HTTPDNS

API Documentation

Product Documentation

Contents

API Documentation
Configuration Information Description

Last updated：2022-06-22 15:56:39

Overview

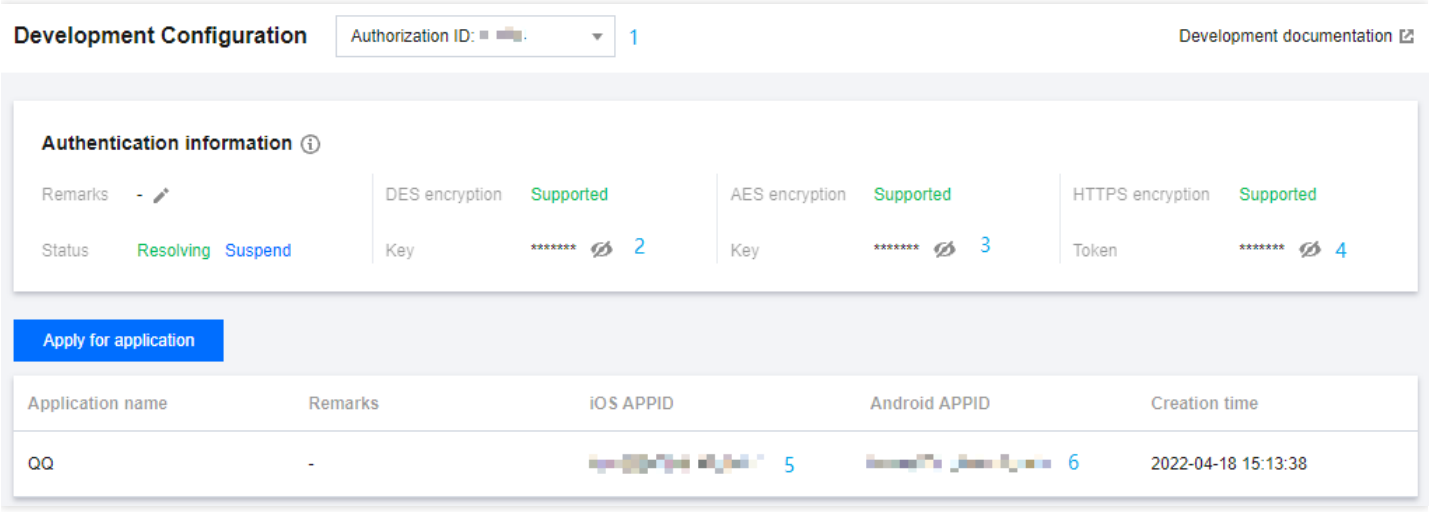
This document describes how to get the configuration information in the HTTPDNS console and what information items mean. You need to get such configuration information before you can connect to HTTPDNS.

Prerequisites

You have activated HTTPDNS as instructed in Activating HTTPDNS.

Operation Guide

Log in to the **Development Configuration** page in the HTTPDNS console to query your configuration information.



- **Authorization ID**: It is the unique ID of a development configuration used in HTTPDNS, i.e., the authorization ID parameter passed in when you call the HTTP query API `http://43.132.55.55` of HTTPDNS.

- **DES encryption key**: The key used to encrypt the DNS request data when you call the HTTP DNS API `http://43.132.55.55` of HTTPDNS with DES encryption used.
- **AES encryption key**: The key used to encrypt the DNS request data when you call the HTTP DNS API `http://43.132.55.55` of HTTPDNS with AES encryption used.
- **HTTPS encryption token**: The token used to authenticate the DNS request data when you call the HTTPS DNS API `https://43.132.55.56` of HTTPDNS.

> Note：
>
> If the following two items are not displayed in the console, **request an application** first to view them. For detailed directions, see SDK Activation Process.

- **iOS APPID**: The `appId (application ID)` authentication information for using the SDK for iOS provided by HTTPDNS.
- **Android APPID**: The `business appkey` authentication information for using the SDK for Android provided by HTTPDNS.

Querying with HTTP Request Methods

Last updated：2022-06-22 15:57:34

Overview

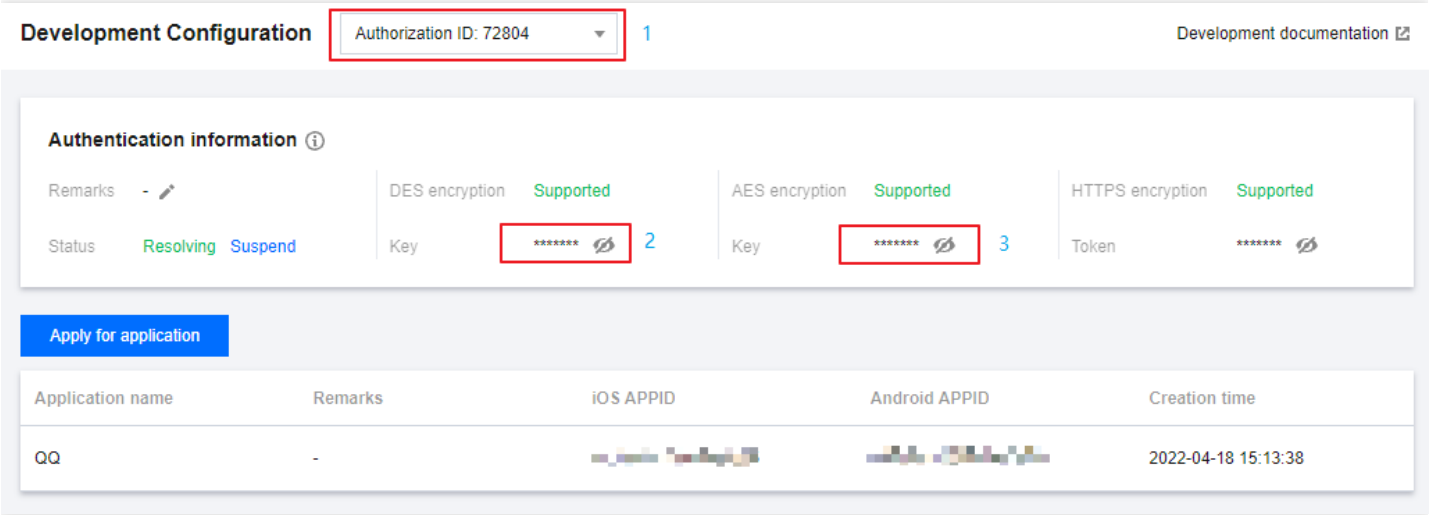
HTTPDNS provides DNS services through HTTP and HTTPS APIs. The services are accessed directly via IP addresses. Multiple service IPs are available. The following takes the query entry `43.132.55.55` for HTTP request as an example.

