XEngine API Service Docment

目录

[XEngine API Service Docment 1](#_Toc27224)

[Preface 4](#_Toc14484)

[Reader 4](#_Toc14190)

[Overview 4](#_Toc1185)

[Associate Module 4](#_Toc27759)

[一 Technical structure 4](#_Toc197)

[1.1 HTTP 4](#_Toc29365)

[二 Configure Env 4](#_Toc6522)

[2.1 WINDOWS 4](#_Toc21275)

[2.1.1 Configure Environment 5](#_Toc29017)

[2.1.2 complie and run 5](#_Toc30323)

[2.2 LINUX 5](#_Toc17430)

[2.2.1 Evnironment Configure 5](#_Toc3846)

[2.2.2 Complie and Run 5](#_Toc29343)

[2.3 Version Requirements 5](#_Toc6854)

[2.3.1 System Version 5](#_Toc2284)

[2.3.2 Software Version 6](#_Toc11350)

[三 Interface Protocol 6](#_Toc21370)

[3.1 IDCard Information Query 6](#_Toc22807)

[3.1.1 Request 6](#_Toc2449)

[3.2 Bank Number Check 7](#_Toc23772)

[3.2.1 Request 7](#_Toc31426)

[3.2.2 Reply 7](#_Toc18416)

[3.3 Language Convert 7](#_Toc24890)

[3.3.1 Request 7](#_Toc29326)

[3.3.2 Reply 7](#_Toc18407)

[3.4 Translation 8](#_Toc2317)

[3.4.1 Request 8](#_Toc10877)

[3.4.2 Reply 8](#_Toc24120)

[3.5 P2XP Protocol 8](#_Toc26327)

[3.5.1 Join P2xp 8](#_Toc23989)

[3.5.2 Sync List 9](#_Toc4695)

[3.6 Distributed Lock 10](#_Toc1646)

[3.6.1 Request 10](#_Toc6946)

[3.6.2 reply 10](#_Toc21835)

[3.7 ZIPCode 10](#_Toc7283)

[3.7.1 Request 10](#_Toc2757)

[3.7.2 Reply 11](#_Toc18012)

[3.8 Log System 11](#_Toc18976)

[3.8.1 Log Create 11](#_Toc31913)

[3.8.2 Log Insert 12](#_Toc18243)

[3.8.3 Log Query 12](#_Toc11716)

[3.8.4 Log Delete 13](#_Toc8331)

[3.9 QRCode 14](#_Toc10694)

[3.9.1 Generate QRCode   14](#_Toc11283)

[3.9.2 Parse QRCode 14](#_Toc17186)

[3.10 Socket Test 15](#_Toc31804)

[3.10.1 Create 15](#_Toc3467)

[3.10.2 Destory 16](#_Toc15741)

[3.10.3 Report 16](#_Toc29942)

[3.A Module Plugin System 17](#_Toc24525)

[3.A.1 Password 17](#_Toc30200)

[3.A.2 Zodiac 17](#_Toc20502)

[3.A.3 Bmi 18](#_Toc18894)

[3.A.4 Timezone 18](#_Toc27290)

[3.A.5 Meter Convert 20](#_Toc11210)

[3.A.6 Phone Information Query 21](#_Toc21812)

[3.A.7 IP Address Inforamtion Query 21](#_Toc19192)

[3.B Lua Plugin System 22](#_Toc10451)

[3.B.1 Math Calculation 22](#_Toc21058)

[3.B.2 Time Get 23](#_Toc19365)

[四 Configure Description 23](#_Toc601)

[4.1 Service Configure 23](#_Toc10703)

[4.1.1 basic configure 23](#_Toc29736)

[4.1.2 Max Configure 23](#_Toc10401)

[4.1.3 Time Configure 24](#_Toc3997)

[4.1.4 Log Configure 24](#_Toc20643)

[4.1.5 Dababase Configure 24](#_Toc25104)

[4.1.6 API Interface Configure 24](#_Toc19708)

[4.1.7 Plugin Configure 24](#_Toc6289)

[4.2 Language Configure 25](#_Toc4852)

[4.3 Plugin Configure 25](#_Toc17884)

