XEngine Storage Service Docment

目录

[XEngine Storage Service Docment 1](#_Toc5949)

[Preface 4](#_Toc14192)

[Reader 4](#_Toc21750)

[Overview 4](#_Toc23924)

[Associate Module 4](#_Toc20090)

[XEngine 4](#_Toc4821)

[System environment 4](#_Toc6829)

[一 Technical structure 4](#_Toc29403)

[1.1 Upload interface 5](#_Toc4064)

[1.2 Download interface 5](#_Toc10770)

[1.3 Mangement Interface 5](#_Toc8681)

[二 Configure Env 5](#_Toc22667)

[2.1 WINDOWS 5](#_Toc17888)

[2.2 LINUX 5](#_Toc18209)

[三 Interface Protocol 6](#_Toc12494)

[3.1 Management protocol 6](#_Toc4076)

[3.1.1 file list 6](#_Toc12500)

[3.1.2 Insert File 7](#_Toc6984)

[3.1.3 Delete File 8](#_Toc7318)

[3.1.4 dir protocol 8](#_Toc7708)

[3.1.5 TaskList Query Protocol 9](#_Toc29420)

[3.2 third interface 10](#_Toc27456)

[3.2.1 user auth 10](#_Toc21799)

[3.2.2 complete notify 11](#_Toc14647)

[3.3 P2XP Protocol 12](#_Toc6251)

[3.3.1 Join P2xp 12](#_Toc18071)

[3.3.2 Sync List 13](#_Toc20203)

[3.3.3 User Query 14](#_Toc23254)

[3.3.4 Request Connect 15](#_Toc28691)

[3.4 UP and Down 17](#_Toc20051)

[3.4.1 UPLoad 17](#_Toc16092)

[3.4.2 Download 17](#_Toc23509)

[四 Configure Description 17](#_Toc18397)

[4.1 Service Configure 17](#_Toc20184)

[4.1.1 basic configure 17](#_Toc23774)

[4.1.2 Max Configure 17](#_Toc32090)

[4.1.3 Time Configure 18](#_Toc25070)

[4.1.4 Log Configure 18](#_Toc12556)

[4.1.5 Database Configure 18](#_Toc8274)

[4.1.6 Storage Configure 18](#_Toc786)

[4.1.7 Proxy Configure 18](#_Toc13767)

[4.1.8 Limit Configure 19](#_Toc1720)

[4.1.9 P2XP Configure 19](#_Toc10984)

[4.1.10 Cert Configure 19](#_Toc17086)

[4.1.11 Version Configure 19](#_Toc20924)

[4.2 LoadBalance Configure 20](#_Toc13874)

[4.2.1 Basic Configure 20](#_Toc21940)

[4.2.2 Loadbalance Configure 20](#_Toc19695)

[4.2.3 Local Loads 20](#_Toc28839)

[4.2.4 Load Attributes 20](#_Toc2361)

[五 Advanced configuration 21](#_Toc5253)

[5.1 Distributed 21](#_Toc4877)

[5.1.1 Network Distributed 21](#_Toc6109)

[5.1.2 Storage Distributed 21](#_Toc8992)

[5.2 HTTPS 21](#_Toc9074)

[FAQ 22](#_Toc11934)

[appendix 22](#_Toc20833)

[Appendix 1 Type Define 22](#_Toc8751)

[Appendix 2 Protocol Define 22](#_Toc12981)

[Appendix 3 Transformation Definition 22](#_Toc11717)

[Appendix 4 update log 22](#_Toc22104)

|  |  |  |  |
| --- | --- | --- | --- |
| File Status：  [ ] Draft  [√] Release | File Name： | XEngine Storage Service Docment | |
| Be A Version： | V3.6 | |
| Released： | 2022-04-02 | |
| Writer： qyt | | |

# Preface

## Reader

Development ,test

## **Overview**

This document contains related technical descriptions and interface definitions for storage services!

