



**Hướng dẫn tích hợp Object Metadata Search  
và giám sát tài nguyên cho dịch vụ  
Viettel Cloud Object Storage (v1.0)**

## MỤC LỤC

<b>1. Giới thiệu chung</b> .....	<b>1</b>
<i>1.1 Làm việc với Object Metadata</i> .....	<i>1</i>
<i>1.2 Trường hợp sử dụng (Usecases)</i> .....	<i>1</i>
<i>1.3 Tổng quan các APIs</i> .....	<i>2</i>
<b>2 Chi tiết APIs</b> .....	<b>2</b>
<i>Xác thực</i> .....	<i>2</i>
<i>Response schema</i> .....	<i>2</i>
<i>2.1 POST /authentication/login</i> .....	<i>3</i>
<i>2.2 GET /usage/monitoring</i> .....	<i>4</i>
<i>2.3 GET /usage/report</i> .....	<i>5</i>
<i>2.4 GET /metadata/tags/keys/</i> .....	<i>7</i>
<i>2.5 POST /metadata/search/object/tags/</i> .....	<i>8</i>
<i>2.6 POST /metadata/search/object/name/</i> .....	<i>11</i>
<b>3 Một số lưu ý khi sử dụng dịch vụ</b> .....	<b>13</b>

## 1. Giới thiệu chung

### 1.1 Làm việc với Object Metadata

Viettel Cloud Object Storage là công nghệ lưu trữ dữ liệu tối ưu được vận hành dựa trên nền tảng hạ tầng tiên tiến. Chúng tạo ra không gian lưu trữ rộng lớn không giới hạn, đảm bảo tính bảo mật cao và tiết kiệm được chi phí tối đa.

Theo thời gian, số lượng object tăng lên đáng kể, việc theo dõi hiện trạng sử dụng tài nguyên và tìm kiếm một đối tượng cụ thể trở nên khó khăn hơn. Các đối tượng trong S3 chứa metadata duy nhất xác định chính xác đối tượng cùng với các thuộc tính của chúng. Viettel IDC cung cấp API mở rộng cho phép khách hàng giám sát hiệu năng sử dụng và tìm kiếm đối tượng thông qua metadata.

Khách hàng có thể đặt S3 Object metadata tại thời điểm upload đối tượng lên. Metadata là một tập hợp các cặp key-value. Sau khi tải đối tượng lên, cách duy nhất để sửa đổi metadata là tạo một bản sao của đối tượng. [Hướng dẫn upload object](https://docs.aws.amazon.com/AmazonS3/latest/userguide/upload-objects.html): <https://docs.aws.amazon.com/AmazonS3/latest/userguide/upload-objects.html>

Có hai loại metadata trong S3 Object Storage: Metadata do hệ thống xác định và metadata do người dùng xác định. [Xem thêm](https://docs.aws.amazon.com/AmazonS3/latest/userguide/UsingMetadata.html):

<https://docs.aws.amazon.com/AmazonS3/latest/userguide/UsingMetadata.html>

- Metadata do hệ thống xác định: Đối với mỗi đối tượng được lưu trữ trong một bucket, S3 duy trì một bộ metadata hệ thống. S3 xử lý metadata hệ thống này khi cần. metadata như ngày tạo đối tượng được hệ thống kiểm soát, trong đó chỉ S3 mới có thể sửa đổi giá trị.
- Metadata do người dùng xác định: Khi tải lên một object, khách hàng cũng có thể gán metadata cho đối tượng. Khách hàng cung cấp thông tin tùy chọn này dưới dạng cặp key-value (khóa-giá trị) khi gửi yêu cầu PUT hoặc POST để tạo đối tượng. Tên metadata tùy chọn do người dùng xác định phải bắt đầu bằng "x-amz-meta-" để phân biệt chúng với các tiêu đề HTTP khác.

Viettel IDC cung cấp tính năng tìm kiếm object theo tên object, metadata key-value do người dùng xác định.