[五 Advanced Mode 25](#_Toc23200)

[5.1 Lib Plugin Written 25](#_Toc6348)

[5.2 Lua Plugin Written 26](#_Toc6665)

[appendix 26](#_Toc3970)

[Appendix 1 Type Define 26](#_Toc32315)

[Appendix 2 Protocol Define 26](#_Toc21653)

[Appendix 3 Transformation Definition 26](#_Toc31268)

[Appendix 4 update log 26](#_Toc7951)

|  |  |  |  |
| --- | --- | --- | --- |
| File Status：  [ ] Draft  [√] Release | File Name： | XEngine Api Service Docment | |
| Be A Version： | V2.8.0.1001 | |
| Released： | 2023-02-07 | |
| Writer： qyt | | |

# Preface

## Reader

Development ,tester,qa

## **Overview**

This document contains related technical descriptions and interface definitions

## Associate Module

The service used XEngine as Network Toolki.if you want to use code,you have to installed XEngine

# 一 Technical structure

The Service implemented through c/c++,protocol through http.

## HTTP

Send and recv Message through post and get method.

# 二 Configure Env

## 2.1 WINDOWS

Need to download XEngine.

Complie and run and debug by vs.

### 2.1.1 Configure Environment

Follow the instructions in the XEngine Readme file to configure the environment.

### 2.1.2 complie and run

When you complete with configuration.you can come in code path.open XEngine.sln by vs

If environment not have error.complie is succesed

And you need copy file under XEngine\_Release to your complied dir.next step copy file under XEngine depend module to your complied dir.

Note: Xengine environment can be copied to your compilation directory through vscopy script, provided that you configure your xengine environment

## 2.2 LINUX

### 2.2.1 Evnironment Configure

If you use linux.you must running on ubuntu20.04 or centos8.x(Compatible with Centos-like systems)...

### 2.2.2 Complie and Run

Configure complete.you can complie it.open terminal in you xengine\_storage dir and execute command.

complie:make

install:make FLAGS=InstallAll

clean:make FLAGS=CleanAll

If there is no error.you can see complied XEngine\_\*App file in XEngine\_Release

## 2.3 Version Requirements

### 2.3.1 System Version

Minimum version requirements:

WINDOWS: win7 sp1

Ubuntu:20.04

Centos:8.x

MacOS:12

### 2.3.2 Software Version

Minimum version requirements:

XEngine:V7.38

# 三 Interface Protocol

## 3.1 IDCard Information Query

ID check will verify that the ID number is correct

### 3.1.1 Request

Method:GET

Parament:params1 = IP Address

Address:http://127.0.0.1:5501/api?function=id&params1=511025191101018792

Reply:nSex:1 male 0 Women.

nCheck:Checksum

nPoliceID:Police ID

#### 3.1.2.2 Reply

{  
    **"code"**:**0**,  
    **"data"**:{  
        **"nBirthDay"**:**1**,  
        **"nBirthMonth"**:**7**,  
        **"nBirthYear"**:**1988**,  
        **"nCheck"**:**2**,  
        **"nPoliceID"**:**87**,  
        **"nSex"**:**1**,  
        **"tszCity"**:**"内江市"**,  
        **"tszCounty"**:**"资中县"**,  
        **"tszIDNumber"**:**"511025191101018792"**,  
        **"tszProvincer"**:**"四川省"**  
    },  
    **"msg"**:**"success"**  
}

## 3.2 Bank Number Check

### 3.2.1 Request

Method:GET

Parament:params1 = Bank Number

Address:http://127.0.0.1:5501/api?function=bank&params1=621400000

### 3.2.2 Reply

{  
    **"code"**:**0**,  
    **"data"**:{  
        **"enBankType"**:**2**,  
        **"tszBankAbridge"**:**"CMB"**,  
        **"tszBankName"**:**"招商银行"**,  
        **"tszBankNumber"**:**"621400000"**  
    },  
    **"msg"**:**"success"**  
}

