XEngine API Service Docment

目录

[XEngine API Service Docment 1](#_Toc211353389)

[Preface 6](#_Toc211353390)

[Reader 6](#_Toc211353391)

[Overview 6](#_Toc211353392)

[Associate Module 6](#_Toc211353393)

[一 Technical structure 6](#_Toc211353394)

[1.1 HTTP 6](#_Toc211353395)

[1.2 Character 6](#_Toc211353396)

[二 Configure Env 6](#_Toc211353397)

[2.1 WINDOWS 7](#_Toc211353398)

[2.1.1 Configure Environment 7](#_Toc211353399)

[2.1.2 complie and run 7](#_Toc211353400)

[2.2 LINUX 7](#_Toc211353401)

[2.2.1 Evnironment Configure 7](#_Toc211353402)

[2.2.2 Complie and Run 7](#_Toc211353403)

[2.3 Version Requirements 8](#_Toc211353404)

[2.3.1 System Version 8](#_Toc211353405)

[2.3.2 Software Version 8](#_Toc211353406)

[三 Interface Protocol 8](#_Toc211353407)

[3.1 IDCard Information Query 8](#_Toc211353408)

[3.1.1 Request 8](#_Toc211353409)

[3.2 Bank Number Check 9](#_Toc211353410)

[3.2.1 Request 9](#_Toc211353411)

[3.2.2 Reply 9](#_Toc211353412)

[3.4 Translation 9](#_Toc211353413)

[3.4.1 Request 9](#_Toc211353414)

[3.4.2 Reply 10](#_Toc211353415)

[3.5 P2XP Protocol 10](#_Toc211353416)

[3.5.1 Join P2xp 10](#_Toc211353417)

[3.5.2 Sync List 11](#_Toc211353418)

[3.5.3 Leave p2xp 12](#_Toc211353419)

[3.5.4 Wan list addresss 12](#_Toc211353420)

[3.5.5 Heartbeat p2xp 13](#_Toc211353421)

[3.6 Distributed Lock 13](#_Toc211353422)

[3.6.1 Request 13](#_Toc211353423)

[3.6.2 reply 14](#_Toc211353424)

[3.7 ZIPCode 14](#_Toc211353425)

[3.7.1 Request 14](#_Toc211353426)

[3.7.2 Reply 14](#_Toc211353427)

[3.8 Log System 15](#_Toc211353428)

[3.8.1 Log Create 15](#_Toc211353429)

[3.8.2 Log Insert 15](#_Toc211353430)

[3.8.3 Log Query 16](#_Toc211353431)

[3.8.4 Log Delete 17](#_Toc211353432)

[3.8.5 Log Table Show 17](#_Toc211353433)

[3.9 QRCode 18](#_Toc211353434)

[3.9.1 Generate QRCode 18](#_Toc211353435)

[3.9.2 Parse QRCode 18](#_Toc211353436)

[3.10 Socket Test 19](#_Toc211353437)

[3.10.1 Create 19](#_Toc211353438)

[3.10.2 Destory 20](#_Toc211353439)

[3.10.3 Report 20](#_Toc211353440)

[3.11 Data Test 21](#_Toc211353441)

[3.11.1 Request 21](#_Toc211353442)

[3.11.2 Reply 21](#_Toc211353443)

[3.12 Short Link 21](#_Toc211353444)

[3.12.1 Create 21](#_Toc211353445)

[3.12.2 Get 22](#_Toc211353446)

[3.12.3 Delete 23](#_Toc211353447)

[3.12.4 Query 23](#_Toc211353448)

[3.13 Words Filter 24](#_Toc211353449)

[3.13.1 Insert 24](#_Toc211353450)

[3.13.2 Delete 25](#_Toc211353451)

[3.13.3 Query 25](#_Toc211353452)

[3.13.4 List 26](#_Toc211353453)

[3.14 Backstage management agreement 26](#_Toc211353454)

[3.14.1 Download 26](#_Toc211353455)

[3.14.2 Delete file 27](#_Toc211353456)

[3.14.3 Delete folder 27](#_Toc211353457)

[3.14.4 upload files 27](#_Toc211353458)

[3.14.5 Get file list 27](#_Toc211353459)

[3.14.6 Create Process 28](#_Toc211353460)