### 1.2 Trường hợp sử dụng (Usecases)

- Khách hàng chủ động giám sát hiện trạng sử dụng tài nguyên theo thời gian thực, từ đó xây dựng kịch bản cảnh báo riêng.

- Tích hợp báo cáo hiệu năng sử dụng tài nguyên lên dashboard quản trị riêng, trích xuất file báo cáo của khách hàng dễ dàng với API mở rộng.
- Tìm kiếm nhanh chóng các đối tượng phù hợp với nội dung được gán trong thẻ. Trường hợp sử dụng để bắt gặp trong thực tế là tìm kiếm các đối tượng media trong một kho dữ liệu lớn như phim ảnh, bài hát,... theo thẻ loại, năm phát hành, diễn viên, nghệ sĩ thể hiện, hoặc theo bất kỳ thẻ nào đã được gán nhãn trước đó.

### 1.3 Tổng quan các APIs

Phiên bản: v1.0			
Mô tả	Endpoint	Gói standard	Gói cold
Xác thực người dùng, nhận token	/api/v1/authentication/login	✓	✓
Giám sát hiện trạng sử dụng tài nguyên	/api/v1/storage/usage/monitoring/{id}	✓	✓
Báo cáo hiệu năng sử dụng tài nguyên	/api/v1/storage/usage/report/{id}	✓	-
Liệt kê metadata keys	/api/v1/storage/keys/{id}	✓	-
Tìm kiếm objects theo metadata	/api/v1/storage/metadata/search/object/tags/{id}	✓	-
Tìm kiếm objects theo tên object	/api/v1/storage/metadata/search/object/name/{id}	✓	-

## 2 Chi tiết APIs

### Xác thực:

Yêu cầu xác thực cho tất cả API được đính kèm trong request header bao gồm:

Tiêu đề	Giá trị
Token	Giá trị Token trả về trong body khi gọi login thành công

### Response schema:

Response	Response body
Thành công	<pre>{   "Success": true,   "Message": null,   "Data": {     // Kết quả   } }</pre>

Thất bại	<pre>{   "Success": false,   "Message": "chi tiết lỗi",   "Data": null }</pre>
----------	--

## 2.1 POST /authentication/login

Đăng nhập xác thực người dùng, nhận Token

**Url:** POST /api/v1/authentication/login

### Ví dụ sử dụng cURL:

```
curl --location --
request POST 'https://platform.viettelidc.com.vn/api/v1/authentication/login' \
--header 'Content-Type: application/json' \
--data-raw '{
  "UserName": "idcportaluser@hostmail.com",
  "Password": "*****"
}'
```

### Thành phần request :

*UserName* (bắt buộc, kiểu string) là email đăng nhập trên portal Viettel IDC tại: <https://viettelidc.com.vn>

*Password* (bắt buộc, kiểu string)

### Thành phần response:

```
{
  "Success": true,
  "Message": null,
  "Data": {
    "Token": "sdP6KKRfIZFCSpFJ5VVqU/PFwY8yf....."
  }
}
```

Trong đó:

*Success* (kiểu boolean): Kết quả hành động xác thực user

*Message* (kiểu string): Mã thông báo có ý nghĩa trong trường hợp xác thực lỗi

*Token* (kiểu string): Mã token đại diện cho tài khoản khách hàng dùng để xác thực cho mỗi lần gọi API.

Lưu ý: Token có hiệu lực trong 30 phút.