## 3.3 Language Convert

### 3.3.1 Request

Method:GET

Parament:params1 = Convert Lanauge

Param2 = language choice

Address:http://192.168.1.12:5501/api?function=language&params1=繁体到简体&params2=1

### 3.3.2 Reply

{  
    **"code"**:**0**,  
    **"data"**:{  
        **"enType"**:**1**,  
        **"tszDestStr"**:**"繁體到簡體"**,  
        **"tszSourceStr"**:**"繁体到简体"**  
    },  
    **"msg"**:**"success"**  
}

## 3.4 Translation

### 3.4.1 Request

Method:GET

Parament:params1 = Translation Source String

Params2 = Translation Type,,Refer:ENUM\_XENGINE\_APISERVICE\_TRANSLATION\_TYPE

Address:http://192.168.1.12:5501/api?function=translation&params1=中国&params2=0

### 3.4.2 Reply

{  
    **"code"**:**0**,  
    **"data"**:{  
        **"enType"**:0,  
        **"tszDestStr"**:**"china"**,  
        **"tszSourceStr"**:**"中国"**  
    },  
    **"msg"**:**"success"**  
}

## 3.5 P2XP Protocol

P2XP Protocol implemented through Http ,P2XP need to user infomation,if you want to use p2xp protocol,you may be develop authorize interface for verifying user infomation.

When user login by p2xp,user have to heartbeat every five second.

### 3.5.1 Join P2xp

Only sent to this protocol.the server will record client.client can be join p2xp network.

This function is mainly for the use of distributed storage networks across network segments. If you don't have this convenient requirement, you don't need to use it

#### 3.5.1.1 Request

Method:POST

Parament:params1 = Operator Code

Address:http://192.168.1.12:5501/api?function=p2p&params1=24577

协议体:

{  
    **"tszUserName"**:**"User name"**,  
    **"tszPrivateAddr"**:**"private or lan address"**,  
    **"tszPublicAddr"**:**"public address"**,  
    **"dwConnectType"**:**0**,  
    **"dwPeerType"**:**0**  
}

#### 3.5.1.2 Reply

{  
    **"code"**:**0**,  
    **"msg"**:**"success"**  
}

### 3.5.2 Sync List

The network address only synchronizes the internal network IP, and the external network address synchronization can synchronize all the internal network IP addresses under this external network IP address. For some large network companies and complex internal networks, this function can be used to easily determine that they are on the same network. Users under

#### 3.5.2.1 Request

Method:POST

Parament:params1 = Operator Code

Address:http://192.168.1.12:5501/api?function=p2p&params1=24579

协议体:

{  
    **"tszUserName"**:**"User name"**,  
    **"tszPrivateAddr"**:**"private or lan address"**,  
    **"tszPublicAddr"**:**"public address"**  
}

#### 3.5.2.2 Reply

{  
    **"code"**:**0**,  
    **"msg"**:**"sucess"**,  
    **"ClientCount"**:**2**,  
    **"ClientArray"**:[  
        {  
            **"ClientAddr"**:**"192.168.1.101"**,  
            **"ClientUser"**:**"123123aa"**  
        },  
        {  
            **"ClientAddr"**:**"192.168.1.102"**,  
            **"ClientUser"**:**"123123bb"**  
        }  
    ]  
}

## 3.6 Distributed Lock

### 3.6.1 Request

Method:GET

Parament:无

Url:http://127.0.0.1:5501/api?function=lock&params1=token&params2=0

Params1 is token,can be system create or by your self

Params2 is type,refer user protocol fileENUM\_XENGINE\_APISERVICE\_LOCKER\_TYPE

Payload:null

### 3.6.2 reply

{

"code":0,

"data":{

"xhToken":1000106561

},

"msg":"success"

}

## 3.7 ZIPCode

### 3.7.1 Request

Method:POST

Address:http://127.0.0.1:5501/api?function=zipcode&params1=0

Parament:params1,0 Query by ZIPCode,1 Query By Name

Payload:

0:

{

"zipcode":100010

}

1:

{

"tszCounty":"资中县"

}

### 3.7.2 Reply

{

"code":0,

"data":{

"dlLat":39.93157,

"dlLng":116.41005,

"nAreaCode":10,

"nLevel":3,

"nZipCode":100010,

"tszCity":"北京市",

"tszCounty":"东城",

"tszPinYin":"dongcheng",

"tszProvincer":"北京"