[3.14.7 Stop Process 28](#_Toc211353461)

[3.14.8 Shut down the system 28](#_Toc211353462)

[3.14.9 Excuting command 28](#_Toc211353463)

[3.14.10 Information Report 28](#_Toc211353464)

[3.14.11 Record Screen 29](#_Toc211353465)

[3.15 Image Process Protocol 29](#_Toc211353466)

[3.15.1 Get Text with image 29](#_Toc211353467)

[3.15.2 Image Info Get 30](#_Toc211353468)

[3.15.3 Image Info Modify 30](#_Toc211353469)

[3.16 Process Deamon 31](#_Toc211353470)

[3.16.1 Request 31](#_Toc211353471)

[3.16.2 Reply 31](#_Toc211353472)

[3.17 Region ID Convert 31](#_Toc211353473)

[3.17.1 Request 31](#_Toc211353474)

[3.17.2 Reply 32](#_Toc211353475)

[3.18 Machine Information 32](#_Toc211353476)

[3.18.1 Insert 32](#_Toc211353477)

[3.18.2 Query 33](#_Toc211353478)

[3.18.3 Delete 34](#_Toc211353479)

[3.19 Oil Price Query 34](#_Toc211353480)

[3.19.1 Request 34](#_Toc211353481)

[3.19.2 Reply 34](#_Toc211353482)

[3.20 weather Query 35](#_Toc211353483)

[3.20.1 Request 35](#_Toc211353484)

[3.20.2 Reply 35](#_Toc211353485)

[3.21 Region ID Query 36](#_Toc211353486)

[3.21.1 Request 36](#_Toc211353487)

[3.21.2 Reply 36](#_Toc211353488)

[3.A Module Plugin System 36](#_Toc211353489)

[3.A.1 Password 36](#_Toc211353490)

[3.A.2 Zodiac 37](#_Toc211353491)

[3.A.3 Bmi 37](#_Toc211353492)

[3.A.4 Timezone 38](#_Toc211353493)

[3.A.5 Meter Convert 40](#_Toc211353494)

[3.B Lua Plugin System 40](#_Toc211353495)

[3.B.1 Math Calculation 40](#_Toc211353496)

[3.B.2 Time Get 41](#_Toc211353497)

[3.B.3 HeartBeat 41](#_Toc211353498)

[3.C APIModule DB Query 42](#_Toc211353499)

[3.C.1 Phone Information Query 42](#_Toc211353500)

[3.C.2 IP Address Inforamtion Query 43](#_Toc211353501)

[3.C.3 MAC Address Vendor Query 44](#_Toc211353502)

[3.D RFC Protocol 44](#_Toc211353503)

[四 Configure Description 44](#_Toc211353504)

[4.1 Service Configure 44](#_Toc211353505)

[4.1.1 basic configure 45](#_Toc211353506)

[4.1.2 Max Configure 45](#_Toc211353507)

[4.1.3 Time Configure 45](#_Toc211353508)

[4.1.4 Log Configure 45](#_Toc211353509)

[4.1.5 Dababase Configure 45](#_Toc211353510)

[4.1.6 API Interface Configure 46](#_Toc211353511)

[4.1.7 Plugin Configure 46](#_Toc211353512)

[4.1.8 Configure file to configure 46](#_Toc211353513)

[4.1.9 Short Link Configure 46](#_Toc211353514)

[4.1.10 User Verification 46](#_Toc211353515)

[4.1.11 Report Configure 47](#_Toc211353516)

[4.1.12 Image to text Configure 47](#_Toc211353517)

[4.1.13 API Module Configure 47](#_Toc211353518)

[4.2 DNS Configure 47](#_Toc211353519)

[4.3 Process Deamon Configure 48](#_Toc211353520)

[4.3.1 Process List 48](#_Toc211353521)

[五 Advanced Mode 48](#_Toc211353522)

[5.1 Lib Plugin Written 48](#_Toc211353523)

[5.2 Lua Plugin Written 49](#_Toc211353524)

[5.3 Short Link Forward 49](#_Toc211353525)

[5.4 Configure Hot reload 50](#_Toc211353526)

[5.4.1 Operator Command 50](#_Toc211353527)

[5.4.2 API Interface 50](#_Toc211353528)