Note：

- **Currently, only the DES encryption method is available (service IP: `43.132.55.55`). HTTPS and AES encryption methods are not available.**
- After activating HTTPDNS, you need to first add a domain to be resolved in the HTTPDNS console as instructed in Adding a Domain.
- We provide two sample entry IPs: `43.132.55.55` for HTTP and `43.132.55.56` for HTTPS.
- Use the official SDK preferably. If the SDK cannot be used in special scenarios, you need to directly access the HTTP API. In this case, please submit a ticket to contact us, and we will provide you with multiple service IPs and applicable security suggestions according to your specific use case.
- For considerations of security risks such as service IP attacks, in order to ensure service availability, HTTPDNS provides multiple service IPs at the same time. When an IP is unavailable under abnormal conditions, you can retry with other IPs.

Preparations

When using the request API `http://43.132.55.55/d? + {request parameters}`, you need to use the following configuration information, which can be obtained on the **Development Configuration** page in the HTTPDNS console:



- **Authorization ID**: It is the unique ID of a development configuration used in HTTPDNS, i.e., the authorization ID parameter passed in when you call the HTTP query API `http://43.132.55.55` of HTTPDNS.
- **DES encryption key**: The key used to encrypt the DNS request data when you call the HTTP DNS API `http://43.132.55.55` of HTTPDNS with DES encryption used.
- **AES encryption key**: The key used to encrypt the DNS request data when you call the HTTP DNS API `http://43.132.55.55` of HTTPDNS with AES encryption used.

API Description

- API request address: `http://43.132.55.55/d? + {request parameters}` .
- Request method: POST or GET.
- For considerations of security risks such as service IP attacks, in order to ensure service availability, we provide multiple service IPs at the same time. If you want to directly request the HTTPDNS service through APIs, please submit a ticket to contact us, and we will provide you with multiple service IPs and applicable security suggestions according to your specific use case.
- Entry IP switch logic: When the connected IP access times out, the returned result is not in IP format, or the response is empty, use another entry IP for access. If all IPs are abnormal, use local DNS for DNS queries.

Request Parameters

Parameter	Description	Required	Value	Encryption	Description

Parameter	Description	Required	Value	Encryption	Description
dn	Queried domain	Yes	The length of a single domain before encryption is 253	Yes	It must be a domain ad in the HTTPDNS consc the form of encrypted s for transfer. • For how to add a do see Adding a Domai • For more informatior encryption, see AES Encryption/Decryptic
id	User ID	Yes	1–10000	No	If you use AES or DES encryption, you must pa the ID but don't need to encrypt it.
alg	Algorithm	Yes	[aes/des]	No	The DES algorithm is u by default. Different algorithms have differer keys.
ip	ECS (EDNS-Client-Subnet) value of the DNS request	No	IPv4/IPv6 address value	Yes	By default, the HTTPDI server will query the cli egress IP in order to qu the IP for the DNS split zone. You can use the `ip=xxx` parameter to sp the split zone's IP addr You can pass in IPv4/IF addresses, which will b automatically identified the API. For more information on encrypti see AES/DES Encryption/Decryption.
query	Queried domain returned in the result	No	1	No	For single-domain quer this parameter requires returned result to carry queried domain.
timeout	Timeout period	No	1000–5000 ms	No	It is the query timeout period, which is 5 seco by default. Value range [1000, 5000] ms

Parameter	Description	Required	Value	Encryption	Description
ttl	Specifies whether to return the TTL value in the query result	No	1	No	If this parameter is not carried, the TTL value not be passed by defau Valid value: 1
type	Query type	No	[aaaa/AAAA/addrs/ADDRS]	No	Valid values: [aaaa,AAAA,addrs,ADI The A record will be qu by default. If AAAA/aaa set, the AAAA record w queried; if addrs/ADDR set, both the A and AA records will be queried.
clientip	Client IP address returned in the query result	No	1	No	Valid value: 1. If this parameter is not carrie clientip value will not be passed by default. If a is assigned to this parameter, the address value will be after the | symbol in the returned result. If the ip paramet carried, the value of the parameter will be retur otherwise, the client IP address will be returne

Note：

The ECS (EDNS-Client-Subnet) protocol adds the IP address of the user requesting DNS in the DNS request packet, based on which the DNS server can return a server IP address for quicker access by the user.

Request Description

The ID `xxx` is used as an example below.

Note：

- The following samples are for AES/DES encryption, where both the domain and IP parameter need to be encrypted. For example, the domain `cloud.tencent.com` needs to be encrypted, while the authorization ID doesn't.
- If HTTPDNS does not find the DNS query result, it will return null.
- HTTPDNS has been connected to BGP Anycast to implement multi-region cross-IDC disaster recovery. However, to guarantee a higher service quality, we recommend you use the failover policy for connection.

