V2ray启动器V5.1 制作方案

程序实现目的： 提供十分简单的翻墙服务

程序具备的功能：

1. 自动化更新
2. 自动下载本体
3. 启动核心程序
4. 配置运行环境
5. 用户权限验证
6. 特殊用户启动定制

GitHub项目位置：<https://github.com/lucycore/V2ray.git>

项目分支：

master----------主分支

develop---------lucycore 的开发分支

develop\_2------- JasonC 的开发分支

注：切勿于主分支发布更新，请在自己的分支完成开发后 申请合并分支

程序实现：Lucycore & JasonC

方案版本：V2.0

创建日期：2019-1-20

更新日期：2019-2-18-4-10

客户端逻辑实现：

服务端逻辑实现：

客户端代码目录结构：

—— main.py

—— mods

|—— server.py

|—— update.py

|—— aess.py

|—— install.py

|—— script.py

|—— strat.py

|—— delold.py

|—— user.py

|—— \_\_init\_\_.py

main.py--------程序主函数

mods----------模块文件夹

server.py-------用于与服务器进行连接传输数据的模块

update.py------用于更新启动程序的模块

aess.py--------用于执行aes加密的模块

install.py-------用于安装v2ray本体文件的模块

script.py-------用于执行定制的可执行文件的模块

delold.py------用于删除旧版本v2ray本体的模块

user.py--------用于操作本地user文件的模块

\_\_init\_\_.py------用于访问不同级目录的模块

客户端运行目录结构（Windows）：

—— C:\pythonz5.1

|—— sun36x64

|—— v2ray

|—— V.zip

|—— user

|—— user.json

客户端运行目录结构（Mac OS）：

—— 运行目录/pythonz5.1

|—— sun36x64

|—— v2ray

|—— V.zip

|—— user

|—— user.json

user.json----用于存储加密过后的uuid

V.zip-------v2ray本地压缩包 解压后为v2ray 文件夹

服务器代码目录结构：

—— main.py

—— key.json

—— userk.json

—— mods

|—— client.py

|—— validate.py

|—— \_\_init\_\_.py

main.py--------程序主函数

key.json--------用于存储密钥的库

userk.json------用于存储用户信息的库

mods----------模块文件夹

client.py--------与客户端进行连接的模块

validate.py------用于验证用户身份权限等核心模块

（服务器安装python3.4环境，无需打包可执行文件）

数据存储标准：

user.json 存储加密过后的 uuid

程序将生成的uuid例：“5d0d3870-b704-41ff-988b-2d57544bcbfd”

去除字符串中的“-”，生成一个存着5个随机字符的列表

将列表循环aes加密为5段等长的密文存入 user.json

key.json 用于存储待使用的key

格式：使用json 哈希表存储 哈希表键为key 值为激活的时间

实例：{“key”:7, ”key\_2”:30}

Userk.json 用于存储用户的信息

格式：使用json 哈希表存储 哈希表键为用户uuid 值为到期时间

实例：{“710f02e772144faba1eeef38b6b4b362”:”2019-2-4-12-15”}

项目端口预留：

2233 主程序

2333 特殊程序验证预留

服务器信息：  
 ip：'www.lucycore.top'

端口： 2233

客户端与服务器对话：

——更新

客户端 服务器

“update” 🡪

🡨 目前程序版本号

客户端与服务器对话：

——注册模式（模式一）

客户端 服务器

“mod\_1”🡪

🡨 发送占位符

“uuid” 🡪

🡨 发送占位符

“key” 🡪

🡨 “ok&no”

If ok 占位 🡪

🡨 配置文件信息

Over

客户端与服务器对话：

——普通模式（模式二）

客户端 服务器

“mod\_2”🡪

🡨 发送占位符

“uuid” 🡪

🡨 “ok&no”

If ok 占位 🡪

🡨 配置文件信息

Over

配置文件信息由 ip地址 端口号 和uuid组成

发送时组合为：ip\*port\*uuid

由\*号分界 组合发送