[5.5 conditional compilation 50](#_Toc211353529)

[六 Other Information 50](#_Toc211353530)

[6.1 Model File 50](#_Toc211353531)

[appendix 51](#_Toc211353532)

[Appendix 1 Type Define 51](#_Toc211353533)

[Appendix 2 Protocol Define 51](#_Toc211353534)

[Appendix 3 Transformation Definition 51](#_Toc211353535)

[Appendix 4 update log 51](#_Toc211353536)

|  |  |  |  |
| --- | --- | --- | --- |
| File Status：  [ ] Draft  [√] Release | File Name： | XEngine Api Service Docment | |
| Be A Version： | V3.13.0.1001 | |
| Released： | 2025-10-15 | |
| 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.

Manage your machine or query information through HTTP API

## 1.1 HTTP

Send and recv Message through post and get method.

## 1.2 Character

Please note: Unless otherwise specified, the character set of all requests and responses are encoded in UTF8

# 二 Configure Env

Env Request：mysql Service。And import sql file in XEngine\_SQL

First import CreateDatabase.sql。And import other sql file

## 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 ubuntu24.04 or centos9.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

Tpis:if you want to enable verification protocol,then you have to add follow this field in json

”tszUserName”:”user name”

”tszUserPass”:”user password”

## 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"**  
}

## 3.4 Translation

### 3.4.1 Request

Method:GET

Parament:msg= Translation Source String

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

Address:http://127.0.0.1:5501/api?function=translation&msg=中文翻译成英文&src=1&dst=2

### 3.4.2 Reply

{

"code":0,

"data":{

"tszDestStr":"Translate from Chinese to English",

"tszFromStr":"zh",

"tszSourceStr":"中文翻译成英文",

"tszToStr":"en"

},

"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

Protocol Body:

{  
    **"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=24581

Protocol Body:

{

"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"**  
        }  
    ]  
}

### 3.5.3 Leave p2xp

#### 3.5.3.1 Request

Method:POST

Parament:params1 = Operator Code

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

Protocol Body:

{

"tszUserName":"User name",

"tszPrivateAddr":"private or lan address",

"tszPublicAddr":"public address"

}

#### 3.5.3.2 Reply

{

"code":0,

"msg":"success"

}

### 3.5.4 Wan list addresss

#### 3.5.4.1 Request

Method:POST

Parament:params1 = Operator Code

Address:http://192.168.1.12:5501/api?function=p2p&params1=24583

Protocol Body:无

#### 3.5.4.2 Reply

{

"ClientArray":[

{

"tszWAddr":"117.172.221.14"

}

],

"ClientCount":0,

"code":0,

"msg":"sucess"

}

### 3.5.5 Heartbeat p2xp

#### 3.5.5.1 Request

Method:POST

Parament:params1 = Operator Code

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

Protocol Body:

{

"tszUserName":"User name",

"tszPrivateAddr":"private or lan address",

"tszPublicAddr":"public address"

}

#### 3.5.5.2 Reply

{

"code":0,

"msg":"success"

}

## 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=other value

Payload:

{

"tszTableName":"xengine"

}

#### 3.8.4.2 Reply

{

"code":0

}

### 3.8.5 Log Table Show

#### 3.8.5.1 Request

Method:POST

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

Payload:None

#### 3.8.5.2 Reply

{

"code": 0,

"data": [

{

"tszTableName": "XLogTemplate"

},

{

"tszTableName": "xengine"

}

],

"msg": "success"

}

## 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=socket&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=socket&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

## 3.11 Data Test

### 3.11.1 Request

Method:POST

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

0:Send What,reply What

1:Reply to confirm

Payload:User Custom

### 3.11.2 Reply

User Custom

## 3.12 Short Link

Users can generate a short connection by themselves, and then access it through the URL of the short connection. It is easy to remember

### 3.12.1 Create

#### 3.12.1.1 Request

Method:POST

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

Payload:

{

"tszFullUrl":"http://bbs.xyry.org/forum.php?mod=viewthread&tid=2&extra=page%3D1",

"tszCvtUrl":"http://app.xyry.org:5501",

"nLength":6

}

#### 3.12.1.2 Reply

{

"code" : 0,

"data" :