## Associate Module

### XEngine

This service used XEngine as the development package. To use this code, the XEngine development environment must be configured and installed.

download link:<https://gitee.com/xyry/libxengine>

<https://github.com/libxengine/xengine>

## System environment

Windows needs to be in WIN7SP1 and above. Have the latest patch. Use VS to compile and run

Linux requires Ubuntu20.04 or Centos8 or above, using X64. Use makefile to compile

# 一 Technical structure

Use http protocol as the basic communication protocol.upload,download,manage interface are all http.

P2P need tcp protocol,because need keep connection.

Should be bind Four port when Start service Distinguish upload, download, and management,P2P.

Four ports cannot be used across ports

## Upload interface

The upload interface is implemented by HTTP,upload interface need to be implemented using the put of the http.

The upload interface is not used form-data field.

The upload path need to create by user.

Such as: PUT /dir/name HTTP/1.1 get url path....

## Download interface

Download interface is implement through get.

Get Download file through url,like this :GET /bucketkey/dir/name HTTP/1.1

BucketKey is a unique key value encoded by each storage path of distributed storage.

## Mangement Interface

Mangement interface is implement through post,POST can be empty of body,can be contain body of the json.

Manage interface need to provided api/type/name through triplet situation.

Here is:the API version, API type, and API name.

# 二 Configure Env

## 2.1 WINDOWS

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

If environment not have error.complie is succesed.contrain 7 module and 1 exe program

And you need copy file under XEngine\_Release to your complied dir

Note: you can copy VSCopy.bat to your program dir.This script will automatically copy dependent modules file

## 2.2 LINUX

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\_StorageApp file in XEngine\_Release

You can running at terminal.

# 三 Interface Protocol

Our interface are all post protocol.and post port.

## 3.1 Management protocol

### 3.1.1 file list

#### 3.1.1.1 Requestion

Request interface:/Api/Manage/File

Request Method:POST

Request content:empty or follow below:

nMode:0 reply directly,1 reyly p2p

{  
    **"lpszTimeStart"**:**"Start Time,can be NULL"**,  
    **"lpszTimeEnd"**:**"end time,Can be NULL"**,  
    **"lpszFileName"**:**"file name,can be NULL"**,  
    **"lpszFileHash"**:**"file HASH,can be NULL",**

**"nMode"**:0  
}

#### 3.1.1.2 Reply

If the reply tszTableName is 127.0.0.1, it means that the local file exists. The LAN file will not be requested. Otherwise, this value is the remote address + port

{  
    **"Code"**:**0**,  
    **"Count"**:**1**,  
    **"List"**:[  
        {  
            **"nFileSize"**:**23897183**,

**"tszBuckKey"**:**"Bucket Name"**,  
            **"tszFileHash"**:**"85E62C9D28FCE775A68DD126E0519F25"**,  
            **"tszFileName"**:**"1.docx"**,  
            **"tszFilePath"**:**"./XEngine\_File"**,  
            **"tszFileTime"**:**"2021-07-12 05:59:17"**,  
            **"tszFileUser"**:**""**  
        }  
    ],  
    **"Msg"**:**"ok"**,  
    **"lpszTimeEnd"**:**""**,  
    **"lpszTimeStart"**:**""**  
}

### 3.1.2 Insert File

If want p2p network fid the file,you must add the file to the storage service.

Base on P2P FILE Transfer need to record in the database.file download,File download needs to be implemented by the user,so when you download complete a file.you wan the file add p2p network,Then you need to submit a record to the local storage service(Please note: It is not submitted to the server, but the local storage network server).