Requesting A record

- **Sample input:**

```
curl "http://43.132.55.55/d?dn={encrypted string of cloud.tencent.com}&id=xxx"
```

- **Decrypted response format:**

```
2.3.3.4;2.3.3.5;2.3.3.6
```

- **Format description**: Multiple returned query results are separated by semicolon.

Carrying TTL information in returned result

- **Sample input:**

```
curl "http://43.132.55.55/d?dn={encrypted string of cloud.tencent.com}&id=xxx&ttl=1"
```

- **Decrypted response format:**

```
2.3.3.4;2.3.3.5;2.3.3.6,120
```

- **Format description**: Multiple returned query results are separated by semicolon. The record values and TTL value are separated by comma.

Carrying the IP address of query split zone in returned result

- **Sample input:**

```
curl "http://43.132.55.55/d?dn={encrypted string of cloud.tencent.com}&id=xxx&clientip=1&ip={encrypted string of the ECS value of the DNS request}&ttl=1"
```

- **Decrypted response format:**

```
12.3.3.4;2.3.3.5;2.3.3.6,120|1.2.3.4
```

- **Format description**: The returned result carries the split zone's IP address separated by '|'. If the "ip=xxx" parameter is not passed in, the egress IP address will be returned; otherwise, the address in the `ip` parameter will be returned.

Requesting A and AAAA records at the same time

- **Sample input:**

```
curl "http://43.132.55.55/d?dn={encrypted string of cloud.tencent.com}&id=xxx&clientip=1&ip={encrypted string of the ECS value of the DNS request}&type=addrs&ttl=1"
```

- **Decrypted response format:**

```
2.3.3.4;2.3.3.5;2.3.3.6,120-2402:4e00:0123:4567:0::2345;2403:4e00:0123:4567:0::2346,120|1.2.3.4
```

- **Format description**: The A record is followed by a hyphen and then the AAAA record.

Carrying queried domain in returned result

- **Sample input:**

```
curl "http://43.132.55.55/d?dn={encrypted string of cloud.tencent.com}&id=xxx&clientip=1&ip={encrypted string of the ECS value of the DNS request}&query=1&ttl=1"
```

- **Decrypted response format:**

```
cloud.tencent.com.:2.3.3.4;2.3.3.5;2.3.3.6,120|1.2.3.4
```

- **Format description**: The response is in the format of "domain.:result".

Batch querying domains

- **Sample input:**

```
curl "http://43.132.55.55/d?dn={encrypted string of cloud.tencent.com, www.qq.com, and www.dnspod.cn}&id=xxx&clientip=1&ip={encrypted string of the ECS value of the DNS request}&ttl=1"
```

- **Decrypted response format:**

```
cloud.tencent.com.:2.3.3.4;2.3.3.5;2.3.3.6,120
www.qq.com.:3.3.3.4;3.3.3.5;3.3.3.6,180
www.dnspod.cn.:4.3.3.4;4.3.3.5;4.3.3.6,60|1.2.3.4
```

- **Format description:** The returned result of multiple domains are separated by line break, with the IP addresses appended at the end of all record values.

Description of Request Error or No Record

Note：

- The following samples are for AES/DES encryption, where both the domain and IP parameter need to be encrypted. For example, the domain `cloud.tencent.com` needs to be encrypted, while the authorization ID doesn't.
- If you use HTTPS, you must change the request address to `43.132.55.56` and pass in the token.

Querying A record

- **Sample input:**

```
curl "http://43.132.55.55/d?dn={encrypted string of cloud.tencent.com}&id=xxx"
```

- **Decrypted response format:** Empty.
- **Format description:** If there are no records, an empty string will be returned.

Carrying domain in returned result

- **Sample input:**

```
curl "http://43.132.55.55/d?dn={encrypted string of cloud.tencent.com}&id=xxx&type=addrs&query=1&ip={encrypted string of the ECS value of the DNS request}"
```

- **Decrypted response format:**

```
cloud.tencent.com|1.2.3.4
```

- **Format description:** 0 indicates no records.

Returning A and AAAA records

- **Sample input:**

```
curl "http://43.132.55.55/d?dn={encrypted string of cloud.tencent.com}&id=xxx&type=addrs&query=1&ip={encrypted string of the ECS value of the DNS request}"
```

- **Decrypted response format:**

```
cloud.tencent.com.:0-0|1.2.3.4
```

- **Format description:** 0 indicates no records. If a record exists, it will be returned in the result. For example, `cloud.tencent.com.:2.3.4.5;3.3.3.3-0|1.2.3.4` indicates that no AAAA records can be found.

Batch querying domains

- **Sample input:**

```
curl "http://43.132.55.55/d?dn={encrypted string of cloud.tencent.com, www.qq.com, and www.dnspod.cn}&id=xxx&clientip=1&ip={encrypted string of the ECS value
```

```
of the DNS request}&ttl=1"
```

- **Decrypted response format**:

```
cloud.tencent.com.:0
www.qq.com.:3.3.3.4;3.3.3.5;3.3.3.6,180
www.dnspod.cn.:4.3.3.4;4.3.3.5;4.3.3.6,60|1.2.3.4
```

- **Format description:** For domains about which no data is found, 0 will be returned. If a record exists, it will be returned in the result.

HTTP Status Codes

The following are the HTTP status codes related to the business logic of the APIs.

Status Code	Description
200 OK	If the API is called correctly, a 200 status code will be returned regardless of whether the query is successful.
404 Not Found	The API does not exist, or the URL actually accesses a resource that does not exist.
429 Too Many Requests	The access requests are too frequent and exceed the limit.
501 Not Implemented	A request method other than "GET" or "POST" is used.

