

SimSun "zh" = 0pt plus 1pt

0.1

2015-11-09 06:35:24

1		3
1.1	3
1.2	3
1.3	4
1.4	9
1.5	9
1.6	10
1.7	12
1.8	13
2		15
HTTP		17

2015-11-09 06:35:23

1.1

readthedocs :

1.1.1

todo:

1.1.2

AppWebApp

App <https://modao.cc/app/POxerAK2EbEA7gmDx2GT>

1.1.3

<https://github.com/Hesong-OpenSource/sanxin-zhuanhao-service-apidoc>

1.2

1. HTTP 1.1 HTTP 2.0 -_- WebAPI
2. ->
3. Restfule API
4. UTF8 JSON
5. HTTPS

1.2.1 HTTP Content

POST PUT JSON Content-Type application/json

Content JSON Content-Type application/json

Content JSON null

, 0.1

```
HTTP/1.1 200 OK
Content-Type: application/json

null
```

1.2.2 HTTP Status Code

200

API 200 OK

Content-Type application/json JSON

500

API 500 Internal Server Error

JSON code text

```
HTTP/1.1 500 Internal Server Error
Content-Type: application/json

{"code": 10013, "text": "callee not allowed"}
```

```
: JSON
```

401

API Status Code

Status Code [RFC 2616](#)

1.3

1.3.1 User API

APPHTTP URL

ANY /api/user/(string: telnum)/*

APP ACCESS-ID ACCESS-KEY

APP ACCESS-ID ACCESS-KEY

URL

Query Parameters

- **accessid** (string) – ID

- **timestamp** (*string*) – Unix 1445851008

```
: 48
```

- **signature** (*string*) – *url path, telnum, password, token* *POST*
/api/user/(string: *telnum*)/login,timestamp *accessid accesskey* 7

APP signature SHA1 16

- 1.url path (/), telnum, password (MD5HEX) , token, timestamp, accessid, accesskey (MD5HEX) 7
- 2.
- 3.SHA-1 (upper case)

```
: POST /api/user/(string: telnum)/login token
```

13887654321

This_Is#My&p@ssw0rd

APPIDKEY developer-001 xm90uojWSd34E8y3

token=4C609E5D5D234A406D446EA42898EFAD50E4541C

1407812629434

URL /api/user/13887654321/path/of/the/api

DCE009D2AF85050E249A6511D1C0F0F180EDFA64

url:

```
http://api/user/13887654321/path/of/the/api?accessid=developer-001&timestamp=1407812629434&signature=
```

:

Java

```
import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;
import java.util.ArrayList;
import java.util.Collections;

public class Signature {

    public static String byteArrayToHex(byte[] byteArray) {
        char[] hexDigits = { '0', '1', '2', '3', '4', '5', '6', '7', '8', '9',
                              'A', 'B', 'C', 'D', 'E', 'F' };
        char[] resultCharArray = new char[byteArray.length * 2];
        int index = 0;
        for (byte b : byteArray) {
            resultCharArray[index++] = hexDigits[b >>> 4 & 0xf];
            resultCharArray[index++] = hexDigits[b & 0xf];
        }
        return new String(resultCharArray);
    }
}
```

```

    }

    public static String hashStr(String input, String digest)
        throws NoSuchAlgorithmException {
        MessageDigest messageDigest = MessageDigest.getInstance(digest);
        byte[] inputByteArray = input.getBytes();
        messageDigest.update(inputByteArray);
        byte[] resultByteArray = messageDigest.digest();
        return byteArrayToHex(resultByteArray);
    }

    public static String calcSignature() throws NoSuchAlgorithmException {
        String accessid = "developer-001";
        String accesskey = hashStr("xm90uojWSd34E8y3", "MD5");
        String urlpath = "/api/user/13887654321/path/of/the/api";
        String telnum = "13887654321";
        String token = "4C609E5D5D234A406D446EA42898EFAD50E4541C";
        String password = hashStr("This_Is#My&p@ssw0rd", "MD5");
        String timestamp = "1407812629434";

        ArrayList<String> tmpList = new ArrayList<String>();
        tmpList.add(accessid);
        tmpList.add(accesskey);
        tmpList.add(urlpath);
        tmpList.add(telnum);
        tmpList.add(token);
        tmpList.add(password);
        tmpList.add(timestamp);
        Collections.sort(tmpList);

        String result = hashStr(String.join("", tmpList), "SHA1");
        return result;
    }

    public static void main(String[] args) {
        try {
            String sigstr = calcSignature();
            System.out.format("Signature = %s", sigstr);
        } catch (NoSuchAlgorithmException e) {
            e.printStackTrace();
        }
    }
}

