XEngine Storage Service Docment

目录

[XEngine Storage Service Docment 1](#_Toc19349)

[Preface 4](#_Toc14580)

[Reader 4](#_Toc9456)

[Overview 4](#_Toc8039)

[Associate Module 4](#_Toc9390)

[一 Technical structure 4](#_Toc26367)

[1.1 Upload interface 4](#_Toc22797)

[1.2 Download interface 4](#_Toc2450)

[1.3 Mangement Interface 5](#_Toc3816)

[二 Configure Env 5](#_Toc31077)

[2.1 WINDOWS 5](#_Toc19683)

[2.1.1 Configure Environment 5](#_Toc21047)

[2.1.2 complie and run 5](#_Toc22725)

[2.2 LINUX 6](#_Toc32683)

[2.2.1 Evnironment Configure 6](#_Toc8370)

[2.2.2 Complie and Run 6](#_Toc27686)

[2.3 Version Requirements 7](#_Toc28452)

[2.3.1 System Version 7](#_Toc9035)

[2.3.2 Software Version 7](#_Toc18397)

[三 Interface Protocol 7](#_Toc28102)

[3.1 query protocol 7](#_Toc32030)

[3.1.1 file list 7](#_Toc22535)

[3.2 third interface 8](#_Toc18413)

[3.2.1 user auth 8](#_Toc25611)

[3.2.2 upload proxy 9](#_Toc17199)

[3.2.3 complete notify 9](#_Toc26236)

[3.2.4 Network P2P File 9](#_Toc26841)

[3.3 P2XP Protocol 12](#_Toc20899)

[3.3.1 Join P2xp 12](#_Toc27030)

[3.3.2 Sync List 13](#_Toc1274)

[3.3.3 User Query 14](#_Toc14981)

[3.3.4 Request Connect 15](#_Toc11888)

[四 Configure Description 17](#_Toc20990)

[4.1 Service Configure 17](#_Toc29331)

[4.1.1 basic configure 17](#_Toc5000)

[4.1.2 Max Configure 17](#_Toc31547)

[4.1.3 Time Configure 17](#_Toc11357)

[4.1.4 Log Configure 17](#_Toc18546)

[4.1.5 Database Configure 18](#_Toc27954)

[ure 18](#_Toc15783)

[4.1.7 Proxy Configure 18](#_Toc21277)

[4.1.8 Limit Configure 19](#_Toc30063)

[4.1.9 P2XP Configure 19](#_Toc16590)

[4.1.10 Version Configure 19](#_Toc30819)

[4.2 LoadBalance Configure 19](#_Toc12922)

[4.2.1 Basic Configure 19](#_Toc6112)

[4.2.2 Loadbalance Configure 19](#_Toc18980)

[4.2.3 Load Attributes 19](#_Toc29604)

[五 Advanced configuration 20](#_Toc16171)

[5.1 Distributed 20](#_Toc10673)

[5.1.1 Network Distributed 20](#_Toc21314)

[5.1.2 Storage Distributed 20](#_Toc27091)

[appendix 21](#_Toc16452)

[Appendix 1 Type Define 21](#_Toc12983)

[Appendix 2 Protocol Define 21](#_Toc28028)

[Appendix 3 Transformation Definition 21](#_Toc27778)

[Appendix 4 update log 21](#_Toc28763)

|  |  |  |  |
| --- | --- | --- | --- |
| File Status：  [ ] Draft  [√] Release | File Name： | XEngine Storage Service Docment | |
| Be A Version： | V2.1 | |
| Released： | 2021-08-06 | |
| Writer： qyt | | |

# Preface

## Reader

Development ,test

## **Overview**

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

## Associate Module

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

And This service used library for jsoncpp

# 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

Support third-party service interface, support NGINX as upload and download engine.

## 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

## 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

Need to download XEngine.

Complie and run and debug by vs2019.

Download address:https://gitee.com/xengine/libxengine

### 2.1.1 Configure Environment

After Download XEngine,if you download is zip file.you have to uncompress for xengine and add user environment value.

You need add to follow two user environment in your system

* XEngine\_Include  XEngine is header path
* XEngine\_Library  XEngine is library path

比如:



You need jsoncpp env.you can install through vcpkg

And you can download for youself:https://github.com/open-source-parsers/jsoncpp/

If you download for youself,you need install and complie by youself and configure project attributes vc++ path in your vs2019

### 2.1.2 complie and run

When you complete with configuration.you can come in code path.open XEngine\_StorageApp.sln by vs2019.and choice x86(debug or release) or x86(just release) complie

If environment not have error.complie is succesed.contrain seven module and one exe program

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

If not have error,you can see follow the infomation



Note: You can run the program directly, the system will prompt you what you need, you can directly enter the XEngine directory to search.

You can also use the VSCopy.bat script in the source code directory to automatically copy the dependent XEngine modules..

## 2.2 LINUX

### 2.2.1 Evnironment Configure

If you use linux.you must running on ubuntu(20.04) or centos(8.x)...

Download XEngine:git clone git@gitee.com:xyry/libxengine.git

After download complete.you can install xengine it by the shell file.

Execute command:sudo XEngine\_RunEnv.sh -i 3

of course,and need install jsoncpp environment to your system.