},

"msg":"success"

}

## 3.8 Log System

The log system can provide log query service for permanent log storage

### 3.8.1 Log Create

#### 3.8.1.1 Request

Method:POST

Address:http://127.0.0.1:5501/api?function=log&params1=0

Parament:params1=0 mean is create,1 mean is insert,2 mean is query,3 mean is delete

Payload:

{

"tszTableName":"xengine"

}

#### 3.8.1.2 Reply

{

"code":0

}

### 3.8.2 Log Insert

#### 3.8.2.1 Request

Method:POST

Address:http://127.0.0.1:5501/api?function=log&params1=1

Payload:

{

"tszTableName":"xengine",

"tszLogBuffer":"d12d",

"nLogSize":4,

"tszFileName":"file.cpp",

"tszFuncName":"xengine\_file\_insert",

"tszLogTimer":"2023-01-13 22:10:01",

"nLogLine":102,

"nLogLevel":4

}

#### 3.8.2.2 Reply

{

"code":0

}

### 3.8.3 Log Query

#### 3.8.3.1 Request

Method:POST

Address:http://127.0.0.1:5501/api?function=log&params1=2

Payload:

{

"tszTableName":"xengine",

"tszTimeStart":"2023-01-13 22:10:00",

"tszTimeEnd":"2023-01-13 22:10:02"

}

#### 3.8.3.2 Reply

{

"code":0,

"data":[

{

"nLogLevel":4,

"nLogLine":102,

"nLogSize":0,

"tszFileName":"file.cpp",

"tszFuncName":"xengine\_file\_insert",

"tszLogBuffer":"adwdad12d21d123d132rd213d32f23df23rf",

"tszLogTimer":"2023-01-13 22:10:01",

"tszTableName":"xengine"

},

{

"nLogLevel":4,

"nLogLine":102,

"nLogSize":0,

"tszFileName":"file.cpp",

"tszFuncName":"xengine\_file\_insert",

"tszLogBuffer":"adwdad12d21d123d132rd213d32f23df23rf",

"tszLogTimer":"2023-01-13 22:10:01",

"tszTableName":"xengine"

}

],

"msg":"success"

}

### 3.8.4 Log Delete

#### 3.8.4.1 Request

Method:POST

Address:http://127.0.0.1:5501/api?function=log&params1=3

Payload:

{

"tszTableName":"xengine"

}

#### 3.8.4.2 Reply

{

"code":0

}

## 3.9 QRCode

QRCode Generate and parse

### 3.9.1 Generate QRCode

#### 3.9.1.1 Request

Method:POST

API:http://127.0.0.1:5501/api?function=qrcode&params1=0

Payload:

{

"tszMsgBuffer":"encode qrcode datas",

"tszFmtBuffer":"will return image format,like is:.png"

}

#### 3.9.1.2 Reply

Content-Type: image/png;

return image format,can be write memory or file

### 3.9.2 Parse QRCode

#### 3.9.2.1 Request

Method:POST

API:http://127.0.0.1:5501/api?function=qrcode&params1=1

Type:Content-Type: image/png

Payload:image

#### 3.9.2.2 Reply

return parse qrcode string information

## 3.10 Socket Test

A service can be tested, which can be used to generate reports or to judge the viability of the service

### 3.10.1 Create

#### 3.10.1.1 Request

Method:POST

API:http://127.0.0.1:5501/api?function=test&params1=0

Payload:

{

"tszAPIUrl":"report address",

"xhToken":123123,

"bTCP":true,

"bConn":true,

"tszAddr":"测试IPAPI",

"nPort":3555,

"nCloseWaitContTime":2000,

"nConnectCount":10,

"nConnectTest":1,

"nContWaitTime":1000,

"tszSDBuffer":"Send msg",

"tszRVBuffer":"Recv msg,if you want to verifaction datas",

"nRVLen":1024,

"nSDLen":1024

}

#### 3.10.1.2 Reply

{

"code":0

}

### 3.10.2 Destory

#### 3.10.2.1 Request

Method:POST

API:http://127.0.0.1:5501/api?function=test&params1=1

Payload:

