

## XNat 内网穿透核心库

XNat 内网穿透由 3 个库组成

xNatClient.pas, 内网穿透的代理客户端（内网服务器）

xNatService.pas, 内网穿透的服务器（公网服务器）

xNatPhysics.pas, 通讯引擎定义, 在 zDefine.inc 中定义 XNat 使用哪种通讯引擎, 只能在 Delphi 编译器下有效（CrossSocket,DIOCP,ICS,Indy），如果是 FPC，默认通讯引擎为 Synapse

注意：

- 1, XNAT 支持断线重连。
- 2, XNAT 是收发的裸包, 如果对安全有要求, 可以使用 CoreCipher 做并发加密。
- 3, XNAT 是内网代理流量路线的穿透, 不是打洞
- 4, 使用 XNat 内网核心库必须自己编写后台程序, 没有现成的。开发工艺请尽量选择 Console（linux 支持）。参考 Demo 在 Examples 中找 " 24.XNat 内网穿透库的使用"

核心技术介绍：

整个 XNAT 所使用的核心技术就是 p2pVM, 因为 p2pVM 是 ZServer4D 独有的隧道技术, 它的特点就是在一个隧道中支持协议栈, 并且它非常稳定, 高效。即使在最恶劣的后台网络接口中（比如 Indy,Synapse,ICS），它也能带起 100 万的内部链接。

开发 XNAT 的点点滴滴

第一步, 我先架设了一套基于 P2PVM 隧道的通讯协议, 这是 CS 的, 我直接抄写了 ZServer4D 自带的 p2pVM 的 Demo 代码, 这一步, 我花了以秒为单位的 2 位数时间。

第二步, 定义数据结构, 我需要把服务器侦听的协议端口和客户端对上号, 花了 1 整天才写完, 并且测试通过。因为起初, 我使用 Record 结构实现, 后来, 我把数据结构改成了 class, 这便是大家看到的 TXServiceListen, TXClientMapping, 两个类, 因为测试 FPC 时, 发现它对 Record 的支持有些问题, 出在定义方法回调上。

第三步, 定义通讯协议, 最开始, 我计划使用带有反馈的 SendStreamCMD 收发状态, 直接使用 SendDirectBuffer 来收发数据, 后来, 我一想, SendStreamCMD 会等待一个反馈, 在高流量下, 这是很废的机制, 于是, 我把 SendStreamCMD 全部改成 SendDirectStreamCMD。然后, 周末冒着小孩哭闹+老婆的歇斯里底, 经过了 2 天调试, 终于把通讯协议理顺。

第四步, 这时候, XNAT 开发已经差不多了, 因为最开始 XNAT 是计划做成工具, 后来我又改成了代码库, 因为工具也是拿给人用, 我还要负责使用设计, 还要弄文档, 弄技术支持。不如做成开源代码库, 让大家自行开发自己的内网穿透。于是, 把工具部分砍了, 包装成符合人类理解的代码库。这一步, 我又花了 1 天。

第五步, 已经把 XNAT 作为代码包装完成了。使用 DLink+两台笔记本+一台高配 PC, 做了一个压力测试, 测试很简单, 就是一个 http 循环, 不断地访问, 断线。整个测试持续 5 小时, 平均每分钟 1 万次请求。在这过程中, 出现了若干极小概率出现的 bug, 顺手 fixed。这一步, 我花了半天。

到今天，开发 XNAT 刚好花了 5 天（等同于 1 周），Pascal 阵营终于有了规范的内网穿透支持库，以后我们有空慢慢给它增加新功能吧。

By qq600585

2018-9-16