## 2.2 GET /usage/monitoring

Giám sát hiện trạng sử dụng tài nguyên thực tế

**Url:** GET /api/v1/storage/usage/monitoring/{id}

**Ví dụ sử dụng cURL:**

```
curl --location -g --
request GET 'https://platform.viettelidc.com.vn/api/v1/storage/usage/monitoring/{id}' \
--header 'Token: {token}'
```

**Thành phần request :**

*Id* (bắt buộc, kiểu số nguyên) là id gói dịch vụ

**Thành phần response:**

```
{
  "Success": true,
  "Message": null,
  "Data": {
    "capacity": 1100,
    "request": 27000000,
    "capacityUsed": 768.5983946397900581,
    "sizeUnit": "GB",
    "objectCount": 21996,
    "requestRate": null,
    "dataTransferIn": 95000,
    "dataTransferOut": 95000
  }
}
```

Trong đó:

*capacity* (kiểu số thực) là dung lượng lưu trữ được cấp phát

*request* (kiểu số nguyên) là số lượng request được cấp phát

*dataTransferIn* (kiểu số nguyên) là dung lượng truyền tải vào được cấp phát

*dataTransferOut* (kiểu số nguyên) là dung lượng truyền tải ra được cấp phát

*capacityUsed* (kiểu số thực) là dung lượng lưu trữ đã sử dụng thực tế

*sizeUnit* (kiểu enum) là đơn vị dung lượng

*objectCount* (kiểu số nguyên) là số lượng object

## 2.3 GET /usage/report

Báo cáo hiệu năng sử dụng tài nguyên trong một khoảng thời gian

**Url:** GET /api/v1/storage/usage/report/{id}

### Ví dụ sử dụng cURL:

```
curl --location -g --
request GET 'https://platform.viettelidc.com.vn/api/v1/storage/usage/report/{id}/?interval=1' \
--header 'Token: {token}'
```

### Thành phần request :

*Id* (bắt buộc, kiểu số nguyên) là id gói dịch vụ

*Interval* (bắt buộc, kiểu số nguyên): khoảng thời gian lấy báo cáo chỉ chấp nhận giá trị trong khoảng [1, 7, 15, 31, 185, 366] tương ứng với 1 ngày, 1 tuần, nửa tháng, 1 tháng, 6 tháng, 1 năm.

### Thành phần response:

```
{
  "Success": true,
  "Message": null,
  "Data": {
    "StorageByte": [
      {
        "groupId": "user-group",
        "userId": "*",
        "operation": "SB",
        "timestamp": "1674964800000",
        "value": "44334792869171",
        "count": "1",
        "maxValue": "44485390083438",
        "averageValue": "44334792869171"
      }
    ],
    "StorageObject": [
      {
        "groupId": "user-group",
        "userId": "*",
        "operation": "SO",
        "timestamp": "1674964800000",
        "value": "94386770",

```

```

        "count": "1",
        "maxValue": "94663613",
        "averageValue": "94386770"
    }
],
"HttpGet": [
    {
        "groupId": "user-group",
        "userId": "*",
        "operation": "HG",
        "timestamp": "1674964800000",
        "value": "1937167074",
        "count": "4280",
        "maxValue": "1758719",
        "averageValue": "452609"
    }
],
"HttpPut": [
    {
        "groupId": "user-group",
        "userId": "*",
        "operation": "HP",
        "timestamp": "1674964800000",
        "value": "298037365101",
        "count": "570887",
        "maxValue": "2621044",
        "averageValue": "522060"
    }
]
}
}

```

Trong đó:

*StorageByte* (kiểu mảng):

timestamp (kiểu số nguyên): mốc thời gian lấy mẫu

averageValue (kiểu số nguyên): Tổng dung lượng lưu trữ trung bình, tính bằng byte

maxValue (kiểu số nguyên): Tổng dung lượng lưu trữ lớn nhất, tính bằng byte

*StorageObject* (kiểu mảng):



timestamp (kiểu số nguyên): mốc thời gian lấy mẫu  
averageValue (kiểu số nguyên): Số lượng object trung bình  
maxValue (kiểu số nguyên): Số lượng object lớn nhất

*HttpGet* (kiểu mảng):

timestamp (kiểu số nguyên): mốc thời gian lấy mẫu  
value (kiểu số nguyên): Tổng lưu lượng truyền tải ra (datatransfer out),  
tính bằng byte  
count(kiểu số nguyên): số lượng request GET trong khoảng thời gian  
lấy mẫu