Querying with HTTPS Request Methods

Last updated：2022-06-22 15:58:39

Overview

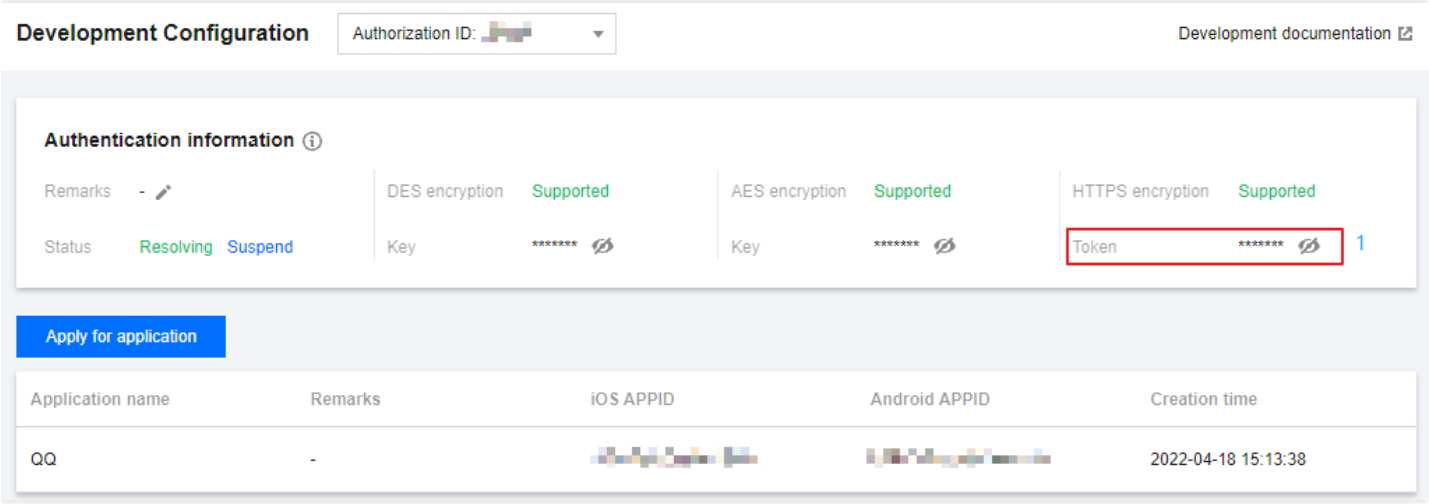
HTTPDNS provides DNS services through HTTP and HTTPS APIs. The services are accessed directly at IP addresses. Multiple service IPs are available. The following takes the query entry `43.132.55.56` for HTTPS request as an example.

Note：

- **Currently, only the HTTP DES encryption method is available (service IP: `43.132.55.55`), while HTTPS and AES encryption methods are not.**
- After activating HTTPDNS, you need to first add a domain to be resolved in the HTTPDNS console as instructed in Adding a Domain.
- We provide two sample entry IPs: `43.132.55.56` for HTTPS and `43.132.55.55` for HTTP.
- Use the official SDK preferably. If the SDK cannot be used in special scenarios, you need to directly access the HTTP API. In this case, please submit a ticket to contact us, and we will provide you with multiple service IPs and applicable security suggestions according to your specific use case.
- For considerations of security risks such as service IP attacks, in order to ensure service availability, HTTPDNS provides multiple service IPs at the same time. When an IP is unavailable under abnormal conditions, you can retry with other IPs.

Preparations

When using the request API `https://43.132.55.56/d? + {request parameters}`, you need to use the following configuration information, which can be obtained on the **Development Configuration** page in the HTTPDNS console:



HTTPS encryption token: The token used to authenticate the DNS request data when you call the HTTPS DNS API `https://43.132.55.56` of HTTPDNS.

API Description

- API request address: `https://43.132.55.56/d? + {request parameters}` .
- Request method: POST or GET.
- For considerations of security risks such as service IP attacks, in order to ensure service availability, we provide multiple service IPs at the same time. If you want to directly request the HTTPDNS service through APIs, please submit a ticket to contact us, and we will provide you with multiple service IPs and applicable security suggestions according to your specific use case.
- Entry IP switch logic: When the connected IP access times out, the returned result is not in IP format, or the response is empty, use another entry IP for access. If all IPs are abnormal, use local DNS for DNS queries.

Request Parameters

Parameter	Description	Required	Value	Encryption	Description
dn	Queried domain	Yes	Strings	No	It must be a domain ad in the HTTPDNS consc For more information, s Adding a Domain.

Parameter	Description	Required	Value	Encryption	Description
token	Flag for using HTTPS	Yes	Integer data	No	For how to get a token, Configuration Informati Description.
ip	ECS (EDNS-Client-Subnet) value of the DNS request	No	IPv4/IPv6 address value	No	By default, the HTTPDI server will query the cli egress IP in order to qu the IP for the DNS split zone. You can use the `ip=xxx` parameter to sp the split zone's IP addr You can pass in IPv4/IP addresses, which will b automatically identified the API.
query	Queried domain returned in the result	No	1	No	For single-domain quer this parameter requires returned result to carry queried domain.
timeout	Timeout period	No	1000–5000 ms	No	It is the query timeout period, which is 5 seco by default. Value range [1000, 5000] ms
ttl	Specifies whether to return the TTL value in the query result	No	1	No	If this parameter is not carried, the TTL value not be passed by defau Valid value: 1
type	Query type	No	[aaaa/AAAA/addrs/ADDRS]	No	Valid values: [aaaa,AAAA,addrs,ADI The A record will be qu by default. If AAAA/aaa set, the AAAA record w queried; if addrs/ADDR set, both the A and AA records will be queried.