{

"tszCreateTime" : "2023-04-11 14:42:32",

"tszCvtUrl" : "http://app.xyry.org:5501",

"tszFullUrl" : "http://bbs.xyry.org/forum.php?mod=viewthread&tid=2&extra=page%3D1",

"tszKeyUrl" : "7g4ey5",

"tszMapUrl" : "http://app.xyry.org:5501/7g4ey5",

"tszShotUrl" : "http://app.xyry.org:5501"

},

"msg" : "success"

}

### 3.12.2 Get

#### 3.12.2.1 Request

Method:POST

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

Payload:

{

"tszMapUrl":"http://app.xyry.org:5501/7g4ey5"

}

#### 3.12.2.2 Reply

{

"code" : 0,

"data" :

{

"tszCreateTime" : "2023-04-11 14:42:32",

"tszCvtUrl" : "http://app.xyry.org:5501",

"tszFullUrl" : "http://bbs.xyry.org/forum.php?mod=viewthread&tid=2&extra=page%3D1",

"tszKeyUrl" : "7g4ey5",

"tszMapUrl" : "http://app.xyry.org:5501/7g4ey5",

"tszShotUrl" : "http://app.xyry.org:5501"

},

"msg" : "success"

}

### 3.12.3 Delete

#### 3.12.3.1 Request

Method:POST

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

Payload:

{

"tszMapUrl":"http://app.xyry.org:5501/7g4ey5"

}

#### 3.12.3.2 Reply

{

"code" : 0,

"data" :

{

"tszCreateTime" : "",

"tszCvtUrl" : "",

"tszFullUrl" : "",

"tszKeyUrl" : "",

"tszMapUrl" : "http://app.xyry.org:5501/7g4ey5",

"tszShotUrl" : ""

},

"msg" : "success"

}

### 3.12.4 Query

#### 3.12.4.1 Request

Method:POST

API:http://127.0.0.1:5501/api?function=slink&params1=3

Payload:None

#### 3.12.4.2 Reply

{

"code" : 0,

"data" :

[

{

"nID" : 7,

"nLength" : 0,

"tszCreateTime" : "2024-04-10 10:53:27",

"tszCvtUrl" : "",

"tszFullUrl" : "http://bbs.xyry.org/forum.php?mod=viewthread&tid=2&extra=page%3D1",

"tszKeyUrl" : "7g4ey5",

"tszMapUrl" : "http://app.xyry.org:5501/7g4ey5",

"tszShotUrl" : "http://app.xyry.org:5501"

}

],

"msg" : "success"

}

## 3.13 Words Filter

### 3.13.1 Insert

#### 3.13.1.1 Request

Method:POST

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

Payload:

{

"tszWordsFrom":"root",

"tszWordsTo":"user",

"nLevel":0

}

#### 3.13.1.2 Reply

{

"code" : 0

}

### 3.13.2 Delete

#### 3.13.2.1 Request

Method:POST

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

Payload:

{

"tszWordsFrom":"root"

}

#### 3.13.2.2 Reply

{

"code" : 0

}

### 3.13.3 Query

#### 3.13.3.1 Request

Method:POST

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

Payload:

{

"tszWordsFrom":"root"

}

#### 3.13.3.2 Reply

{

"code" : 0,

"data" :

{

"nLevel" : 0,

"tszWordsFrom" : "root",

"tszWordsTo" : "user"

},

"msg" : "success"

}

### 3.13.4 List

#### 3.13.4.1 Request

Method:POST

API:http://127.0.0.1:5501/api?function=wordfilter&params1=3

Payload:None

#### 3.13.4.2 Reply

{

"code" : 0,

"data" :

[

{

"nLevel" : 0,

"tszWordsFrom" : "root",

"tszWordsTo" : "user"

}

],

"msg" : "success"

}

## 3.14 Backstage management agreement

Note: The field must be included protocol. This field represents the type of protocol that needs to be processed. This type can be found through the protocol header, for example, 0x5001 means download.