```

NodeJs

```

var crypto = require('crypto');

(function () {
    let hashStr = function (input, algorithm) {
        let hasher = crypto.createHash(algorithm);
        hasher.update(input);
        return hasher.digest('hex').toUpperCase();
    }
}

```

```

let accessid = "developer-001";
let accesskey = hashStr("xm90uojWSd34E8y3", "md5");
let urlpath = "/api/user/13887654321/path/of/the/api";
let telnum = "13887654321";
let token = "4C609E5D5D234A406D446EA42898EFAD50E4541C";
let password = hashStr("This_Is#My&p@ssw0rd", "md5");
let timestamp = "1407812629434";

let tmpList = [accessid,accesskey, urlpath, telnum, token, password, timestamp];
tmpList.sort();
let signature = hashStr(tmpList.join(''), 'sha1');

console.log("signature = " + signature);
})();

```

Php

```

<?php
$accessid = 'developer-001';
$accesskey = strtoupper(md5('xm90uojWSd34E8y3'));
$url_path = '/api/user/13887654321/path/of/the/api';
$telnum = '13887654321';
$token = '4C609E5D5D234A406D446EA42898EFAD50E4541C';
$password = strtoupper(md5('This_Is#My&p@ssw0rd'));
$timestamp = '1407812629434';
$tmp_arr = array($accessid, $accesskey, $url_path, $telnum, $token, $password, $timestamp);
sort($tmp_arr, SORT_STRING);
$signature = strtoupper(sha1(implode($tmp_arr)));
echo(signature);

```

Python

Python 3.0+

```

from hashlib import sha1, md5
accessid = b'developer-001'
accesskey = bytes(md5(b'xm90uojWSd34E8y3').hexdigest().upper(), 'ascii')
url_path = b'/api/user/13887654321/path/of/the/api'
telnum = b'13887654321'
token = b'4C609E5D5D234A406D446EA42898EFAD50E4541C'
password = bytes(md5(b'This_Is#My&p@ssw0rd').hexdigest().upper(), 'ascii')
timestamp = b'1407812629434'
signature = sha1(b''.join(sorted([accessid, accesskey, url_path, telnum, token, password, timestamp])))
print(signature)

```

Unix

Unix :

, 0.1

C

```
#include <time.h>          /* time_t, struct tm, time ... */

/// ... ..
time_t val = time(NULL);
int ts = (int) val;

/// ... ..
```

C#

```
int tx = (Int32) (DateTime.UtcNow.Subtract (new DateTime (1970, 1, 1))).TotalSeconds;
```

Java

```
long ts = System.currentTimeMillis()/1000L;
```

Javascript

```
var ts = Math.floor(Date.now()/1000);
```

Php

```
<?php
$ts = time();
```

Python

```
import time
ts = int(time.time())
```

1.3.2 CTI API

ANY /api/cti/*
CTIWebAPPAPI

Web CTI

1.SSL

WebCTISSLSSTHTTP [nginx SSL http://nginx.org/en/docs/http/nginx_http_ssl_module.html](http://nginx.org/en/docs/http/nginx_http_ssl_module.html)

2.HTTP

WebCTIHTTPHTTP

1.4

1.4.1

POST `/api/user/(string: telnum)/login`
telnum

Request JSON Object

- **password** (*string*) – MD5

Response JSON Object

- **token** (*string*) – *ANY* `/api/user/(string: telnum)/*`

```
: token
```

1.4.2

POST `/api/user/(string: telnum)/logout`
telnum

1.5

API

```
: ID
```

1.5.1

GET `/api/user/(string: telnum)`
telnum

Response JSON Object

- **telnum** (*string*) –
- **name** (*string*) –
- **createtime** (*string*) – ISO
- **avatar** (*string*) – BASE64

1.5.2

POST `/api/user`

Request JSON Object

- **telnum** (*string*) – ()
- **name** (*string*) – ()
- **avatar** (*string*) – BASE64 ()