Parameter	Description	Required	Value	Encryption	Description
clientip	Client IP address returned in the query result	No	1	No	Valid value: 1. If this parameter is not carrie clientip value will not be passed by default. If a v is assigned to this parameter, the address value will be after the | symbol in the returned result. If the ip paramet carried, the value of the parameter will be returr otherwise, the client IP address will be returne

Note：

- The ECS (EDNS-Client-Subnet) protocol adds the IP address of the user requesting DNS in the DNS request packet, based on which the DNS server can return a server IP address for quicker access by the user.
- If you make a query with an HTTPS request method, the transferred data will be protected through encryption because of the TLS channel, so you don't need to encrypt the data passed in.
- For security and authentication reasons, you need to pass in the HTTPS token.

Request Description

The domain `cloud.tencent.com` and token `yyyy` are used as an example below.

Note：

- If HTTPDNS does not find the DNS query result, it will return null.
- HTTPDNS has been connected to BGP Anycast to implement multi-region cross-IDC disaster recovery. However, to guarantee a higher service quality, we recommend you use the failover policy for connection.

Requesting A record

- **Sample input:**

```
curl "https://43.132.55.56/d?dn=cloud.tencent.com&token=yyyy"
```

- **Decrypted response format:**

```
2.3.3.4;2.3.3.5;2.3.3.6
```

- **Format description**: Multiple returned query results are separated by semicolon.

Carrying TTL information in returned result

- **Sample input:**

```
curl "https://43.132.55.56/d?dn=cloud.tencent.com&token=yyyy&ttl=1"
```

- **Decrypted response format:**

```
2.3.3.4;2.3.3.5;2.3.3.6,120
```

- **Format description**: Multiple returned query results are separated by semicolon. The record values and TTL value are separated by comma.

Carrying the IP address of query split zone in returned result

- **Sample input:**

```
curl "https://43.132.55.56/d?dn=cloud.tencent.com&token=yyyy&clientip=1&ip=1.2.3.4&ttl=1"
```

- **Decrypted response format:**

```
12.3.3.4;2.3.3.5;2.3.3.6,120|1.2.3.4
```

- **Format description**: The returned result carries the split zone's IP address separated by '|'. If the "ip=xxx" parameter is not passed in, the egress IP address will be returned; otherwise, the address in the `ip` parameter

will be returned.

Requesting A and AAAA records at the same time

- **Sample input:**

```
curl "https://43.132.55.56/d?dn=cloud.tencent.com&token=yyyy&clientip=1&ip=1.2.3.4&type=addrs&ttl=1"
```

- **Decrypted response format:**

```
2.3.3.4;2.3.3.5;2.3.3.6,120-2402:4e00:0123:4567:0::2345;2403:4e00:0123:4567:0::2346,120|1.2.3.4
```

- **Format description**: The A record is followed by a hyphen and then the AAAA record.

Carrying queried domain in returned result

- **Sample input:**

```
curl "https://43.132.55.56/d?dn=cloud.tencent.com&token=yyyy&clientip=1&ip=1.2.3.4&query=1&ttl=1"
```

- **Decrypted response format:**

```
cloud.tencent.com.:2.3.3.4;2.3.3.5;2.3.3.6,120|1.2.3.4
```

- **Format description**: The response is in the format of "domain.:result".

Batch querying domains

- **Sample input:**

```
curl "https://43.132.55.56/d?dn=cloud.tencent.com,www.qq.com,www.dnspod.cn&token=yyyy&clientip=1&ip=1.2.3.4&ttl=1"
```

- **Decrypted response format:**

```
cloud.tencent.com.:2.3.3.4;2.3.3.5;2.3.3.6,120
www.qq.com.:3.3.3.4;3.3.3.5;3.3.3.6,180
www.dnspod.cn.:4.3.3.4;4.3.3.5;4.3.3.6,60|1.2.3.4
```

- **Format description:** The returned result of multiple domains are separated by line break, with the IP addresses appended at the end of all record values.

Description of Request Error or No Record

Querying A record

- **Sample input:**

```
curl "https://43.132.55.56/d?dn=cloud.tencent.com&token=yyyy&id=xxx"
```

- **Decrypted response format:** Empty.
- **Format description:** If there are no records, an empty string will be returned.

Carrying domain in returned result

- **Sample input:**

```
curl "https://43.132.55.56/d?dn=cloud.tencent.com&token=yyyy&type=addrs&query=1
&ip=1.2.3.4"
```

- **Decrypted response format:**

```
cloud.tencent.com|1.2.3.4
```

- **Format description:** 0 indicates no records.

Returning A and AAAA records

- **Sample input:**

```
curl "https://43.132.55.56/d?dn=cloud.tencent.com&token=yyyy&type=addrs&query=1
&ip=1.2.3.4"
```

- **Decrypted response format:**

```
cloud.tencent.com.:0-0|1.2.3.4
```

- **Format description:** 0 indicates no records. If a record exists, it will be returned in the result. For example, `cloud.tencent.com.:2.3.4.5;3.3.3.3-0|1.2.3.4` indicates that no AAAA records can be found.

Batch querying domains

- **Sample input:**

```
curl "https://43.132.55.56/d?dn=cloud.tencent.com,www.qq.com,www.dnspod.cn&toke
n=yyyy&clientip=1&ip=1.2.3.4&ttl=1"
```

- **Decrypted response format**:

```
cloud.tencent.com.:0
www.qq.com.:3.3.3.4;3.3.3.5;3.3.3.6,180
www.dnspod.cn.:4.3.3.4;4.3.3.5;4.3.3.6,60|1.2.3.4
```

- **Format description:** For domains about which no data is found, 0 will be returned. If a record exists, it will be returned in the result.

