust as a Sample）

> You can get the secret here. It is recommended to handle asynchronous processing logic, as there is currently a 5-second timeout set.

```
use hex::FromHex;
use sha2::Sha256;
use warp::Filter;
use bytes::Bytes;
use hmac::{Hmac, Mac};
```

```rust

const SECRET: &[u8] = b"6601f1243d2ff70f";

#[tokio::main]
async fn main() {
    let routes = warp::post()
        .and(warp::body::bytes())
        .and(warp::header::header("Signature"))
        .and_then(handle_post_request);

         warp::serve(routes).run(([0, 0, 0, 0], 8080)).await;
}

async fn handle_post_request(payload: Bytes, signature: String) -> Result<impl warp::Reply, warp::Rejection> {
    let payload_str = std::str::from_utf8(&payload)
        .map_err(|_| warp::reject::custom(ErrorKind::InvalidPayload))?;

    let validated_payload = validate_payload(payload_str, &signature, SECRET)?;
    println!("Validated Payload: {}", validated_payload);

    Ok(warp::reply::with_status("OK", warp::http::StatusCode::OK))
}

#[derive(Debug)]
enum ErrorKind {
    InvalidHeader,
    NoValidSignature,
    NotSigned,
    InvalidPayload,
}

impl warp::reject::Reject for ErrorKind {}

fn validate_payload<'a>(payload: &'a str, sig_header: &'a str, secret: &[u8]) -> Result<&'a str, warp::reject::Rejection> {
    let header = parse_signature_header(sig_header)?;

    let expected_signature = compute_signature(header.timestamp, payload.as_bytes(), secret);

    if expected_signature == header.signature {
        return Ok(payload);
    }

    Err(warp::reject::custom(ErrorKind::NoValidSignature))
}

struct SignedHeader {
```

```rust
    timestamp: u64,
    signature: Vec<u8>,
}

fn parse_signature_header(header: &str) -> Result<SignedHeader, warp::reject::Rejection> {
    let mut sh = SignedHeader {
        timestamp: 0,
        signature: Vec::new(),
    };

    if header.is_empty() {
        return Err(warp::reject::custom(ErrorKind::NotSigned));
    }

    let pairs: Vec<&str> = header.split(',').collect();
    for pair in pairs {
        let parts: Vec<&str> = pair.split('=').collect();
        if parts.len() != 2 {
            return Err(warp::reject::custom(ErrorKind::InvalidHeader));
        }

        match parts[0] {
            "t" => {
                let timestamp = parts[1].parse::<u64>().map_err(|_| warp::reject::custom(ErrorKind::InvalidHeader))?;
                sh.timestamp = timestamp;
            }
            "v1" => {
                let sig = Vec::from_hex(parts[1]).ok();
                if let Some(sig) = sig {
                    sh.signature = sig;
                }
            }
            _ => continue,
        }
    }

    if sh.signature.is_empty() {
        return Err(warp::reject::custom(ErrorKind::NoValidSignature));
    }

    Ok(sh)
}

type HmacSha256 = Hmac<Sha256>;

fn compute_signature(timestamp: u64, payload: &[u8], secret: &[u8]) -> Vec<u8> {
```

```
    let mut mac = HmacSha256::new_from_slice(secret).expect("HMAC can take key of any size");
    mac.update(timestamp.to_string().as_bytes());
    mac.update(b".");
    mac.update(payload);
    mac.finalize().into_bytes().to_vec()
}
```