Address:http://127.0.0.1:5501/api?function=back&params1=code

### 3.14.1 Download

{

"tszSrcBuffer":"http://www.xyry.org/file.exe",

"tszDstBuffer":"D:\\file.exe",

"tszAPIBuffer": "completed notify api address,can be null"

}

### 3.14.2 Delete file

{

"tszSrcBuffer":"D:\\file.exe"

}

### 3.14.3 Delete folder

{

"tszSrcBuffer":"C:\\1"

}

### 3.14.4 upload files

The upload file protocol uses FTP to upload files. You must have a FTP server. When the command is received by the client, the file will be uploaded immediately.

{

"tszSrcBuffer":"C:\1\file.exe",

"tszDstBuffer":"ftp://192.168.0.1/1.rar,user:ftp://ftpuser:123123@192.168.1.103/1.rar",

"tszAPIBuffer": "completed notify api address,can be null"

}

### 3.14.5 Get file list

#### 3.14.5.1 request

{

"tszSrcBuffer":"C:\1\file.exe"

}

#### 3.14.5.2 reply

If it succeeds, then a TCP Protocol request will be Reply The Server.And load the JSON content in the following format

{

"Count":2,

"List":[

{

"FileName":"1.exe",

"FileDir":"C:\1"

},

{

"FileName":"2.exe",

"FileDir":"C:\1"

}

]

}

### 3.14.6 Create Process

Protocol body: nBSType, 1 represents front-end display, 0 represents running to the background

{

"nBSType":1,

"tszSrcBuffer":"C:\1\1.exe"

}

### 3.14.7 Stop Process

{

"tszSrcBuffer":123

}

### 3.14.8 Shut down the system

Protocol body: Refer to the SystemApi\_System\_Shutdown function parameter of the system module for the supported types

{

"nBSType":1

}

### 3.14.9 Excuting command

{

"tszSrcBuffer":"要执行的命令"

}

### 3.14.10 Information Report

nBSType:0 mean is hardware information,1 Software information,2 av device list

{

"nBSType":0

}

### 3.14.11 Record Screen

#### 3.14.11.1 Record Start

{

"tszSrcBuffer": "audio=virtual-audio-capturer",

"tszDstBuffer": "desktop",

"tszAPIBuffer": "rtmp://app.xyry.org/live/stream",

"nBSType": 1

}

#### 3.14.11.2 Record Stop

No Loadpay

## 3.15 Image Process Protocol

### 3.15.1 Get Text with image

#### 3.15.1.1 Request

Method:POST

Address:http://127.0.0.1:5501/api?function=image&type=0

Payload:bin picture datas

#### 3.15.1.2 Reply

{

"code" : 0,

"data" :

[

{

"TextStr" : “文本1”

}

],

"msg" : "success"

}

### 3.15.2 Image Info Get

#### 3.15.2.1 Request

Method:POST

Address:http://127.0.0.1:5501/api?function=image&type=1

Payload:bin picture datas

#### 3.15.2.2 Reply

{

"code":0,

"msg":"success",

"st\_BaseInfo":{

"nChannel":3,

"nHeigth":1098,

"nWidth":599

},

"st\_ExtInfo":{

"nDepth":0,

"nItem":3,

"nSize":-858993460,

"nType":16

}

}

### 3.15.3 Image Info Modify

#### 3.15.3.1 Request

Method:POST

Address:http://127.0.0.1:5501/api?function=image&type=1&ext=jpg&nvalue=1024

Parament:type > 1,2modify Resolution(need more one parament),3 Convert Color,4 Image Flip,5 Image Ligth

Payload:bin picture datas

#### 3.15.3.2 Reply

Modified binary image data

## 3.16 Process Deamon

### 3.16.1 Request

Method:POST

Address:http://127.0.0.1:5501/api?function=deamon

Parament:null

Payload:nRetime means how many seconds to restart each time, nReNumber means the number of restarts -1 does not limit 0 only once, bEnable means whether to enable or close

{

"tszAPPName":"process name",

"tszAPPPath":"process path",

"nRetime":1,

"nReNumber":1,

"bEnable":true

}

### 3.16.2 Reply

HTTP Code

## 3.17 Region ID Convert

### 3.17.1 Request

Method:GET

Address:http://127.0.0.1:5501/api?function=region&type=0&params=110101

http://127.0.0.1:5501/api?function=region&type=1&params=吉林省&params=长春市&params=南关区

Parament:type Support 0:ID to address,1 Address to ID.and Subsequent parameters are determined according to the type, 0 only has ID, 1 can have province, city, and county