*HttpPut* (kiểu mảng):

timestamp(kiểu số nguyên): mốc thời gian lấy mẫu  
value (kiểu số nguyên): Tổng lưu lượng truyền tải vào (datatransfer  
in), tính bằng byte  
count (kiểu số nguyên): số lượng request PUT/POST trong khoảng  
thời gian lấy mẫu

## 2.4 GET /metadata/tags/keys/

Liệt kê metadata keys theo bucket từ 10.000 objects đầu tiên

**Url:** GET /api/v1/storage/metadata/tags/keys/{id}

**Ví dụ sử dụng cURL:**

```
curl --location -g --  
request GET 'https://platform.viettelidc.com.vn/api/v1/storage/metadata/tags/keys/{id}' \  
'?bucketName={bucket}' \  
--header 'Token: {token}'
```

### Thành phần request

*Id* (bắt buộc, kiểu số nguyên) là id gói dịch vụ

*bucketName* (bắt buộc, kiểu string): tên bucket

### Thành phần response

```
{  
  "Success": true,  
  "Message": null,  
  "Data": [  
    {  
      "timestamp": 1511111111,  
      "averageValue": 10,  
      "maxValue": 100,  
      "minValue": 10,  
      "count": 100  
    }  
  ]  
}
```

```
"sha256",
"s3b-last-modified",
"album",
"description",
"local"
]
}
```

Trong đó:

*Data* (kiểu mảng): danh sách keys có trong bucket

*sha256, s3b-last-modified*: các keys của hệ thống định nghĩa

*album, description, local*: các keys định nghĩa bởi người dùng

## 2.5 POST /metadata/search/object/tags/

Cho phép người dùng tìm kiếm chính xác và nhanh chóng các object theo các trường metadata đã được thêm.

**Url:** POST /api/v1/storage/metadata/search/object/tags/{id}

**Ví dụ sử dụng cURL:** ví dụ sau đây tìm kiếm 100 objects đầu tiên thuộc bucket có tên “tailieu1234” có metadata thỏa mãn tất cả điều kiện sau: Nhãn “local” có chứa chính xác cụm từ “Ha Noi” và nhãn “album” có chứa chính xác cả hai cụm từ “Love” và “Rose” không phân biệt hoa thường.

```
url --location -g --
request POST 'https://platform.viettelidc.com.vn/api/v1/storage/metadata/search/object
/tags/{id}' \
--header 'Token: {token}' \
--header 'Content-Type: application/json' \
--data-raw '{
  "bucketName": "tailieu1234",
  "tags": {
    "local": [
      "Ha Noi"
    ],
    "album": [
      "Love",
      "Rose"
    ]
  },
  "from": 0,
  "size": 100,
```

```
"keySort": "asc"
}'
```

### Thành phần request

*Id* (bắt buộc, kiểu số nguyên) là id gói dịch vụ

*bucketName* (bắt buộc, kiểu string): tên bucket

*Key* (bắt buộc, kiểu string): tên khóa (nhãn), phân biệt hoa thường.

Lưu ý: Key là phần chuỗi kí tự rút gọn sau tiền tố “x-amz-meta-“

*Value* (bắt buộc, kiểu mảng string): mảng các giá trị, không phân biệt hoa thường

*from*, *size* (tùy chọn, kiểu số nguyên): số lượng *size* kết quả trả về phù hợp với điều kiện, bắt đầu từ *from*. *From* nhận giá trị mặc định là 0, *Size* mặc định/tối đa là 1000.

