**2017年终总结**

我从5月3号开始来公司上班写起，之前的就略过了。

第一个任务是写一个基于TCP的多对一个通信小程序，权当练习。主要使用Qt来写界面，其他的用c++实现。程序分为客户端和服务端。客户端通过ip和port建立tcp链接和服务器进行，界面就是一个简化版的QQ聊天界面；服务端由一个线程进行接受客户端链接的处理，另一个线程进行消息的接收和显示，另一个线程进行消息的发生处理，同时在发送消息时可以选择发送给特定的客户端。这就使我了解了一点点qt知识，在此之前我从没有接触过qt。

第二个任务是关于NM3000的双机热备的部分处理程序(linux平台)。任务包括主备节点状态的切换、相关数据文件由主到备的同步、ip在主备机之间的快速切换(被外界感知)。首先，节点的状态变换主要依靠主备机之间主动的发送tcp心跳包来解决，处理的条件比较多，这里就不多说了；第二个，相关数据文件的同步，在主端相关文件夹中的文件或目录只要存在增、删和改都需要3到5秒内被同步到备注相同的目录中，基本的每一个文件的大小都在70M左右；对文件目录的变化监控采用了Linux的inotify功能；文件的同步部分，本来是自己写的多线程文件传输，可是测了很多次，不太稳定，效果不是和好(现在来看，当时处理的不太好，太菜)，最后无奈使用Linux中的高级IO函数，同样也是通过tcp链接进行文件传输。最后，同一个ip要在主备机之间进行随着切换进行来回飘移，主备机上配置相同的ip对于业务端口，在主备状态进行切换时，在由备到主的机器上，使用arping命令主动发送ARP包，以便外界可以快速更新arp缓存，来达到ip可以在新的机器上快速被使用的目的；最主要的是不能通过开关网卡的动作来实现，这样的话，10s都不一定能让外界知道ip已经到了一台机机器上；正确的做法是网卡正常启动着，只不过备端的业务网卡的ip全部都被置为0，当切换时将业务ip设置到网卡即可，达到3s备外界感知的目的。

第三个任务是让ghc1000linux版在NM3000这个机器上运行，但是失败了，很可能是因为NM3000本来就是一个简化定制版的linux版，它的大小只有10M,和正常的系统不同。

第三个任务PDS\_ser的代码，主要了解历史数据文件的入库操作、相关数据文件产生的原因。写Web\_DataSend.exe程序，将实时数据文件也解析后使用tcp链接通过正向隔离发送给外网的实时数据缓存库；将pds\_ser产生的历史数据文件使用tcp链接通过正向隔离传输外网机器上，然后再由外网的程序将历史数据文件解析，放进外网的数据库中，提供给web界面使用；写Web\_DataCenter.exe程序，这个程序我主要写了Web\_HstData.dll组件，主要功能是接收内网通过正向隔离器发送的历史数据文件和解析收到的历史数据文件并且放入外网的数据库中。经过一段时间时间的测试，发现网络故障时程序中的tcp链接反应缓慢，同样的道理给所有的tcp链接都加上了心跳，这样就可以快速的响应网络的变化。为了降低程序的耦合性，也为了后来的重用，所以准备将之前的程序进行拆分处理。此处所有的程序都是linux和Windows都有的。本来以为这一切都和快的，没想到唉唉唉。