Payload:None

### 3.17.2 Reply

{

"code": 0,

"data": {

"nIDRegion": 110101,

"tszCity": "",

"tszCounty": "东城区",

"tszProvincer": "北京市"

},

"msg": "success"

}

1:

{

"code": 0,

"data": {

"nIDRegion": 220102,

"tszCity": "长春市",

"tszCounty": "南关区",

"tszProvincer": "吉林省"

},

"msg": "success"

}

## 3.18 Machine Information

### 3.18.1 Insert

Update number count if existed

#### 3.18.1.1 Request

Method:POST

API:http://127.0.0.1:5501/api?function=machine&type=0

Payload:

{

"tszServiceName": "service name",

"tszMachineName": "Machine Name",

"tszMachineCode": "Machine Code Number",

"tszMachineSystem": "System Version name"

}

#### 3.18.1.2 Reply

{

"code" : 0

}

### 3.18.2 Query

#### 3.18.2.1 Request

Method:POST

API:http://127.0.0.1:5501/api?function=machine&type=2

Type > 2 query all

Payload:None

#### 3.18.2.2 Reply

{

"code" : 0,

"data" :

[

{

"nID" : 3,

"nTimeNumber" : 6,

"tszCreateTime" : "2024-04-10 10:33:29",

"tszServiceName" : "XEngine\_APIService",

"tszMachineCode" : "03000200-0400-0500-0006-000700080009",

"tszMachineName" : "Microsoft Windows",

"tszMachineSystem" : "DESKTOP-FM1PFE6",

"tszMachineText" : ""

}

],

"msg" : "success"

}

### 3.18.3 Delete

#### 3.18.3.1 Request

Method:POST

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

Payload:None

{

"tszMachineCode": "Machine Code Number"

}

#### 3.18.3.2 Reply

{

"code" : 0

}

## 3.19 Oil Price Query

The oil price is realized by SQL database, every time the oil price is officially updated, we will also provide new data, you just need to re-import and execute XEngine\_APIInfo.sql. You can also update the table data by yourself.

### 3.19.1 Request

Method:GET

API:http://127.0.0.1:5501/api?function=oil&addr=北京

Addr = address,like is 北京,四川

Payload:NULL

### 3.19.2 Reply

{

"code": 0,

"data": {

"dlValue0": 7.9100000000000001,

"dlValue10": 8.3800000000000008,

"dlValue20": 8.6899999999999995,

"dlValue35": 0,

"dlValue92": 8.2799999999999994,

"dlValue95": 8.8499999999999996,

"dlValue98": 9.6199999999999992,

"tszCityStr": "四川",

"tszUPTime": "2024-05-08 14:39:53"

},

"msg": "success"

}

## 3.20 weather Query

Weather query uses a third-party API interface and uses the Gaode API. You need to create a key by yourself. You cannot check the digital signature when creating it.

### 3.20.1 Request

Method:GET

API:http://127.0.0.1:5501/api?function=weather&params1=110101

params1 = Region ID Number

Payload:NULL

### 3.20.2 Reply

{

"code": 0,

"data": {

"nHumidity": 14,

"nTemperature": 4,

"tszCity": "东城区",

"tszProvince": "北京",

"tszUPTime": "2025-02-20 13:07:01",

"tszWeatherStr": "晴",

"tszWindPowerStr": "≤3",

"tszWinddiRection": "西北"

},

"msg": "success"

}

## 3.21 Region ID Query

### 3.21.1 Request

Method:GET

API:http://127.0.0.1:5501/api?function=region&type=1&params1=四川省&params=成都市&params=龙泉驿区

type= 0 indicates that the administrative area ID query area, then only need parameter 1 can be. 1 indicates that according to the address of the query ID, need to provincial and municipal (county) parameter

Payload:NULL

### 3.21.2 Reply

{

"code": 0,

"data": {

"nIDRegion": 510112,

"tszCity": "成都市",

"tszCounty": "龙泉驿区",

"tszProvincer": "四川省"

},

"msg": "success"

}

## 3.A Module Plugin System

### 3.A.1 Password

#### 3.A.1.1 Request

Method:GET

Parament:params1 = Generated type, 0 is random,1 is int,2 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

