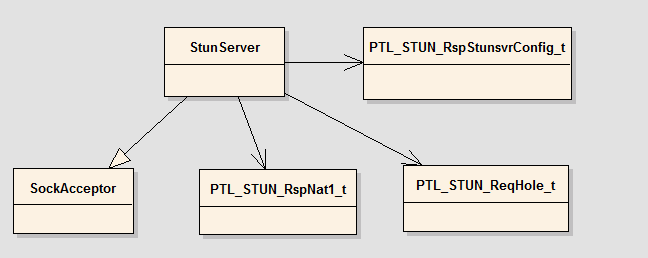
Stun server主要负责探测nat的类型，转发peer请求打洞的消息，主要StunServer模块组成。



## 1 StunServer

StunServer:与peer端StunClient相对应的server端，也是最主要的类；

SockAcceptor：接收UDP消息的接口；

### 1.1主要成员变量

StunServer \*m\_stun2； //

PTL\_STUN\_RspStunsvrConfig\_t m\_local\_config;

PTL\_STUN\_RspStunsvrConfig\_t m\_remote\_config;

sockaddr\_in nat1\_addr;

PTL\_STUN\_Header\_t head;

PTL\_STUN\_RspBindingAddr\_t rsp\_binding;

PTL\_STUN\_RspStunBAddr\_t rsp\_stunb;

PTL\_STUN\_RspStunsvrConfig\_t rsp\_config;

PTL\_STUN\_RspNat1\_t rsp\_nat1;

PTL\_STUN\_RspNat2\_t rsp\_nat2;

PTL\_STUN\_RspNat4\_t rsp\_nat4;

PTL\_STUN\_ReqHole\_t req\_hole;

PTL\_STUN\_RspHole\_t rsp\_hole;

PTL\_STUN\_ReportNattype\_t inf\_nattype;

PTL\_STUN\_ReportConnSF\_t inf\_connsf;

PTL\_STUN\_ReportConnStat\_t inf\_connstat;

### 1.2主要成员函数

stun\_check\_config：请求获取对方stun的配置；

handle\_input：接收UDP数据的线程，接收UDP的数据，并交与on\_line函数处理；

on\_line：处理接收到的消息，消息的种类可见协议中关于命令的定义；

on\_ptl\_request\_binding\_addr：对于peer端请求获取自己公网ip的响应；

on\_ptl\_request\_stunb\_addr：对于peer端请求获取stunB的响应；

on\_ptl\_request\_stunsvr\_config：请求对方stun的配置；

on\_ptl\_response\_stunsvr\_config：响应获取stun的配置；

on\_ptl\_request\_nat1：对于peer端探测nat1的响应，转发stunB2；

on\_ptl\_response\_nat1：收到stunA1发的nat1探测，发往请求探测nat1的peer端；

on\_ptl\_request\_nat4：对于peer请求nat4探测的响应，A1，A2，A3，A4返回的地址及port有一个不同就为nat4；

on\_ptl\_request\_nat2：对于peer请求nat2探测的响应，使用stunA2响应；

on\_ptl\_request\_hole：对于peer请求打洞的响应，所请求打洞的消息转发给对方peer；

on\_ptl\_response\_hole：对于对方peer收到stun转发的请求打洞消息的响应，向对方peer发响应打洞的包；

on\_ptl\_report\_nattype：对于peer上报nat类型的处理；

on\_ptl\_report\_connsf：对于peer上报连接信息的处理；

on\_ptl\_report\_connstat：对于peer上报连接状态的处理；

ptl\_request\_stuns\_config：stunA（stunB）向stunB（stunA）请求获取StunB（stunA）的配置；