#### 3.1.2.1 Request

with tszFileName and tszFilePath field Cant be empty,it is file path.other can be empty

Request interface:/Api/Manage/Insert

Request Method:POST

Request content:Only one of bucket and filepath can exist

{  
    **"Count"**:**1**,  
    **"List"**:[  
        {  
            **"nFileSize"**:**23897183**,

**"tszBuckKey"**:**"Bucket Name"**,  
            **"tszFileHash"**:**"85E62C9D28FCE775A68DD126E0519F25"**,  
            **"tszFileName"**:**"1.docx"**,  
            **"tszFilePath"**:**"./XEngine\_File"**,  
            **"tszFileTime"**:**"2021-07-12 05:59:17"**,  
            **"tszFileUser"**:**""**  
        }  
    ]  
}

#### 3.1.2.2 Reply

HTTP CODE 200 = Success

### 3.1.3 Delete File

Delete Request can be delete file,you can submit a delete request to service

#### 3.1.3.1 Request

Can be use file address or just use hash

Request interface:/Api/Manage/Delete

Request Method:POST

Request content:Only one of bucket and filepath can exist

{  
    **"Count"**:**1**,  
    **"List"**:[  
        {  
            **"tszFileHash"**:**"85E62C9D28FCE775A68DD126E0519F25"**,  
            **"tszFileName"**:**"1.docx"**,

**"tszBuckKey"**:**"Bucket Key Name",**  
            **"tszFilePath"**:**"./XEngine\_File"**  
        }  
    ]  
}

#### 3.1.3.2 Reply

HTTP CODE

### 3.1.4 dir protocol

The protocol can be provide folder query, create, delete operations

#### 3.1.4.1 Requestion

Request Interface:/Api/Manage/Diry

Request Method:POST

Request content:

{  
    **"nOPerator"**:**0**,

**"lpszUserDir"**:**"Operator sub dir",**  
    **"lpszBuckKey"**:**"bucket name"**  
}

* 0:Query
* 1:Create
* 2:Delete

#### 3.1.4.2 Reply

Create and delete the standard HTTP message of the reply, query the following content of the reply

{  
    **"Code"**:**0**,  
    **"Count"**:**3**,  
    **"List"**:[  
        {  
            **"tszFilePath"**:**"./XEngine\_File/Scene/scene1"**  
        },  
        {  
            **"tszFilePath"**:**"./XEngine\_File/Scene/scene1/tmp"**  
        },  
        {  
            **"tszFilePath"**:**"./XEngine\_File/Scene/scene2"**  
        }  
    ],  
    **"Msg"**:**"ok"**  
}

### 3.1.5 TaskList Query Protocol

The task query protocol can query all task with uploaded and downloaded by the current storage service, which is convenient for users to make management interfaces.

#### 3.1.5.1 Request

Request Interface:/Api/Manage/Task

Request Method:POST

Request content:nothing

#### 3.1.5.2 Reply

* ullCount:total File size
* ullRWCount:The total size that the client needs to read and write
* ullRWLen:now read and write size
* ullPosStart:The start offset required by the client
* ullPosEnd:The tail offset required by the client

{  
    **"Code"**:**0**,  
    **"Msg"**:**"ok"**,  
    **"DLCount"**:**1**,  
    **"UPCount"**:**1**,  
    **"DLList"**:[  
        {  
            **"tszClientAddr"**:**"client ip address"**,  
            **"tszFilePath"**:**"file path"**,  
            **"ullCount"**:**100**,  
            **"ullRWCount"**:**90**,  
            **"ullRWLen"**:**50**,  
            **"ullPosStart"**:**10**,  
            **"ullPosEnd"**:**100**  
        }  
    ],  
    **"UPList"**:[  
        {  
            **"tszClientAddr"**:**"client ip address"**,  
            **"tszFilePath"**:**"file path"**,  
            **"ullCount"**:**100**,  
            **"ullRWCount"**:**90**,  
            **"ullRWLen"**:**50**,  
            **"ullPosStart"**:**10**,  
            **"ullPosEnd"**:**100**  
        }  
    ]  
}

## 3.2 third interface

Third-interface is used to access the third-party server

### 3.2.1 user auth

User auth just support the http of basic now. You can configure implement user auth through userlist.txt file of mine.and use implement auth by http pass proxy.