1.5.3

PUT /api/user/(string: telnum)
telnum

Request JSON Object

- **name** (*string*) –
- **avatar** (*string*) – BASE64

: JSON *null*

1.5.4

DELETE /api/user/(string: telnum)
telnum

1.6

API

1.6.1

GET /api/user/(string: telnum)/vtelnum
telnum

Query Parameters

- **page** (*int*) – 11
- **perPage** (*int*) –

Response Headers

- *X-Pagination-Current-Page* –
- *X-Pagination-Per-Page* – 1
- *X-Pagination-Totle-Pages* –
- *X-Pagination-Totle-Entries* –

Response JSON Object

- **vtelnum** (*string*) –

(*telnum*) 123 53~4

Request

```
GET /api/user/123/vtelnum?page=2&perPage=2 HTTP/1.1
Host: example.com
```

Response

```

HTTP/1.1 200 OK
Content-Type: application/json
X-Pagination-Current-Page: 2
X-Pagination-Per-Page: 2
X-Pagination-Totle-Pages: 3
X-Pagination-Totle-Entries: 5

[{"vtelnum": "10003"}, {"vtelnum": "10004"}]

```

1.6.2

GET /api/user/(string: telnum)/availablevtelnum
telnum

Query Parameters

- **page** (*int*) – 11
- **perPage** (*int*) –

Response Headers

- *X-Pagination-Current-Page* –
- *X-Pagination-Per-Page* – 1
- *X-Pagination-Totle-Pages* –
- *X-Pagination-Totle-Entries* –

Response JSON Object

- **vtelnum** (*string*) –

1.6.3

POST /api/user/(string: telnum)/vtelnum
telnum

Request JSON Object

- **vtelnum** (*string*) –

1.6.4

DELETE /api/user/(string: telnum)/vtelnum/(string: vtelnum)
telnum

1.6.5

POST /api/user/(string: telnum)/vtelnum/(string: vtelnum)/replace
telnum

Request JSON Object

- **vtelnum** (*string*) –

1.7

1.7.1

POST /api/user/(string: telnum)/makecall
telnum

Request JSON Object

- **caller** (*string*) –
- **callee** (*string*) –

Response JSON Object

- **callid** (*string*) – ID

: API *telnum*

2 *telnum caller caller callee*

:

: (POST /api/user/(string: *telnum*)/cancelcall) *telnum caller*

1001 2001 3001

Request

```
POST /api/user/1001/makecall HTTP/1.1
Host: example.com
Content-Type: application/json

{"caller": "2001", "callee": "3001"}
```