VD: *from*: 0, *size*: 5 thì sẽ trả về 5 dữ liệu đầu tiên phù hợp với điều kiện

*keySort* (tùy chọn, kiểu enum): nhận 2 giá trị “asc” và “desc” sắp xếp theo *bucketName*

### Thành phần response

```
{
  "Success": true,
  "Message": null,
  "Data": {
    "total": {
      "value": 2,
      "relation": "eq"
    },
    "mappingResult": [
      {
        "key": "tailieu1234%2Fanhcuoi.pdf",
        "bucketName": "tailieu1234",
        "objectName": "anhcuoi.pdf",
        "lmt": "2023-01-13T08:45:19.389Z",
        "objectSize": 1935515,
        "contentType": "application/pdf",
        "esTime": "Fri Jan 13 17:13:05 ICT 2023",
        "userMetadata": {
          "sha256": "06993fa047dffad0c2285594cb9b7df9835a619c51e962a3f6480424a5729379",
          "album": "The rose for love, I love you",

```

```

        "description": "Anh cưới ngay đẹp trời",
        "local": "85 Vu Trong Phung, Thanh Xuan, Ha Noi",
        "s3b-last-modified": "20220607T071954Z"
    }
},
{
    "key": "tailieu1234%2Ftailieu.pdf",
    "bucketName": "tailieu1234",
    "objectName": "tailieu.pdf",
    "lmt": "2023-01-19T02:46:19.319Z",
    "objectSize": 19355445,
    "contentType": "application/pdf",
    "esTime": "Fri Jan 19 11:11:15 ICT 2023",
    "userMetadata": {
        "sha256": "09643fa047dffad0c2285594cb9b7df9835a619c51e962a3f6480
424b3654279",
        "album": "all love for the Rose",
        "description": "tai lieu 1000 bi kip giu hanh phuc gia dinh",
        "local": "56 Yen Hoa, Yen Hoa, Ha Noi",
        "s3b-last-modified": "20220622T071954Z"
    }
}
],
"from": 0,
"size": 100
}
}

```

Trong đó:

*total* (kiểu mảng):

value (kiểu số nguyên): tổng số object phù hợp

relation (kiểu enum): cho biết số lượng tài liệu phù hợp trong tham số

value là chính xác(eq) hay giới hạn dưới(gte)

*mappingResult* (kiểu mảng): bao gồm các object phù hợp

key (kiểu string): url encode của bucketName/objectName

bucketName (kiểu string): tên bucket

objectName (kiểu string): tên object

lmt (kiểu string): mốc thời gian object được tải lên

objectSize (kiểu số nguyên): dung lượng object. Tính bằng byte

contentType (kiểu string): kiểu định dạng object

esTime (kiểu string): thời gian thực hiện truy vấn dữ liệu

userMetadata.sha256 userMetadata.s3b-last-modified (kiểu string):

các keys định nghĩa bởi hệ thống

userMetadata.album, userMetadata.description, userMetadata.local

(kiểu string): các keys định nghĩa bởi người dùng, chứa thông tin có thể tìm kiếm.

*from, size* (tùy chọn, kiểu số nguyên): số lượng *size* kết quả trả về phù hợp với điều kiện, bắt đầu từ *from*. *From* nhận giá trị mặc định là 0, *Size* mặc định/tối đa là 1000.

## 2.6 POST /metadata/search/object/name/

Cho phép người dùng tìm các object theo tên của chúng.

**Url:** POST /api/v1/storage/metadata/search/object/name/{id}

**Ví dụ sử dụng cURL:** ví dụ sau đây tìm kiếm 100 objects đầu tiên thuộc bucket “tailieu1234” và trong tên object có chứa “tailieu” và “pdf” không phân biệt hoa thường.

```
url --location -g --
request POST 'https://platform.viettelidc.com.vn/api/v1/storage/metadata/search/object
/name/{id}' \
--header 'Token: {token}' \
--header 'Content-Type: application/json' \
--data-raw '{
  "bucketName": "tailieu1234",
  "objectName": [
    "tailieu",
    "pdf"
  ],
  "from": 0,
  "size": 100,
  "keySort": "asc"
}'
```

### Thành phần request

*Id* (bắt buộc, kiểu số nguyên) là id gói dịch vụ

*bucketName* (bắt buộc, kiểu string): tên bucket

*objectName* (bắt buộc, kiểu mảng string): giá trị tìm kiếm, không phân biệt hoa thường.