#### 3.2.1.1 local auth

Use userlist.txt implement in XEngine\_Config

Each row represents a user.The middle is separated by a space, the front indicates the user name, and the back indicates the password.you only need configure this file to implement http basic.

#### 3.2.1.2 proxy auth

Proxy auth is send http post message implement through service,return 200 is success.other is failure.

Send http post address sent is specified by user.payload is json format.The content is as follows:

{

**"lpszMethod"**:**"http method"**,  
    **"lpszPostUrl"**:**"client post url"**,  
    **"lpszClientAddr"**:**"client is address"**,  
    **"lpszUser"**:**"user"**,  
    **"lpszPass"**:**"pass"**  
}

If sucess.return 200..

### 3.2.2 complete notify

Completion notification means that the server has received an upload and download request after process complete.Whether it is necessary to send an HTTP POST protocol notification to the specified service. It can be configured through the configuration file

Upload and download payload of Completion notifycation is same,follow payload below send to your service.

bDown:1 mean is download,0 mean is upload

{

**"bDown"**:1,

**"lpszBuckKey"**:**"Bucket key name"**,  
    **"lpszFileName"**:**"file name"**,  
    **"lpszFileHash"**:**"file HASH,maybe NULL"**,  
    **"lpszClientAddr"**:**"client ip address"**,  
    **"nFileSize"**:**33333**  
}

## 3.3 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.

P2XP used p2xp port

P2XP protocol use standard protocol(refer to XEngine SDK document)+JSON transfer data

### 3.3.1 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

#### 3.3.1.1 Request

Protocol Header:

wHeader = XENGIEN\_COMMUNICATION\_PACKET\_PROTOCOL\_HEADER

xhToken = 0

unOperatorType = ENUM\_XENGINE\_COMMUNICATION\_PROTOCOL\_TYPE\_P2XP

unOperatorCode = XENGINE\_COMMUNICATION\_PROTOCOL\_OPERATOR\_CODE\_AUTH\_REQLOGIN

unPacketSize = JSONSIZE

byVersion = 2

byIsReply = TRUE

wReserve = 0

wPacketSerial = 0

wTail = XENGIEN\_COMMUNICATION\_PACKET\_PROTOCOL\_TAIL

Protocol Body:

{  
    **"tszUserName"**:**""**,  
    **"tszPrivateAddr"**:**"Private Address,Location IP"**,  
    **"tszPublicAddr"**:**"Public Address"**,  
    **"dwConnectType"**:**0**,  
    **"dwPeerType"**:**0**  
}

#### 3.3.1.2 Reply

Protocol Header:

wHeader = XENGIEN\_COMMUNICATION\_PACKET\_PROTOCOL\_HEADER

xhToken = 0

unOperatorType = ENUM\_XENGINE\_COMMUNICATION\_PROTOCOL\_TYPE\_P2XP

unOperatorCode = XENGINE\_COMMUNICATION\_PROTOCOL\_OPERATOR\_CODE\_AUTH\_REPLOGIN

unPacketSize = JSONSIZE

byVersion = 2

byIsReply = FALSE

wReserve = 0

wPacketSerial = 0

wTail = XENGIEN\_COMMUNICATION\_PACKET\_PROTOCOL\_TAIL

Protocol Body:

{  
    **"nCode"**:**0**,  
    **"lpszMsgBuffer"**:**"sucess"**  
}

### 3.3.2 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

#### 3.3.2.1 Request

Protocol Header:

wHeader = XENGIEN\_COMMUNICATION\_PACKET\_PROTOCOL\_HEADER

xhToken = 0

unOperatorType = ENUM\_XENGINE\_COMMUNICATION\_PROTOCOL\_TYPE\_P2XP

unOperatorCode = XENGINE\_COMMUNICATION\_PROTOCOL\_OPERATOR\_CODE\_P2XP\_REQLANLIST

unPacketSize = JSONSIZE

byVersion = 2

byIsReply = TRUE

wReserve = 0

wPacketSerial = 0

wTail = XENGIEN\_COMMUNICATION\_PACKET\_PROTOCOL\_TAIL

Protocol Body:

{  
    **"tszPublicAddr"**:**"Public Address"**,  
    **"tszPrivateAddr"**:**"Private Addre,if NULL means Get all ip address under the public network"**  
}

#### 3.3.2.2 Reply

Protocol Header:

wHeader = XENGIEN\_COMMUNICATION\_PACKET\_PROTOCOL\_HEADER

xhToken = 0

unOperatorType = ENUM\_XENGINE\_COMMUNICATION\_PROTOCOL\_TYPE\_P2XP

unOperatorCode = XENGINE\_COMMUNICATION\_PROTOCOL\_OPERATOR\_CODE\_P2XP\_REPLANLIST

unPacketSize = JSONSIZE

byVersion = 2

byIsReply = FALSE

wReserve = 0

wPacketSerial = 0

wTail = XENGIEN\_COMMUNICATION\_PACKET\_PROTOCOL\_TAIL

Protocol Body:

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

### 3.3.3 User Query

#### 3.3.3.1 Request

Protocol Header:

wHeader = XENGIEN\_COMMUNICATION\_PACKET\_PROTOCOL\_HEADER

xhToken = 0

unOperatorType = ENUM\_XENGINE\_COMMUNICATION\_PROTOCOL\_TYPE\_P2XP

unOperatorCode = XENGINE\_COMMUNICATION\_PROTOCOL\_OPERATOR\_CODE\_P2XP\_REQUSERQUERY

unPacketSize = JSONSIZE

byVersion = 2

byIsReply = TRUE

wReserve = 0

wPacketSerial = 0

wTail = XENGIEN\_COMMUNICATION\_PACKET\_PROTOCOL\_TAIL

Protocol Body:

{  
    **"tszUserName"**:**"UserName"**  
}

#### 3.3.3.2 Reply

Protocol Header:

wHeader = XENGIEN\_COMMUNICATION\_PACKET\_PROTOCOL\_HEADER

xhToken = 0

unOperatorType = ENUM\_XENGINE\_COMMUNICATION\_PROTOCOL\_TYPE\_P2XP

unOperatorCode = XENGINE\_COMMUNICATION\_PROTOCOL\_OPERATOR\_CODE\_P2XP\_REPUSERQUERY

unPacketSize = JSONSIZE

byVersion = 2

byIsReply = FALSE

wReserve = 0

wPacketSerial = 0

wTail = XENGIEN\_COMMUNICATION\_PACKET\_PROTOCOL\_TAIL

Protocol Body:

{  
    **"nCode"**:**0**,  
    **"lpszMsgBuffer"**:**"sucess"**,  
    **"dwConnectType"**:**1**,  
    **"dwPeerType"**:**1**,  
    **"tszConnectAddr"**:**"192.168.1.101:33990"**,  
    **"tszPrivateAddr"**:**"192.168.1.101"**,  
    **"tszPublicAddr"**:**"114.114.114.114"**,  
    **"tszUserLocation"**:**"china"**,  
    **"tszUserISP"**:**"ISP"**,  
    **"tszUserName"**:**"123123aa"**  
}

### 3.3.4 Request Connect

The request connection protocol can require a client to connect to your specified address and port. This protocol is used for the first protocol requested when transmitting data point-to-point. The server will forward this data to the client after receiving the request connection protocol. , The client will receive the same protocol as the request server. At this time, the client needs to actively connect to the specified IP address and port

We have not formulated a point-to-point transmission protocol, so developers need to implement it, we are only responsible for getting through the data!