Ubuntu:sudo apt install libjsoncpp-devel -y

Centos:sudo dnf install jsoncpp-devel -y

### 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\_StorageApp file in XEngine\_Release

You can running at terminal.if there is no error,you can infomation:



## 2.3 Version Requirements

### 2.3.1 System Version

Minimum version requirements:

WINDOWS: win7 sp1

Ubuntu:20.04

Centos:8.x

### 2.3.2 Software Version

Minimum version requirements:

XEngine:V7.18

JsonCpp:V1.9.4

# 三 Interface Protocol

Our interface are all post protocol.and post port.

## 3.1 query protocol

### 3.1.1 file list

#### 3.1.1.1 Requestion

Query interface:/api/query/file

Payload:empty or follow below:

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

#### 3.1.1.2 Reply

{  
    **"Code"**:**0**,  
    **"Count"**:**1**,  
    **"List"**:[  
        {  
            **"nFileSize"**:**23897183**,  
            **"tszFileHash"**:**"85E62C9D28FCE775A68DD126E0519F25"**,  
            **"tszFileName"**:**"1.docx"**,  
            **"tszFilePath"**:**"./XEngine\_File"**,  
            **"tszFileTime"**:**"2021-07-12 05:59:17"**,  
            **"tszFileUser"**:**""**  
        }  
    ],  
    **"Msg"**:**"ok"**,  
    **"lpszTimeEnd"**:**""**,  
    **"lpszTimeStart"**:**""**  
}

## 3.2 third interface

Third-interface is used to access the third-party server.now supported nginx is nginx upload module and nginx download proxy

### 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:

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

If sucess.return 200..

### 3.2.2 upload proxy

Nginx upload file implement by nginx upload module.after your install and configure complete.you need configure proxy\_pass <http://192.168.1.7:5000/Api/Event/UPFile;> and Point to our service.the service can be accepted request and return .

### 3.2.3 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..

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

### 3.2.4 Network P2P File

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).

#### 3.2.4.1 Add File

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

##### 3.2.4.1.1 Request

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

POST:/Api/Manage/Add

{  
    **"Count"**:**1**,  
    **"List"**:[  
        {  
            **"nFileSize"**:**23897183**,  
            **"tszFileHash"**:**"85E62C9D28FCE775A68DD126E0519F25"**,  
            **"tszFileName"**:**"1.docx"**,  
            **"tszFilePath"**:**"./XEngine\_File"**,  
            **"tszFileTime"**:**"2021-07-12 05:59:17"**,  
            **"tszFileUser"**:**""**  
        }  
    ]  
}

##### 3.2.4.1.2 Reply

HTTP CODE 200 = Success

#### 3.2.4.2 Delete File

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

##### 3.2.5.2.1 Request

Can be use file address or just use hash

POST:/Api/Manage/Del

{  
    **"Count"**:**1**,  
    **"List"**:[  
        {  
            **"tszFileHash"**:**"85E62C9D28FCE775A68DD126E0519F25"**,  
            **"tszFileName"**:**"1.docx"**,  
            **"tszFilePath"**:**"./XEngine\_File"**  
        }  
    ]  
}

##### 3.2.5.2.2 Reply

HTTP CODE

#### 3.2.4.3 Request File

Request distributed files,Used to request a list of files in the local area network. This is an HTTP GET request

##### 3.2.4.3.1 Request

<http://www.xyry.org/File> HASH Value

##### 3.2.4.3.2 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"**:**92674736**,  
            **"tszFileHash"**:**"EC9B9B75A04F3B323EFD348F9B795539"**,  
            **"tszFileName"**:**"qq.exe"**,  
            **"tszFilePath"**:**"./XEngine\_File/scene1"**,  
            **"tszFileTime"**:**"2021-07-30 11:01:04"**,  
            **"tszFileUser"**:**""**,  
            **"tszTableName"**:**"127.0.0.1"**  
        }  
    ],  
    **"Msg"**:**"ok"**  
}

## 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"**  
}

# 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,support the database of mysql or sqlite

If you want to enable p2p features.please set the client databse to sqlite

* SQLType:database enable type.0 not use.1 mysql,2 sqlite
* SQLFile:just useful sqltype is 2

### ure

XStorage Configure

* nHashMode:HASH algorithm,1 MD5,2 HASH1,you can see openssl define
* bRename:Whether to automatically change the name and path, valid for files uploaded by NGINX
* tszFileDir:Save Dir

### 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

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

### 4.1.9 P2XP Configure

P2XP Configure:XP2xp

nMode:0 not enable,1 use broadcase mode,2 use Straight mode,if you service is mainly,must be set 0

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 Version Configure

XVer Configure:Show version for user

## 4.2 LoadBalance Configure

Configure File:XEngine\_LBConfig.json

### 4.2.1 Basic Configure

* tszIPAddr:Service IP Address
* bDistributed:Whether enable distributed,if zero,not enable

### 4.2.2 Loadbalance Configure

LBConfig

* nServerMode:Distributed Service Mode,1 Random Selection

### 4.2.3 Load Attributes

Make sure content of Load balance by nUseMode

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

* nUseMode:0,by self,1 upload.2 download,3 center
* CenterAddr:task processing load address pool
* DownloadAddr:download address pool
* UPLoaderAddr:upload address pool

# 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

# 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