{

"xhToken":123123,

"bTCP":true,

"bConn":true

}

#### 3.10.2.2 Reply

{

"code":0

}

### 3.10.3 Report

#### 3.10.3.1 Request

Method:POST

API:Create tszAPIUrl Parament

Payload:

{

"lpszIPAddr":"Test ipaddr",

"nIPPort":3555,

"xhToken":3133131,

"nNumber":10,

"nFailed":0,

"nSuccess":10,

"nStatus":0

}

#### 3.10.3.2 Reply

Null

## 3.A Module Plugin System

### 3.A.1 Password

#### 3.A.1.1 Request

Method:GET

Parament:params1 = Generated type, 1 is random,2 is int,3 is words

params2 = Password length

Address:http://192.168.1.12:5501/api?function=pass&params1=1&params2=4

#### 3.A.1.2 Reply

{  
    **"code"**:**0**,  
    **"data"**:{  
        **"tszParamLength"**:**"4"**,  
        **"tszParamType"**:**"1"**,  
        **"tszPassword"**:**"9xKm"**  
    },  
    **"msg"**:**"success"**  
}

### 3.A.2 Zodiac

#### 3.A.2.1 Request

Method:GET

Parament:params1 = process date,example:1988 year 1 month 21day is:19880121

Address:http://192.168.1.12:5501/api?function=zodiac&params1=19880121

#### 3.A.2.2 Reply

{  
    **"code"**:**0**,  
    **"data"**:{  
        **"tszChineseZodiac"**:**"龙"**,  
        **"tszDate"**:**"19880121"**,  
        **"tszEnglishZodiac"**:**"水瓶座"**  
    },  
    **"msg"**:**"success"**  
}

### 3.A.3 Bmi

#### 3.A.3.1 Request

Method:GET

Parament:params1 = height, in meters

params2 = weight, in kilograms

Address:http://127.0.0.1:5501/api?function=bmindex&params1=1.78&params2=62

#### 3.A.3.2 Reply

{  
    **"code"**:**0**,  
    **"data"**:{  
        **"Result"**:**"标准"**,  
        **"flHigh"**:**1.78**,  
        **"flValue"**:**19.568236333796236**,  
        **"flWeight"**:**62**  
    },  
    **"msg"**:**"success"**  
}

### 3.A.4 Timezone

#### 3.A.4.1 Count Query

##### 3.A.4.1.1 Request

Method:GET

Parament:params1 = Type,0 is query count

Address:http://127.0.0.1:5501/api?function=timezone&params1=0

##### 3.A.4.1.2 Reply

{  
    **"code"**:**0**,  
    **"data"**:{  
        **"Count"**:**91**  
    },  
    **"msg"**:**"success"**  
}

#### 3.A.4.2 List Query

##### 3.A.4.2.1 Request

Method:GET

Parament:params1 = Type,1 is list query

params2 = the range of the list to query

Address:http://127.0.0.1:5501/api?function=timezone&params1=1&params2=1-10

##### 3.A.4.2.2 Reply

{  
    **"Array"**:[  
        {  
            **"tszTimeCountry"**:**"大西洋标准时间（加拿大）"**,  
            **"tszTimeZone"**:**"AST"**  
        },  
        {  
            **"tszTimeCountry"**:**"夏威夷-阿拉斯加标准时间"**,  
            **"tszTimeZone"**:**"AHST"**  
        },  
        {  
            **"tszTimeCountry"**:**"以色列标准时间"**,  
            **"tszTimeZone"**:**"IST"**  
        }

],  
    **"Count"**:**3**,  
    **"code"**:**0**,  
    **"msg"**:**"success"**,  
    **"nPosEnd"**:**3**,  
    **"nPosStat"**:**1**  
}

#### 3.4.A.3 Timezone Convert

##### 3.A.4.3.1 Request

Method:GET

Parament:params2 = type,2 is timezone convert

params2 = convert type

Params3 = time

Address:http://127.0.0.1:5501/api?function=timezone&params1=2&params2=AST&params3=2022-07-21\_12:10:01