Lưu ý: *objectName* chấp nhận giá trị UTF-8. Có nghĩa là “báo cáo.pdf” và “bao cao.pdf” sẽ trả về kết quả khác nhau.

*from*, *size* (tùy chọn, kiểu số nguyên): số lượng *size* kết quả trả về phù hợp với điều kiện, bắt đầu từ *from*. *From* nhận giá trị mặc định là 0, *Size* mặc định/tối đa là 1000.

VD: *from*: 0, *size*: 5 thì sẽ trả về 5 dữ liệu đầu tiên phù hợp với điều kiện

*keySort* (tùy chọn, kiểu enum): nhận 2 giá trị “asc” và “desc” sắp xếp theo *bucketName*

### Thành phần response

Response tương tự mục 2.5. Ví dụ kết quả trả về 02 objects phù hợp có tên “**tailieu.pdf**” và “**tailieu/mau hop dong.pdf**”:

```
{
  "Success": true,
  "Message": null,
  "Data": {
    "total": {
      "value": 2,
      "relation": "eq"
    },
    "mappingResult": [
      {
        "key": "tailieu1234%2Ftailieu.pdf",
        "bucketName": "tailieu1234",
        "objectName": "tailieu.pdf",
        "lmt": "2023-01-19T02:46:19.319Z",
        "objectSize": 19355445,
        "contentType": "application/pdf",
        "esTime": "Fri Jan 19 11:11:15 ICT 2023",
        "userMetadata": {
          "sha256": "09643fa047dffad0c2285594cb9b7df9835a619c51e962a3f6480424b3654279",
          "album": "all love for the Rose",
          "description": "tai lieu 1000 bi kip giu hanh phuc gia dinh",
          "local": "56 Yen Hoa, Yen Hoa, Ha Noi",
          "s3b-last-modified": "20220622T071954Z"
        }
      }
    ]
  }
}
```

```

    }
  },
  {
    "key": "tailieu1234%2Ftailieu%2Fmauhopdong.pdf",
    "bucketName": "tailieu1234",
    "objectName": "tailieu/mau hop dong.pdf",
    "lmt": "2023-01-20T06:16:13.132Z",
    "objectSize": 4324324,
    "contentType": "application/pdf",
    "esTime": "Fri Jan 23 06:16:13 ICT 2023",
    "userMetadata": {
      "sha256": "09643fa047dffad0c2285594cb9b7df9835a619c51e962a3f6480
424b3654279",
      "local": "85 Vu Trong Phung, Thanh Xuan, Ha Noi",
      "s3b-last-modified": "20230120T071954Z"
    }
  },
],
"from": 0,
"size": 100
}
}

```

### 3 Một số lưu ý khi sử dụng dịch vụ

- Dung lượng metadata sẽ được tính vào dung lượng lưu trữ của gói dịch vụ. Tiêu đề (Header) request PUT được giới hạn ở kích thước 8 KB. Trong đó metadata do người dùng định nghĩa có kích thước giới hạn ở 2 KB. Kích thước của metadata do người dùng định nghĩa được tính bằng cách lấy tổng số byte trong mã hóa US-ASCII của từng key và value.
- Metadata key, value, tag có thể chấp nhận sử dụng để gắn nhãn các đối tượng chứa dữ liệu bí mật như thông tin nhận dạng cá nhân, mật khẩu, tài khoản ngân hàng,... Tuy nhiên, bản thân chúng không được chứa bất kỳ thông tin bí mật nào.
- Dữ liệu thực tế API mở rộng trả về như báo cáo tài nguyên sử dụng, search object theo metadata, list keys... sẽ được đồng bộ với dữ liệu lưu trữ của khách hàng liên tục muộn nhất sau 1 giờ