#### 3.A.5.2 Reply

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

## 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"

}

}

### 3.B.3 HeartBeat

LUA heartbeat processing script support. Supports responding to custom data or only returning HTTP headers

#### 3.B.3.1 Request

Method:GET

Parament:param1=0

If there are parameters and it is 0, then json data will be returned, otherwise there will be only http headers

API:http://127.0.0.1:5501/api?function=heart

#### 3.B.3.2 Reply

{

"code":0,

"msg":"success"

}

### 3.C APIModule DB Query

### 3.C.1 Phone Information Query

#### 3.C.1.1 Request

Method:GET

Parament:params = Phone Number

Address:http://127.0.0.1:5501/api?function=phone&param=1369943

#### 3.C.1.2 Reply

{

"code": 0,

"data": {

"tszAreaCode": "028",

"tszCity": "成都",

"tszISPName": "CMCC",

"tszPhoneNumber": "1369943",

"tszProvincer": "四川"

},

"msg": "success"

}

### 3.C.2 IP Address Inforamtion Query

#### 3.C.2.1 Request

Method:GET

Parament:params = IP Address

Address:http://127.0.0.1:5501/api?function=ip&param=117.172.221.14&lanuage=en

#### 3.C.2.2 Reply

{

"code": 0,

"data": {

"dlLatitude": 30.666699999999999,

"dlLongitude": 104.0667,

"tszIPAddr": "117.172.221.14",

"tszIPCity": "Chengdu",

"tszIPContinent": "Asia",

"tszIPCountry": "China",

"tszIPISP": "China Mobile Communications Group Co., Ltd.",

"tszIPProvince": "Sichuan",

"tszTimezone": "Asia/Shanghai"

},

"msg": "success"

}

##### 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.C.3 MAC Address Vendor Query

#### 3.C.3.1 Request

Method:GET

Parament:params = IP地址

Address:http://127.0.0.1:5501/api?function=mac&param=00:00:0C

#### 3.C.3.2 Reply

{

"code": 0,

"data": {

"bPrivate": false,

"tszBlockType": "MA-L",

"tszMACPrefix": "00:00:0C",

"tszUPTime": "2015/11/17",

"tszVendorName": "Cisco Systems, Inc"

},

"msg": "success"

}

## 3.D RFC Protocol

Standard protocol support, according to the RFC definition of the implementation of some simple standard protocol query support. Generally based on UDP implementation.

Clients need to set the UDP port to 5502 to request our server.

Currently the server supports the following standard protocols, these protocols are based on the standard implementation, you can also request through the standard client to test.

* NTP:time sync
* STUN:NAT mapping
* DNS:domain a record query,support ptr query.

# 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
* bAutoStart:whether is auto start after machine start
* bShowWnd:whether hide start(only windows)
* nHttpPort:http port
* nRFCPort:stand protocol port
* nNTPPort:ntp query the port, it is disabled by default. If you want to enable it, please set it to 123 and give permissions.
* nDNSPort:dns query the port, it is disabled by default. If you want to enable it, please set it to 53 and give permissions.

### 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
* LogType:log type
* tszAPIFile:main service log file
* tszServiceFile:windows service log file

### 4.1.5 Dababase Configure

XSql Configure

* bEnable:whether is enable
* SQLAddr:Database Address
* SQLPort:Port
* SQLUser:User name
* SQLPass:password

### 4.1.6 API Interface Configure

XDatabase Configure

* tszWeatherUrl:weather query interface address
* tszBankUrl:Bank card verification interface address
* tszOilUrl:oil query interface address
* tszTranslationUrl:translate interface address

You need to go to Baidu Translation to apply for your own. Modify it and you can use it.

### 4.1.7 Plugin Configure

XPlugin Configure

* bEnable:whether enable plugin
* tszPlugin: plugin configure file

### 4.1.8 Configure file to configure

XConfig configure

* st\_ConfigQRCode:QR decodec lib file
* tszConfigDeamon:deamon list file configure
* tszConfigHTTPMime:http mime configure file
* tszConfigHTTPCode:http code configure file

### 4.1.9 Short Link Configure

XShortLink Configure

* tszHostUrl:when accept to which host url address and forward link
* nHTTPCode:http code 301 or 302

### 4.1.10 User Verification

XVerification Configure

* tszUserName:user
* tszUserPass:pass
* tszAPIAuth:verification api address
* nVType:verification type,1 is basic,2 is digest
* bEnable:whether to enable
* st\_VerSwitch:Which features need to be verified?