HTTP Status Codes

The following are the HTTP status codes related to the business logic of the APIs.

Status Code	Description
200 OK	If the API is called correctly, a 200 status code will be returned regardless of whether the query is successful.
404 Not Found	The API does not exist, or the URL actually accesses a resource that does not exist.

Status Code	Description
429 Too Many Requests	The access requests are too frequent and exceed the limit.
501 Not Implemented	A request method other than "GET" or "POST" is used.

AES/DES Encryption/Decryption

Last updated：2022-06-22 15:59:32

Overview

This document describes how to use the DES and AES encryption algorithms. They can be used to encrypt the request parameters and decrypt the response data so as to prevent requests in plaintext from being maliciously altered during transfer.

> Note：
>
> If you make a query with an HTTPS request method, the transferred data will be protected through encryption because of the TLS channel, so you don't need to encrypt the data passed in.

Prerequisites

- You have activated HTTPDNS and obtained the configuration information such as authorization ID, encryption key, and HTTPS token. For more information, see Configuration Information Description.
- You have added the domain to be queried in the HTTPDNS console as instructed in Adding a Domain.

Flowchart

Step 1. Determine the encryption method. Currently, HTTP requests to HTTPDNS can be encrypted with DES or AES.

> Note：
>
> - If you make a query with an HTTPS request method, see Querying with HTTPS Request Methods.
> - Encrypt the domain to be resolved with the corresponding key and algorithm (if you want to use the `ip` parameter, you also need to encrypt it) and use the encrypted result and ID (which does not need to be encrypted) as the request parameters.

Step 2. Send an encrypted request.

Step 3. Receive an encrypted response.

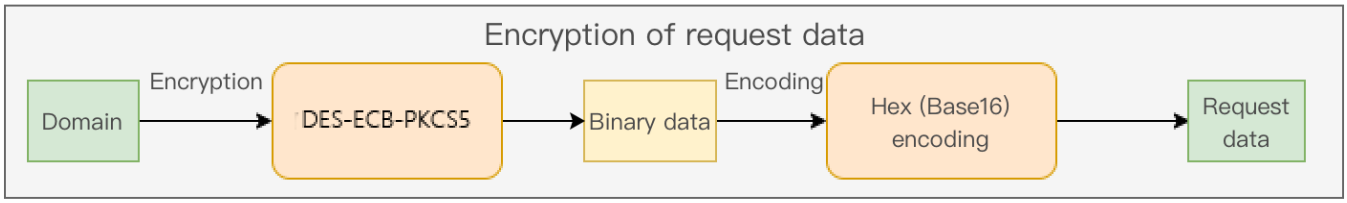
Step 4. Decrypt the result.

Encryption and Decryption Algorithm Use Instructions

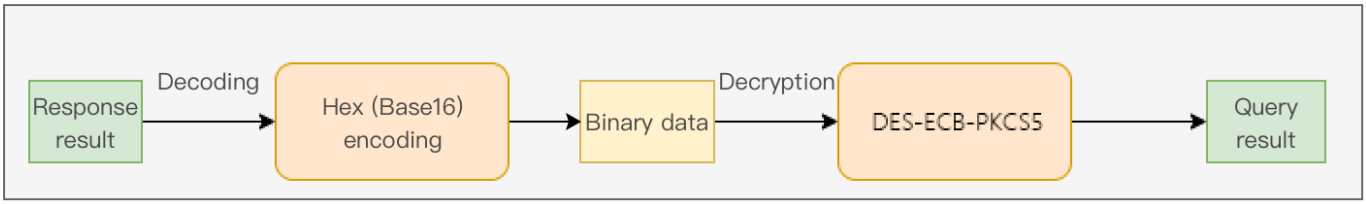
DES algorithm

> Note：
>
> For encryption and decryption with DES, the key is 8 characters in length, the block cipher mode is `ECB`, and the padding algorithm is `PKCS5Padding`.

The encrypted data is encoded by using `Hex(Base16)` to convert the binary data into a visible hexadecimal ID, and the length of the encoded data will double. The detailed process is as shown below:



Decryption of the response data involves decoding the data to binary data with `Hex(Base16)` first and then decrypting the binary data with the DES algorithm into plaintext data. The detailed process is as shown below:



For example, if your domain is `www.dnspod.cn` and the encryption key is `dnspodpass`, the process will be as follows:

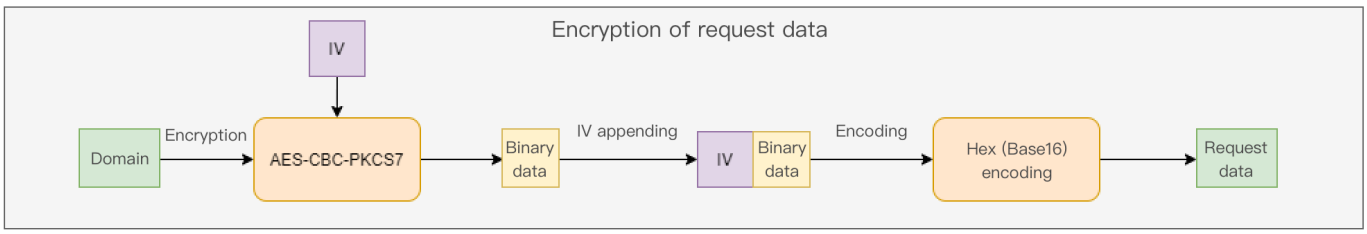
1. Add the domain in the HTTPDNS console.
2. Encrypt the domain with the encryption algorithm `DES-ECB-PKCS5` and DES encryption key `dnspodpass`, and you will get the encrypted string `87ae992c1321f299da3c0210a9900ae7`.
3. Call the `curl "http://43.132.55.55/d?dn=87ae992c1321f299da3c0210a9900ae7&id={authorization ID}"` API to request the A record. You will get an encrypted string with a doubled length, such as `55915a682ea20840ff74aa6e7bebf11454ed0f4050a63e93e6e89521553a01a8`.
4. Decrypt the encrypted string with the encryption algorithm `DES-ECB-PKCS5` and DES encryption key `dnspodpass`, and you will get the plaintext data `121.12.53.35;106.227.19.35`.