#### 3.3.4.1 Request

Protocol Header:

wHeader = XENGIEN\_COMMUNICATION\_PACKET\_PROTOCOL\_HEADER

xhToken = 0

unOperatorType = ENUM\_XENGINE\_COMMUNICATION\_PROTOCOL\_TYPE\_P2XP

unOperatorCode = XENGINE\_COMMUNICATION\_PROTOCOL\_OPERATOR\_CODE\_P2XP\_REQCONNECT

unPacketSize = JSONSIZE

byVersion = 2

byIsReply = TRUE

wReserve = 0

wPacketSerial = 0

wTail = XENGIEN\_COMMUNICATION\_PACKET\_PROTOCOL\_TAIL

Protocol Body:

{  
    **"tszSourceUser"**:**"Source User"**,  
    **"tszDestUser"**:**"Dest User"**,  
    **"tszConnectAddr"**:**"Connect Addr"**,  
    **"nDestPort"**:**5000**,  
    **"bIsTcp"**:**1**  
}

#### 3.3.4.2 Reply

Protocol Header:

wHeader = XENGIEN\_COMMUNICATION\_PACKET\_PROTOCOL\_HEADER

xhToken = 0

unOperatorType = ENUM\_XENGINE\_COMMUNICATION\_PROTOCOL\_TYPE\_P2XP

unOperatorCode = XENGINE\_COMMUNICATION\_PROTOCOL\_OPERATOR\_CODE\_P2XP\_REPCONNECT

unPacketSize = JSONSIZE

byVersion = 2

byIsReply = FALSE

wReserve = 0

wPacketSerial = 0

wTail = XENGIEN\_COMMUNICATION\_PACKET\_PROTOCOL\_TAIL

Protocol Body:

{  
    **"nCode"**:**0**,  
    **"lpszMsgBuffer"**:**"sucess"**  
}

## 3.4 UP and Down

### 3.4.1 UPLoad

Upload can be used post and put method,upload interface need to bucket name by system configure.upload is interface must contain Content-Length field,if you want use resume trans,then need range field.

System return 200 after upload is complete,and return 3.2.2 json payload

### 3.4.2 Download

Download interface use get method.download resume trans is supported

System will be add filehash filed in http header after Request download,it mean is this file is hash code.

# 四 Configure Description

## 4.1 Service Configure

Basic Configure File:XEngine\_Config.json

### 4.1.1 basic configure

* bDeamon: 1 deamon process run 0 is terminal run
* nCenterPort:Business processing port,HTTP POST PORT
* nStorageDLPort:download port,get protocol
* nStorageUPPort:upload port,put protocol
* nP2XPPort:P2XP Protocol Port

### 4.1.2 Max Configure

XMax Configure

* MaxClient Allow Max Client Count
* MaxQueue Allow Max Queue
* IOThread:network io process threads number
* CenterThread:business handle threads number
* nStorageUPThread:upload handle threads number
* nStorageDLThread:download handle threads number
* nP2XPThread:P2XP Number Thread of Process

### 4.1.3 Time Configure

XTime Configure

* bHBTime,1 is enable heartbeat,0 disable
* nDBMonth:database save time,default month
* nTimeCheck:check time
* nStorageTimeOut:how time check once
* nP2XPTimeOut:P2XP Heartbeat timeout

### 4.1.4 Log Configure

XLog Configure

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

### 4.1.5 Database Configure

XSql Configure,The server database uses MYSQL. If the server database is not enabled, the client's SQLITE database is used by default

* SQLType:database enable type.0 not use.1 mysql,2 sqlite
* SQLFile:client db file

### 4.1.6 Storage Configure

XStorage Configure

* nHashMode:HASH algorithm,1 MD5,2 HASH1,you can see openssl define
* bResumable:send mode,1:Use global speed limit mode,2:Use full speed mode
* bUPHash:Whether to force the uploaded file to bring the file hash field

### 4.1.7 Proxy Configure

XProxy Configure

#### 4.1.7.1 Auth Proxy

XProxyAuth Configure

* bAuth:Whether enable auth
* tszUserList: user list address
* tszAuthProxy:use remote auth,auth for http.please read 3.2.1.2,if empty.use local list file.