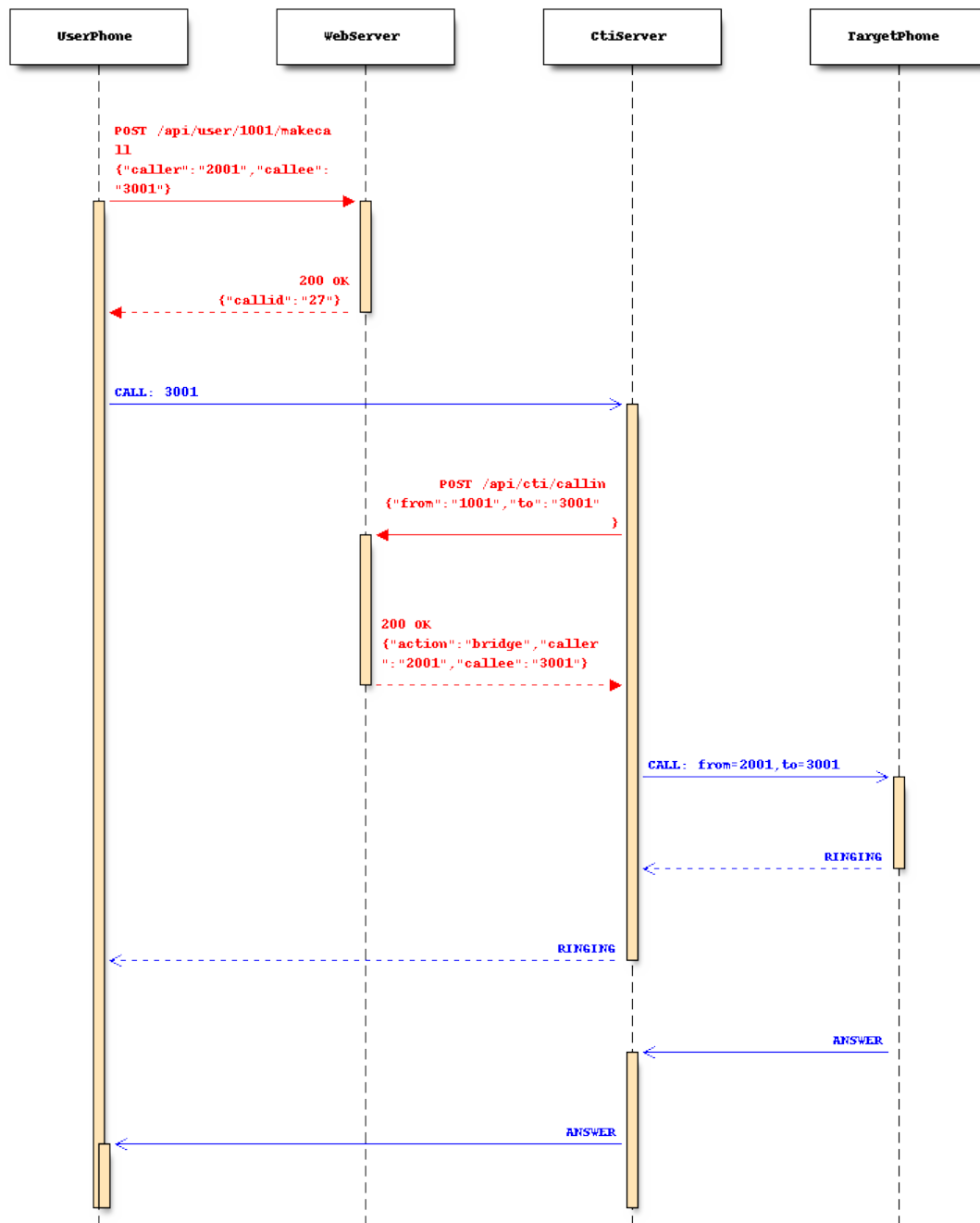
Response

```
HTTP/1.1 200 OK
Content-Type: application/json

{"callid": "12870498123"}
```

1001 2001 3001

WebAPI



1.7.2

`POST /api/user/(string: telnum)/cancelcall`

1.8

(CTI)Web

: HTTP

1.8.1

POST /api/cti/callin
(CTI)API WebAPICTI

Request JSON Object

- **from** (*string*) –
- **to** (*string*) –

Response JSON Object

- **action** (*string*) – CTI
 - *bridge* :
 - *refuse* :

- **caller** (*string*) –

: action “bridge”

- **callee** (*string*) –

: action “bridge”

1001 2001 3001

APP (*POST* /api/user/(*string*: telnum)/makecall) APP 2001 CTIWebWebCTI 2001 3001 1001 3001

Request

```
POST /api/cti/callin HTTP/1.1
Host: example.com
Content-Type: application/json

{"from": "1001", "to": "2001"}
```

Response

```
HTTP/1.1 200 OK
Content-Type: application/json

{"action": "bridge", "caller": "2001", "callee": "3001"}
```

-
-
- search
 - routingtable

/api

```
GET /api/user/(string: telnum),9
GET /api/user/(string:
    telnum)/availablevtelnum,11
GET /api/user/(string:
    telnum)/vtelnum,10
POST /api/cti/callin,14
POST /api/user,9
POST /api/user/(string:
    telnum)/cancelcall,13
POST /api/user/(string:
    telnum)/login,9
POST /api/user/(string:
    telnum)/logout,9
POST /api/user/(string:
    telnum)/makecall,12
POST /api/user/(string:
    telnum)/vtelnum,11
POST /api/user/(string:
    telnum)/vtelnum/(string:
    vtelnum)/replace,11
PUT /api/user/(string: telnum),10
DELETE /api/user/(string: telnum),10
DELETE /api/user/(string:
    telnum)/vtelnum/(string:
    vtelnum),11
ANY /api/cti/*,8
ANY /api/user/(string: telnum)/*,4
```