> Note：
>
> The above strings are used as an example only and cannot be used for normal requests.

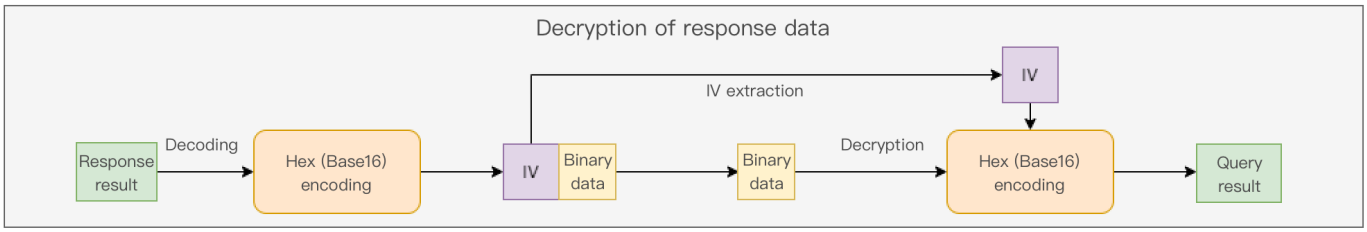
AES algorithm

> Note：
>
> For encryption and decryption with AES, the key is 16 characters in length, the block cipher mode is `CBC`, and the padding algorithm is `PKCS7`.

The CBC mode requires a random `IV` as the initial input for encryption and decryption, so the `IV` will also be carried in the request and response. The encrypted data along with the `IV` is encoded by using `Hex` and converted into a visible hexadecimal ID. The detailed process is as shown below:



During decryption, the data is decoded to binary data by using `Hex`, where the first 16 bytes is the `IV` value, and the bytes after `IV` is the data to be decrypted with the AES algorithm. The plaintext data will be obtained after decryption. The detailed process is as shown below:

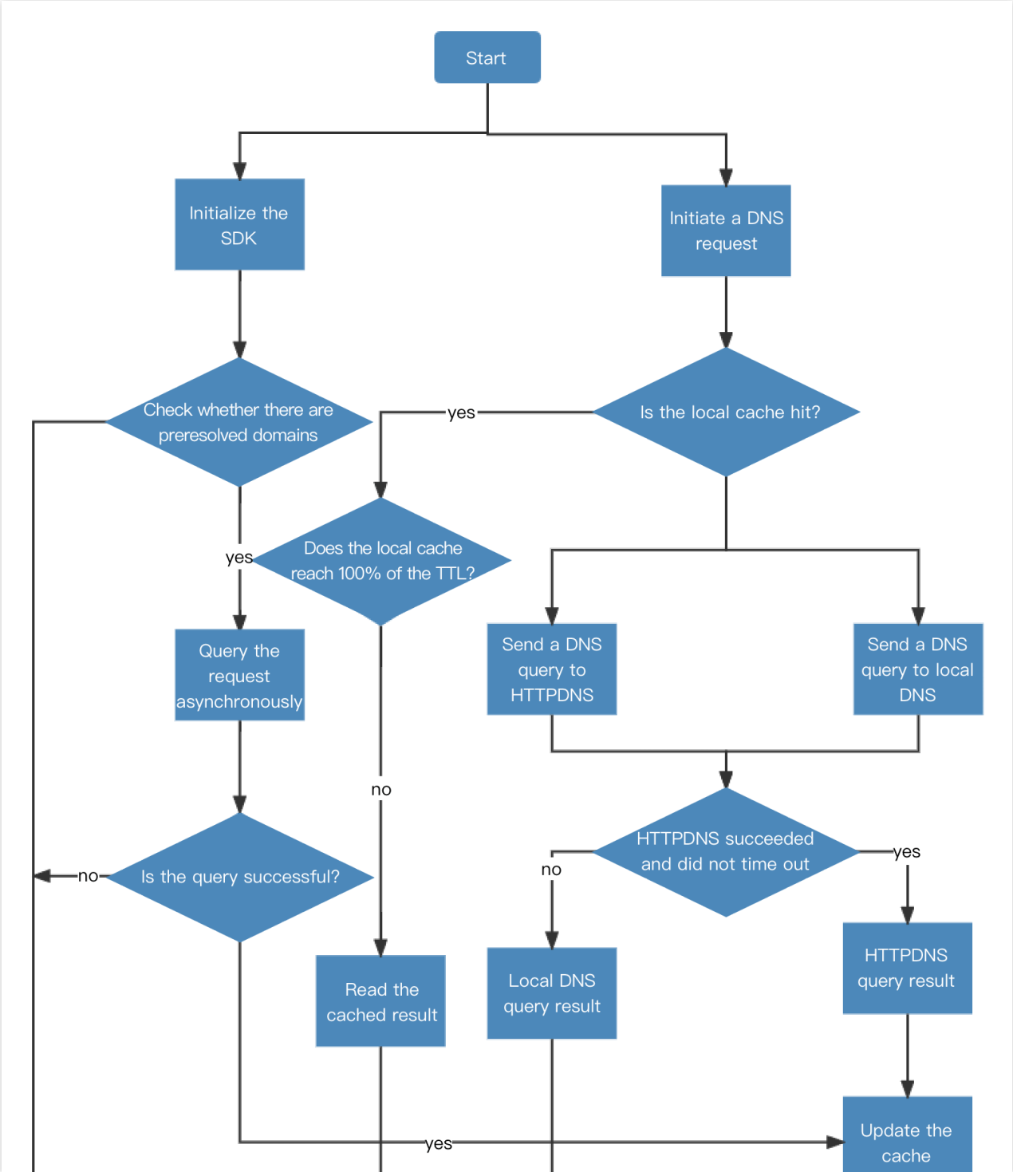


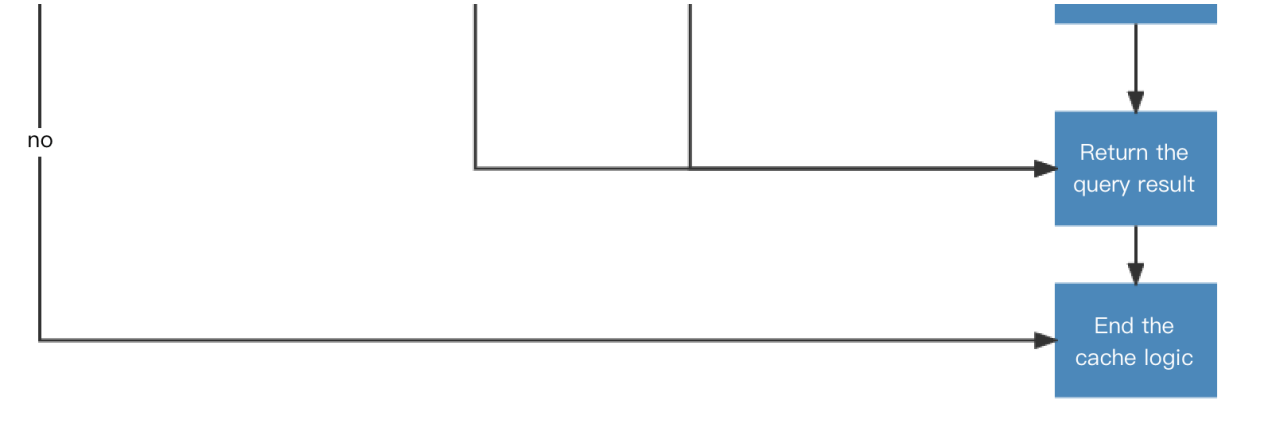
Practical Tutorial of API Connection

Last updated：2024-06-28 14:38:45

Client Connection Process

When connecting to HTTPDNS, you need to modify the DNS mechanism on the mobile client as instructed below:





Design Policy

Follow the following two design policies during modification:

Failover policy

Although HTTPDNS has been connected to BGP Anycast and implemented multi-region cross-IDC disaster recovery, to ensure that the DNS on the client will not be affected even in the worst case, we recommend you use the following failover policy:

1. Send a DNS query to HTTPDNS.
2. If the result returned in response to an HTTPDNS query is not an IP address (empty, not IP format, or connection timeout), the DNS query will be performed by the local DNS. We recommend you set the timeout period to 5 seconds.

Cache policy

As the network environments of mobile internet users are complex, to minimize the delay caused by DNS query, we recommend you perform local caching. The caching rules are as follows:

Cache validity: We recommend you set the cache validity to 120s–600s. It should not be below 60s.

Cache update: The cache should be updated in the following two scenarios:

When the user's network status changes: When a mobile internet user switches from 3G to Wi-Fi or from Wi-Fi to 3G, the network of their access point may change. At this point, you need to send a DNS request to HTTPDNS again to get the optimal direction for the user's current network.

When the cache expires: When the cache of the DNS query result expires, the client should send a new DNS request to HTTPDNS to get the IP associated with the latest domain. To shorten the wait before a new DNS query is performed, we recommend that the DNS query be performed at 75% of the TTL. For example, if the TTL of the local cache is 600s, the client should perform a DNS query at the 600 * 0.75 = 450th second.

In addition to the above suggestions, reducing the number of DNS queries can also effectively reduce network interactions and improve the user access experience. We recommend you minimize the number of domains as long as

your business permits. To distinguish between different resources, we recommend you use URLs.

Notes

In order for you to better modify the mobile client, read the following notes first:
Implement different features with the same domain but different URLs so as to reduce the number of DNS queries. This delivers a better user experience and makes failover easier. This is because each additional domain, even if it has a hit in the cache, will result in an increase of at least 100 milliseconds in the access delay.
The TTL value of the cache should not be too low (at least above 60s) so as to avoid sending frequent requests to HTTPDNS.
The business connected to HTTPDNS needs to retain the user's local DNS as a backup, so that when HTTPDNS cannot work properly (due to an unstable mobile network or HTTPDNS problem), the local DNS can be used.
A 404 error may occur in Android applications, but if the browser access is normal, it may be a permission problem. For more information, see Android HTTP request 404 not found issue.
For `bytetohex` and `hextobyte`, you need to implement the APIs on your own to convert between hexadecimal strings and bytes.
For HTTPS problems, you need to hook the client to check whether the domain and domain extension of the certificate contain the host in the request and replace the IP with the original domain before performing a certificate verification again. You can also ignore certificate verification (similar to the `-k` parameter in curl).
We recommend you set the timeout period to 500 ms for the first HTTPDNS request and 2–5s for subsequent requests.
When the network type changes, for example, from 5G/4G to Wi-Fi or from one Wi-Fi network to another, you need to execute HTTPDNS requests again to purge the local cache.