XEngine代理服务器文档

目录

[XEngine代理服务器文档 1](#_Toc26387)

[前言 3](#_Toc29728)

[阅读者 3](#_Toc24496)

[概述 3](#_Toc10949)

[相关模块 3](#_Toc3685)

[一 技术结构 3](#_Toc2651)

[1.1 目录结构 3](#_Toc429)

[1.2 源码目录 3](#_Toc31500)

[1.3 协议解释 4](#_Toc19996)

[二 配置环境 4](#_Toc16335)

[2.1 WINDOWS 4](#_Toc28864)

[2.2 LINUX 4](#_Toc14271)

[2.3 版本要求 4](#_Toc20862)

[2.3.1 系统版本 4](#_Toc892)

[2.3.2 软件需求 4](#_Toc19840)

[三 配置说明 5](#_Toc30316)

[3.1 基础配置 5](#_Toc12782)

[3.2 最大配置 5](#_Toc15114)

[3.3 时间配置 5](#_Toc4481)

[3.4 日志配置 5](#_Toc2728)

[四 通信协议 5](#_Toc29629)

[4.1 Tunnel 5](#_Toc7250)

[4.2 Socks 6](#_Toc4809)

[4.3 Forwad 6](#_Toc19377)

[4.3.1 登录 6](#_Toc17730)

[4.3.2 枚举 7](#_Toc16416)

[4.3.3 绑定 7](#_Toc23427)

[4.4.4 转发 8](#_Toc23890)

[附录 9](#_Toc13357)

[附录1 更新历史 9](#_Toc21047)

|  |  |  |  |
| --- | --- | --- | --- |
| 文件状态：  [ ] 草稿  [√] 正式发布 | 文件名称： | XEngine代理服务器文档 | |
| 适用版本： | V1.4.0.1001 | |
| 发布日期： | 2022-06-17 | |
| 拟 制： qyt | | |

# 前言

## 阅读者

开发人员,测试人员

## **概述**

此文档包含了代理服务器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.

请参考Readme配置环境

# 一 技术结构

采用的是TCP协议来作为基础通信协议

服务器启动会绑定两个端口,用来处理协议.

## 目录结构

* XEngine\_APPClient:应用例子代码目录
* XEngine\_Docment:文档目录
* XEngine\_Release:编译发布目录
* XEngine\_Source:源代码目录

## 源码目录

* XEngine\_ModuleAuthorize:用户验证模块
* XEngine\_ModuleConfigure:配置文件模块
* XEngine\_ServiceApp:服务程序
* XEngine\_ThirdPart:三方库

## 协议解释

TCP私有协议采用的是XEngine定义的协议,你可以参考XEngine的协议文档详细了解协议的定义与解释.

# 二 配置环境

## 2.1 WINDOWS

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

如果环境没有错误,编译会直接成功.包含5个模块和一个EXE程序

成功后,你需要拷贝XEngine\_Release下的文件到你编译的目录下.然后拷贝依赖的XEngine模块到你的编译目录下,运行XEngine\_ServiceApp.exe即可

注意：XEngine环境可以通过VSCopy脚本拷贝到你的编译目录下,前提是你配置好你的XEngine环境.

## 2.2 LINUX

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\_ServiceApp

## 2.3 MacOS

MacOS需要12以及以上版本,编译和运行方式可以参考Linux.

## 2.4 版本要求

### 2.4.1 系统版本

最低版本要求:

WINDOWS: win7 sp1

Ubuntu:20.04

Centos:8.x

MacOS:12

### 2.4.2 软件需求

最低版本要求:

XEngine:V7.38

# 三 配置说明

配置文件:XEngine\_Config.json

## 3.1 基础配置

* tszIPAddr:服务器IP地址
* bDeamon:是否守护进程运行
* nSocksPort:SOCKS代理服务端口
* nTunnelPort:HTTP隧道代理服务端口

## 3.2 最大配置

XMax 配置

* MaxClient 最大允许客户端
* MaxQueue 最大允许队列
* IOThread:网络IO处理线程个数

## 3.3 时间配置

XTime配置

* nTimeCheck:检测次数
* nSocksTimeOut:SOCKS心跳检测超时时间
* nTunnelTimeOut:Tunnel心跳检测超时时间

## 3.4 日志配置

配置信息:XLog

* nMaxSize:最大日志文件大小
* nMaxCount:文件备份个数
* nLogLeave:日志级别
* tszLogFile:日志保存路径

# 四 通信协议

## 4.1 Tunnel

基于HTTP的TUNNEL标准协议,请参考相关RFC文档

## 4.2 Socks

基于SOCKSV5的标准协议,请参考相关RFC文档

## 4.3 Forwad

转发协议是私有的定制协议,可以通过这个协议获取可以转发的用户和列表,并且转发数据.