## Handle Request（Using Python as a Sample）

> You can get the secret here. It is recommended to handle asynchronous processing logic, as there is currently a 5-second timeout set.

```python
import hmac
import hashlib
import json
from flask import Flask, request, jsonify
from datetime import datetime
import time

app = Flask(__name__)

SECRET = "6601f1243d2ff70f"
SIGNING_VERSION = "v1"

class WebhookError(Exception):
    pass

class InvalidHeaderError(WebhookError):
    pass

class NoValidSignatureError(WebhookError):
    pass

class NotSignedError(WebhookError):
    pass

def parse_signature_header(header):
    if not header:
        raise NotSignedError("Webhook has no Signature header")

    pairs = header.split(",")
    parsed_header = {}
    for pair in pairs:
        key, value = pair.split("=")
        parsed_header[key] = value

    if 't' not in parsed_header or SIGNING_VERSION not in parsed_header:
        raise InvalidHeaderError("Webhook has invalid Signature header")
```

```python

    timestamp = int(parsed_header['t'])
    signature = bytes.fromhex(parsed_header[SIGNING_VERSION])
    return timestamp, signature

def compute_signature(timestamp, payload, secret):
    mac = hmac.new(secret.encode(), digestmod=hashlib.sha256)
    mac.update(f"{timestamp}".encode())
    mac.update(b".")
    mac.update(payload)
    return mac.digest()

def validate_payload(payload, sig_header, secret):
    timestamp, received_signature = parse_signature_header(sig_header)
    expected_signature = compute_signature(timestamp, payload, secret)
    if not hmac.compare_digest(expected_signature, received_signature):
        raise NoValidSignatureError("Webhook had no valid signature")

def construct_event(payload, sig_header, secret):
    validate_payload(payload, sig_header, secret)
    return json.loads(payload)

@app.route("/", methods=["POST"])
def handle_post_request():
    signature = request.headers.get("Signature")

    try:
        payload = request.data
        event = construct_event(payload, signature, SECRET)
    except WebhookError as e:
        return str(e), 500
    except Exception as e:
        return "Failed to read request body", 400

    return "OK", 200

if __name__ == "__main__":
    app.run(port=8080)
```

# 12. API Server

> The APIs here are designed for API server configuration.

## 12.1 Upload HTTPS Certificate

This API allows you to update the HTTPS certificate for the Access API, applicable only to port **12445**. **Please note that restarting the Access application is necessary to apply these changes.** If the Private Key and Certificate do not match, it will result in a "parameter error" message.

- Request URL: `/api/v1/developer/api_server/certificates`
- Permission Key: `edit:api_server`
- Method: `POST`
- UniFi Access Requirement: `Version 2.2.10 or later`

### Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

### Request Body

| Parameter | Required | Type | Description | Example |
| --- | --- | --- | --- | --- |
| key | T | File | Private Key | server.key |
| cert | T | File | Certificate | server.crt |

### Response Sample

```
{
    "code": "SUCCESS",
    "data": "success",
    "msg": "success"
}
```

### Request Sample

The request body should be a JSON object containing the following fields:

```
curl -XPOST
'{{host}}/api/v1/developer/api_server/certificates' \
--H 'Authorization: Bearer wHFmHR******kD6wHg/yg' \
--form 'key=@"/server.key"' \
--form 'cert=@"/server.crt"'
```

# 12.2 Delete HTTPS Certificate

This API allows you to delete the HTTPS certificate for the Access API, applicable only to port **12445**. **Please note that restarting the Access application is necessary to apply these changes.**

- Request URL: `/api/v1/developer/api_server/certificates`
- Permission Key: `edit:api_server`
- Method: `DELETE`
- UniFi Access Requirement: `Version 2.2.10 or later`

## Request Header

| Parameter | Required | Type | Description |
| --- | --- | --- | --- |
| Authorization | T | String | Token required for authentication and access control. |

## Response Sample

```
{
    "code": "SUCCESS",
    "data": "success",
    "msg": "success"
}
```

## Request Sample

The request body should be a JSON object containing the following fields:

```
curl -XDELETE
'{{host}}/api/v1/developer/api_server/certificates' \
--H 'Authorization: Bearer wHFmHR******kD6wHg/yg'
```

# 13. Change Logs

## V3.1.30

- Supports webhooks for custom header configuration and forwarding (#11.3)
- Supports emergency event webhook notifications (#11.2)
- Supports DPS status change webhook notifications for EAH8 and UA-Hub-Gate (#11.2)
- Supports the user deletion API (#3.23)
- Supports the user search API (#3.24)
- Supports the NFC card update API (#6.10)
- Supports temporary unlock for EAH8, UA-Hub-Door-Mini, and UA-Ultra. Requires the latest device versions. (#7.10-#7.11)

## V2.2.6

- Supports user groups and user group policies. (#3.11-#3.22,#10.5-#10.6)
- Supports remote door unlock for Access Ultra (UA-Ultra). (#7.9)

## V2.2.10

- Supports webhook API (#11.2-#11.7)
- Supports API server certificates (#12.1-#12.2)