XEngine服务文档

目录

[XEngine服务文档 1](#_Toc32675)

[前言 3](#_Toc12354)

[阅读者 3](#_Toc19068)

[概述 3](#_Toc15034)

[相关模块 3](#_Toc9136)

[一 技术结构 4](#_Toc6256)

[1.1 TCP 4](#_Toc21817)

[1.2 HTTP 4](#_Toc4079)

[二 配置环境 4](#_Toc1549)

[2.1 WINDOWS 4](#_Toc1184)

[2.1.1 配置环境 4](#_Toc11027)

[2.1.2 编译运行 4](#_Toc17002)

[2.2 LINUX 5](#_Toc18626)

[2.2.1 环境配置 5](#_Toc19279)

[2.2.2 编译运行 5](#_Toc21214)

[2.3 版本要求 5](#_Toc24253)

[2.3.1 系统版本 5](#_Toc8939)

[2.3.2 软件需求 5](#_Toc28854)

[三 接口协议 5](#_Toc834)

[3.1 TCP 5](#_Toc14083)

[3.1.1 示例 6](#_Toc31744)

[3.2 HTTP 6](#_Toc25852)

[3.2.1 请求 6](#_Toc14362)

[3.2.2 回复 7](#_Toc12987)

[四 配置说明 7](#_Toc19546)

[4.1 服务器配置 7](#_Toc29355)

[4.1.1 基本配置 7](#_Toc14107)

[4.1.2 最大配置 7](#_Toc20694)

[4.1.3 时间配置 7](#_Toc26355)

[4.1.4 日志配置 7](#_Toc29025)

[五 高级配置 8](#_Toc18358)

[附录 8](#_Toc4110)

[附录1 类型定义 8](#_Toc4206)

[附录2 协议定义 8](#_Toc4296)

[附录3 转换定义 8](#_Toc8767)

[附录4 更新历史 8](#_Toc12666)

|  |  |  |  |
| --- | --- | --- | --- |
| 文件状态：  [ ] 草稿  [√] 正式发布 | 文件名称： | XEngin服务文档 | |
| 适用版本： | V1.0.0.1001 | |
| 发布日期： | 2022-03-02 | |
| 拟 制： qyt | | |

# 前言

## 阅读者

开发人员,测试人员,QA

## **概述**

此文档包含了存储服务的相关技术说明和接口定义!

## 相关模块

此服务使用了XEngine作为开发包.要使用此服务代码必须配置安装好XEngine开发环境.

# 一 技术结构

服务器基于C/C++实现,协议基于TCP和HTTP实现

## TCP

TCP协议使用了XEngine的标准协议头来实现消息通信.TCP协议速度更快,功能更多.

## HTTP

消息通过Get或者Post方法

# 二 配置环境

## 2.1 WINDOWS

需要下载XEngine.

通过VS来编写代码,编译,调试.

下载地址:https://gitee.com/xyry/libxengine

https://github.com/libxengine/xengine

### 2.1.1 配置环境

按照XEngine Readme文件的说明配置环境

### 2.1.2 编译运行

在完成环境配置后.你可以进入代码目录.使用VS打开XEngine.sln,然后选择x86 debug 编译.

如果环境没有错误,编译会直接成功

成功后,你需要拷贝XEngine\_Release下的文件到你编译的目录下.运行XEngine\_\*App.exe即可.

注意：你可以直接运行程序，系统会提示你需要哪些，你直接进入XEngine目录搜索.也可以使用VSCopy-\*.bat脚本拷贝依赖模块

## 2.2 LINUX

### 2.2.1 环境配置

如果使用LINUX来开发运行,那么你需要在UBUNTU或者CENTOS下面才可以使用,UBUNTU需要20.04 以上系统.CENTOS需要8.x版本(兼容所有类centos版本)

### 2.2.2 编译运行

配置完环境后,你可以编译它,打开终端,进入源码目录执行以下命令

编译:make

安装:make FLAGS=InstallAll

清理:make FLAGS=CleanAll

如果没有错误,你可以在XEngine\_Release目录下看到编译的XEngine\_\*App

## 2.3 版本要求

### 2.3.1 系统版本

最低版本要求:

WINDOWS: win7 sp1

Ubuntu:20.04

Centos:8.x

### 2.3.2 软件需求

最低版本要求:

XEngine:V7.30

# 三 接口协议

## 3.1 TCP

请注意:协议头的wReserver字段将作为服务器回复处理结果使用.为0表示处理成功,其他值表示失败!  
 协议负载无论是通用协议还是HTTP协议,他们负载的内容都可以负载JSON.具体参考协议说明.

### 3.1.1 示例

#### 3.1.1.1 请求

协议头:

wHeader = XENGIEN\_COMMUNICATION\_PACKET\_PROTOCOL\_HEADER

xhToken = 0

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

unOperatorCode = XENGINE\_COMMUNICATION\_PROTOCOL\_OPERATOR\_CODE\_MQ\_REQPOST

unPacketSize = MSGLEN

byVersion = 1

byIsReply = TRUE

wReserve = 0

wPacketSerial = 0

wTail = XENGIEN\_COMMUNICATION\_PACKET\_PROTOCOL\_TAIL

协议体:DATA数据

#### 3.1.1.2 回复

协议头:

wHeader = XENGIEN\_COMMUNICATION\_PACKET\_PROTOCOL\_HEADER

xhToken = 0

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

unOperatorCode = XENGINE\_COMMUNICATION\_PROTOCOL\_OPERATOR\_CODE\_MQ\_REPPOST

unPacketSize = 0

byVersion = 1

byIsReply = FALSE

wReserve = 0

wPacketSerial = 0

wTail = XENGIEN\_COMMUNICATION\_PACKET\_PROTOCOL\_TAIL

## 3.2 HTTP

HTTP目前支持POST和GET请求

### 3.2.1 请求

方法:

地址:

### 3.2.2 回复

# 四 配置说明

## 4.1 服务器配置

基础配置文件:XEngine\_Config.json

### 4.1.1 基本配置

* tszIPAddr: 本机IP地址
* bDeamon: 1为守护进程后台运行0为前台运行
* nCenterPort:TCP通信端口
* nHttpPort:HTTP通信端口

### 4.1.2 最大配置

XMax 配置

* nMaxClient 最大允许客户端
* nMaxQueue 最大允许队列
* nIOThread:网络IO处理线程个数
* nCenterThread:TCP处理线程个数
* nHttpThread:HTTP处理线程个数

### 4.1.3 时间配置

XTime配置

* nTimeCheck:检测次数
* nCenterTimeOut:TCP客户端过期时间
* nHttpTimeOut:HTTP客户端过期时间

### 4.1.4 日志配置

XLog 配置

* MaxSize:日志文件大小
* MaxCount:日志文件个数
* LogLeave:允许保存的级别

# 五 高级配置

# 附录

## 附录1 类型定义

参考文件 XEngine\_CommHdr.h

## 附录2 协议定义

参考文件XEngine\_ProtocolHdr.h

## 附录3 转换定义

参考文件 XEngine\_Types.h

## 附录4 更新历史

|  |  |  |  |
| --- | --- | --- | --- |
| 更新日期 | 更新说明 | 修改者 | 适用版本 |