##### 3.A.4.3.2 Reply

{  
    **"code"**:**0**,  
    **"data"**:{  
        **"tszTimeCountry"**:**"大西洋标准时间（加拿大）"**,  
        **"tszTimeStr"**:**"2022-07-21 08:10:01"**,  
        **"tszTimeZone"**:**"AST"**  
    },  
    **"msg"**:**"success"**  
}

### 3.A.5 Meter Convert

#### 3.A.5.1 Request

Method:GET

Parament:type=type:0 length,1 tempareture,2 power,3 speed

source=source sub type,refer export head file

value=convert value

Address:http://127.0.0.1:5501/api?function=meter&type=0&source=0&value=10

#### A.5.2 Reply

Depending on the type, the returned value is also different, it is recommended to test it yourself

### 3.A.6 Phone Information Query

#### 3.A.6.1 Request

Method:GET

Parament:params1 = Phone Number

Address:http://127.0.0.1:5501/api?function=phone&params1=13699428888

#### 3.A.6.2 Reply

{  
    **"code"**:**0**,  
    **"data"**:{  
        **"nAreaCode"**:**28**,  
        **"nPhoneNumber"**:**13699428888**,  
        **"nZipCode"**:**610000**,  
        **"tszCity"**:**"成都"**,  
        **"tszProvincer"**:**"四川"**  
    },  
    **"msg"**:**"success"**  
}

### 3.A.7 IP Address Inforamtion Query

#### 3.A.7.1 Request

Method:GET

Parament:params1 = IP Address

Address:http://127.0.0.1:5501/api?function=ip&params1=1.29.164.255

#### 3.A.7.2 Reply

{  
    **"code"**:**0**,  
    **"data"**:{  
        **"tszIPAddr"**:**"1.29.164.255"**,  
        **"tszIPAddress"**:**"内蒙古通辽市霍林郭勒市"**,  
        **"tszIPCity"**:**"通辽市"**,  
        **"tszIPCountry"**:**"中国"**,  
        **"tszIPCounty"**:**"霍林郭勒市"**,  
        **"tszIPISP"**:**"联通"**,  
        **"tszIPProvince"**:**"内蒙古"**  
    },  
    **"msg"**:**"success"**  
}

##### 3.A.7.2.1 Error Reply

If http is not 200 or code is not 0,that’s mean is incorrent.

Msg field is json have.stream return type just is code msg only has json return type, data stream return type is only code, the same below

{  
    **"code"**:**1001**,  
    **"msg"**:**"parse is incorrent"**  
}

## 3.B Lua Plugin System

### 3.B.1 Math Calculation

#### 3.B.1.1 Request

Method:GET

Parament:param1=Operator value

param2=operated value

type=0 add, 1 subtract, 2 multiply, 3 divide

Address:http://127.0.0.1:5501/api?function=cal&param1=1&param2=2&type=0

#### 3.B.1.2 Reply

{

"code":0,

"msg":"success",

"data":{

"nType":0,

"nValue1":1,

"nValue2":1,

"nCal":2

}

}

### 3.B.2 Time Get

#### 3.B.2.1 Request

Method:GET

Parament:type=operator type

0get system time,1 get program start up time(second),2 system timestamp

API:http://127.0.0.1:5501/api?function=time&type=1

#### 4.B.2.2 Reply

{

"code":0,

"msg":"success",

"data":{

"Time":"2023-01-13 14:52:05"

}

}

# 四 Configure Description

## 4.1 Service Configure

Basic Configure File:XEngine\_Config.json

### 4.1.1 basic configure

* tszIPAddr: location ip address
* bDeamon: 1 deamon process run 0 is terminal run
* nHttpPort:http port

### 4.1.2 Max Configure

XMax Configure

* nMaxClient Allow Max Client Count
* nMaxQueue Allow Max Queue
* nIOThread:network io process threads number
* nHttpThread:http process threads number

### 4.1.3 Time Configure

XTime Configure

* nTimeCheck:check time
* nP2PTimeOut:P2P Timeout
* nHttpTimeOut:same nTCPTimeOut