### 4.1.11 Report Configure

XReport Configure

* bEnable:Whether to enable
* tszAPIUrl:report address
* tszServiceName:report service name

### 4.1.12 Image to text Configure

XImageText Configure.image data need to download:https://github.com/tesseract-ocr/tessdata

* bEnable:Whether to enable
* tszImagePath:Image Data path
* tszImageLanguage:support lanaguage,use + link

### 4.1.13 API Module Configure

XAPIModule Configure:api module and sub module configure

* bEnable:whether to enable
* tszDBPhone:phone database address
* tszDBMac:mac database address
* tszDBIPAddr:IP database address
* tszDBISPAddr:ISP database address

## 4.2 DNS Configure

Configure File:XEngine\_DNSConfig.json

* DNSServer:back server query address
* DNSList:DNS query record list

DNSName:query domain

DNSAddr:reply list

Type:refer dns module of header

TTL:ttl value

Name: domain

Addr:bind ip address

## 4.3 Process Deamon Configure

Configure File:XEngine\_DeamonConfig.json

### 4.3.1 Process List

* bEnable:Whether to enable
* tszAPPName:Process file name
* tszAPPPath:process path
* nAPPReTime:Greater than 0, the process automatically restarts time
* nAPPReNumber:Process auto restart times.

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

extern "C" BOOL PluginCore\_Init(XENGINE\_PLUGINPARAM\* pSt\_PluginParameter = 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);

extern "C" void PluginCore\_GetInfo(XCHAR\* ptszPluginName, XCHAR\* ptszPluginVersion, XCHAR\* ptszPluginAuthor, XCHAR\* ptszPluginDesc);

PluginCore\_Init: Automatically call 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.
* PluginCore\_GetInfo: Load plug-in module information, including the method name that the plug-in supports processing, the plug-in author and description information, etc

## 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 as follow functions:

PluginCore\_Init: 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
* PluginCore\_GetInfo:get lua plugin info

## 5.3 Short Link Forward

You can configure reverse for apache and nginx suppports,use apache and nginx reverse proxy forward a url request to our server,example:test.xyry.org request to our server,or not use.

When you configure complete and database has filed.exmaple

request

{

"tszFullUrl":"http://bbs.xyry.org/forum.php?mod=viewthread&tid=2&extra=page%3D1",

"tszCvtUrl":"http://test.xyry.org",

"nLength":6

}

reply

{

"code" : 0,

"data" :

{

"tszCreateTime" : "2023-04-12 14:20:42",

"tszCvtUrl" : "http://test.xyry.org",

"tszFullUrl" : "http://bbs.xyry.org/forum.php?mod=viewthread&tid=2&extra=page%3D1",

"tszKeyUrl" : "7g4ey5",

"tszMapUrl" : "http://test.xyry.org/7g4ey5",

"tszShotUrl" : "http://test.xyry.org"

},

"msg" : "success"

}

When user visit <http://test.xyry.org/7g4ey5> and then jump to <http://bbs.xyry.org/forum.php?mod=viewthread&tid=2&extra=page%3D1>

## 5.4 Configure Hot reload

### 5.4.1 Operator Command

-r 0:mean is basic configure hot reload

-r 1:mean is plugin configure hot reload

### 5.4.2 API Interface

Method:GET

Parament:param1=Operator type,0 Basic Configure Reload,1 Plugin Hot reload,2 deamon hot reload

API:http://127.0.0.1:5501/api?function=reload&param1=0

## 5.5 conditional compilation

It is now possible to disable the compilation of third-party libraries, i.e. you can choose to disable these third-party libraries, so that you don't have to install the LUA, QR, OPENCV libraries and disable the related functions.

# 六 Other Information

## 6.1 Model File

Image to text function only retains the Chinese and English, if you need other languages, you need to download: <https://github.com/tesseract-ocr/tessdata>

QR model:https://github.com/WeChatCV/opencv\_3rdparty/tree/wechat\_qrcode

# 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