### 4.3.1 登录

不登录无法获取得到列表信息

#### 4.3.1.1 请求

协议头:

wHeader = XENGIEN\_COMMUNICATION\_PACKET\_PROTOCOL\_HEADER

xhToken = 0

unOperatorType = *ENUM\_XENGINE\_COMMUNICATION\_PROTOCOL\_TYPE\_AUTH*

unOperatorCode = XENGINE\_COMMUNICATION\_PROTOCOL\_OPERATOR\_CODE\_FORWARD\_LOGREQ

unPacketSize = 0

byVersion = 0

byIsReply = TRUE

wReserve = 0

wPacketSerial = 0

wTail = XENGIEN\_COMMUNICATION\_PACKET\_PROTOCOL\_TAIL

#### 4.3.1.2 回复

协议头:wReserve表示成功还是失败,0等于成功

wHeader = XENGIEN\_COMMUNICATION\_PACKET\_PROTOCOL\_HEADER

xhToken = 0

unOperatorType = *ENUM\_XENGINE\_COMMUNICATION\_PROTOCOL\_TYPE\_AUTH*

unOperatorCode = XENGINE\_COMMUNICATION\_PROTOCOL\_OPERATOR\_CODE\_FORWARD\_LOGREP

unPacketSize = 0

byVersion = 0

byIsReply = FALSE

wReserve = 0

wPacketSerial = 0

wTail = XENGIEN\_COMMUNICATION\_PACKET\_PROTOCOL\_TAIL

### 4.3.2 枚举

#### 4.3.2.1 请求

协议头:

wHeader = XENGIEN\_COMMUNICATION\_PACKET\_PROTOCOL\_HEADER

xhToken = 0

unOperatorType = ENUM\_XENGINE\_COMMUNICATION\_PROTOCOL\_TYPE\_USER\_FORWARD

unOperatorCode = XENGINE\_COMMUNICATION\_PROTOCOL\_OPERATOR\_CODE\_FORWARD\_LISTREQ

unPacketSize = 0

byVersion = 0

byIsReply = TRUE

wReserve = 0

wPacketSerial = 0

wTail = XENGIEN\_COMMUNICATION\_PACKET\_PROTOCOL\_TAIL

#### 4.3.2.2 回复

协议头:wReserve表示成功还是失败,0等于成功

wHeader = XENGIEN\_COMMUNICATION\_PACKET\_PROTOCOL\_HEADER

xhToken = 0

unOperatorType = ENUM\_XENGINE\_COMMUNICATION\_PROTOCOL\_TYPE\_USER\_FORWARD

unOperatorCode = XENGINE\_COMMUNICATION\_PROTOCOL\_OPERATOR\_CODE\_FORWARD\_LISTREP

unPacketSize = JSONSIZE

byVersion = 0

byIsReply = FALSE

wReserve = 0

wPacketSerial = 0

wTail = XENGIEN\_COMMUNICATION\_PACKET\_PROTOCOL\_TAIL

协议体:

{  
    **"Count"**:**2**,  
    **"Array"**:[  
        **"192.168.1.5:1000"**,  
        **"192.168.1.6:1001"**  
    ]  
}

### 4.3.3 绑定

绑定一个转发客户端在请求后,另一端也会收到同样的请求.另一端收到的请求只有协议头.

#### 4.3.3.1 请求

协议头:

wHeader = XENGIEN\_COMMUNICATION\_PACKET\_PROTOCOL\_HEADER

xhToken = 0

unOperatorType = ENUM\_XENGINE\_COMMUNICATION\_PROTOCOL\_TYPE\_USER\_FORWARD

unOperatorCode = XENGINE\_COMMUNICATION\_PROTOCOL\_OPERATOR\_CODE\_FORWARD\_BINDREQ

unPacketSize = JSONSIZE

byVersion = 0

byIsReply = TRUE

wReserve = 0

wPacketSerial = 0

wTail = XENGIEN\_COMMUNICATION\_PACKET\_PROTOCOL\_TAIL

{  
    **"tszDstAddr"**:**"192.168.1.6:1000"**  
}

#### 4.3.3.2 回复

协议头:wReserve表示成功还是失败,0等于成功

wHeader = XENGIEN\_COMMUNICATION\_PACKET\_PROTOCOL\_HEADER

xhToken = 0

unOperatorType = *ENUM\_XENGINE\_COMMUNICATION\_PROTOCOL\_TYPE\_*FORWARD

unOperatorCode = XENGINE\_COMMUNICATION\_PROTOCOL\_OPERATOR\_CODE\_FORWARD\_BINDREP

unPacketSize = 0

byVersion = 0

byIsReply = FALSE

wReserve = 0

wPacketSerial = 0

wTail = XENGIEN\_COMMUNICATION\_PACKET\_PROTOCOL\_TAIL

### 4.4.4 转发

如果绑定成功,以后的数据都会转发到相应的客户端

# 附录

## 附录1 更新历史

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