### 4.1.4 Log Configure

XLog Configure

* MaxSize:Log file size
* MaxCount:Log File Number
* LogLeave:Allow save level

### 4.1.5 Dababase Configure

XSql Configure

* SQLAddr:Database Address
* SQLPort:Port
* SQLUser:User name
* SQLPass:password

### 4.1.6 API Interface Configure

XDatabase Configure

* tszIPData:IP Database address
* tszIDData:ID Database address
* tszPhoneData:phone database address
* tszBankUrl:Bank card verification interface address
* tszTranslationUrl:translate interface address

### 4.1.7 Plugin Configure

XPlugin Configure

* bEnable:whether enable plugin
* tszPluginLib:lib plugin configure file
* tszPluginLua:lua plugin configure file

## 4.2 Language Configure

Configure File XEngine\_OPenccConfig.json

## 4.3 Plugin Configure

Configure File:XEngine\_PluginLib.json XEngine\_PluginLua.json

This is json array:

* PluginEnable:whether is enable
* PluginMethod:The name of the method to be registered, when function= this name, the system will automatically hand it over to the module for processing
* PluginFile:module file address

# 五 Advanced Mode

## 5.1 Lib Plugin Written

A module plug-in is a dll or so or dylib module written in a programming language. A module plug-in needs to implement three functions: PluginCore\_Init, PluginCore\_UnInit, and PluginCore\_Call

extern "C" *BOOL* PluginCore\_Init(*LPVOID* lParam = *NULL*);

extern "C" *void* PluginCore\_UnInit();

extern "C" *BOOL* PluginCore\_Call(*TCHAR* \* \*\*pppHDRList, int nListCount, int\* pInt\_HTTPCode, *TCHAR* \* ptszMsgBuffer, int\* pInt\_MsgLen, *LPCTSTR* lpszMsgBuffer = *NULL*, int nMsgLen = 0);

You only need to complete your own code internally. For the specific meaning, please refer to the comments in the module code

You must enable your plugin in plugin configure,now you can use your plugin module

PluginCore\_Init: an untyped pointer parameter, a boolean return value, which is automatically called when the plug-in system is initialized.

PluginCore\_UnInit: no return value and no parameters, it will be called automatically when the plugin is uninstalled

PluginCore\_Call: It has 7 parameters and a boolean return value. It is called once when the specified plugin is found.

* pppHDRList:Three-level character pointer. It is a request URL header. For example, http://192.168.1.8:5501/api after http://192.168.1.8:5501/api?function=cal&params1=1&params2=1&param3=0 ?function=calparams1=1 params2=1 param3=0, here are three string arrays
* nListCount:The number of parameters in the requested URL header, here are three
* pInt\_HTTPCode: integer pointer, output HTTP status code
* ptszMsgBuffer:character pointer, output the content of the load
* pInt\_MsgLen:the content size of the output payload
* lpszMsgBuffer:Enter the HTTP payload content of the request, only valid if nMsgLen is greater than 0
* nMsgLen:Enter the payload content size of the request.

## 5.2 Lua Plugin Written

You can now write plug-ins through lua scripts, no need to learn programming languages. It is very convenient for everyone to develop their own query system,you can refer lua script in XEngine\_LuaPlugin dir

The lua plugin needs to implement three functions: PluginCore\_Init, PluginCore\_UnInit and PluginCore\_Call

PluginCore\_Init: an untyped pointer parameter, a boolean return value, which is automatically called when the plug-in system is initialized.

PluginCore\_UnInit: no return value and no parameters, it will be called automatically when the plugin is uninstalled

PluginCore\_Call: It has 4 parameters and a boolean return value. It is called once when the specified plugin is found.

* lpszStrUrl:URL header parameter, used for the parameter content after function=api?, such as param1=1&param2=2&param3=0
* nListCount:Indicates the number of header parameters, here is 3
* lpszMsgBuffer:The request data of the load, if nMsgLen is greater than 0, it means it exists
* nMsgLen:The size of the payload data

# appendix

## Appendix 1 Type Define

reference file:XEngine\_CommHdr.h

## Appendix 2 Protocol Define

reference file:XEngine\_ProtocolHdr.h

## Appendix 3 Transformation Definition

reference file:XEngine\_Types.h only LINUX

## Appendix 4 update log