#### 4.1.7.2 Complete Notify

XProxyPass Configure

* bUPGet:Whether enable upload complete notify
* bDLGet:whether enable download complete notify
* tszUPPass:upload complete notify address
* tszDLPass:download complete notify address

Note: When this is configured, the service will wait for your return result, and return success and failure to the client according to whether it is 200

### 4.1.8 Limit Configure

XLimit Configure

* nDLTry:download retry number
* nMaxUPLoad:Max upload speed..0 unlimited.,unit:(BYTE)
* nMaxDNLoad:Max download speed...

### 4.1.9 P2XP Configure

P2XP Configure:XP2xp

nTime:max process network time,unit second.

nSDPort:send port,use broadcast

nRVPort:recv port,use broadcast

tszQQWryFile:load ip address path

### 4.1.10 Cert Configure

Cert Configure:XCert

bDLEnable:1 Enable Download SSL

bUPEnable:1 Enable UPLoad SSL

bCHEnable:1 Enable Center SSL

nSslType:SSL or TLS,refer XEngine\_OPssl module

tszCertChain:cert chain file

tszCertKey:cert key

### 4.1.11 Version Configure

XVer Configure:Show version for user

## 4.2 LoadBalance Configure

Configure File:XEngine\_LBConfig.json

### 4.2.1 Basic Configure

* bDistributed:Whether enable distributed,if zero,not enable

### 4.2.2 Loadbalance Configure

LBDistributed,configuration for redirection

* nCenterMode:Center Distributed Service Mode,0 no enable,1 random select,2 order select,3 reverse select
* nUPLoadMode:upload distributed service
* nDownldMode:download distributed service
* nStorageMode:storage distributed service

### 4.2.3 Local Loads

LBLocation Configure

* nCenterMode:not used
* nUPLoadMode:storage key selection ,1 random select,2 order select,3 reverse select
* nDownldMode:not used
* nStorageMode:not used

### 4.2.4 Load Attributes

The following content is achieved through redirection,client must support 302 redirect request

* CenterAddr:task processing load address pool
* DownloadAddr:download address pool
* UPLoaderAddr:upload address pool
* StorageAddr:distributed storage

bEnable:Whether to enable

nLevel:storage priority level, the smaller the higher the priority

Size:Allowed size, KB, MB, GB

XEngine\_Key:Bucket key,used to hide the path. Corresponding path

XEngine\_Path:Saved path

# 五 Advanced configuration

## 5.1 Distributed

### 5.1.1 Network Distributed

Distributed services are implemented through HTTP redirection, and the client must support 302 to implement distributed protocols.

Processing distributed through configuration.You can configure one or more distributed processing logic through arrays

When use to distributed.distributed configure file must be close at final

#### 5.1.1.1 Configure Mulit

Distributed Configure:LoadBalance

You can add multiple address to other server with arrays.at

at the time,the server is a distributed server.it is not process your ,It does not process the nUseMode business configuration you specify, but hands it over to multiple backends for processing.

Then the back-end processing needs to set nUseMode to close.

### 5.1.2 Storage Distributed

Storage distributed requires deploying this storage server on your client machine and enabling the SQLITE database.

Now you can query and download files through the lan network. You can request the local storage service first, and the storage service will automatically broadcast and query the files in the lan network. If it exists, it will return you the content of section 3.1.1.2, and will add a field tszTableName to indicate Server IP and port. It will return a list of all queried available local area network nodes, you can choose one download or connect multiple through the HTTP Range field to do distributed block download

## 5.2 HTTPS

After version 3.1, the service supports HTTPS access. To use HTTPS, you need to have a certificate. You can create a certificate by yourself or apply for a free one or buy one.

The certificate use is a certificate chain certificate and a certificate key (like NGINX), without verifying the client certificate. If you have advanced requirements, you can modify the code to use other certificate types or require the client to have a certificate.

# FAQ

1. We are not recommend using upload speed limit. It is best to set it to 0.

# 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