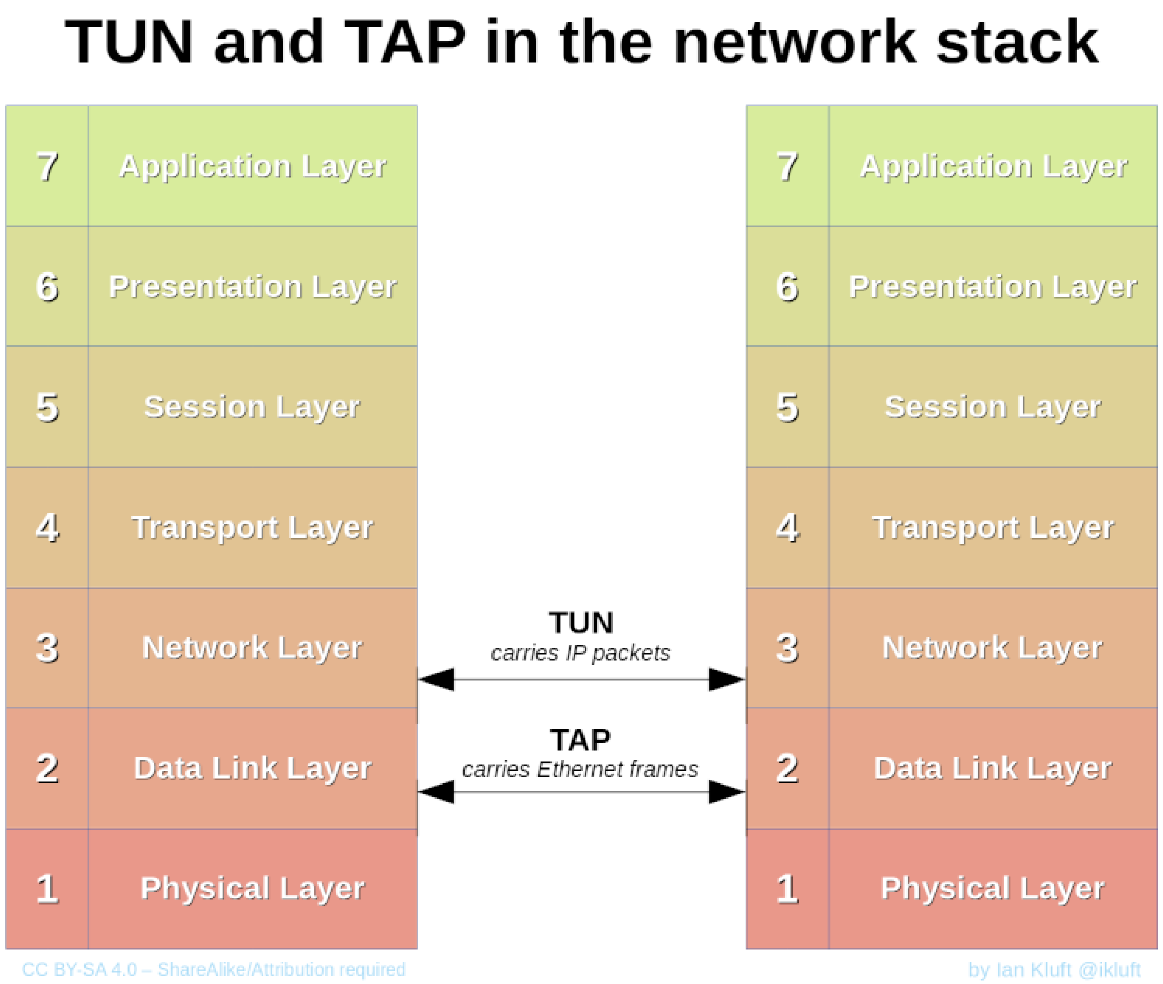
VpnService简介

以ToyVpn为例

什么是VpnService

- 安卓提供给开发者的用户级的VPN服务的接口
- 创建一个虚拟网卡（TUN）
- 返回给APP一个"文件描述符"（File descriptor）
- 工作在网络层（收发IP数据包）



TUN and TAP in the network stack

7	Application Layer		7	Application Layer
6	Presentation Layer		6	Presentation Layer
5	Session Layer		5	Session Layer
4	Transport Layer		4	Transport Layer
3	Network Layer	TUN *carries IP packets*	3	Network Layer
2	Data Link Layer	TAP *carries Ethernet frames*	2	Data Link Layer
1	Physical Layer		1	Physical Layer

by Ian Kluft @ikluft

TUN/TAP provides packet reception and transmission for user space programs. It can be seen as a simple Point-to-Point or Ethernet device, which,instead of receiving packets from physical media, receives them from user space program and instead of sending packets via physical media writes them to the user space program.

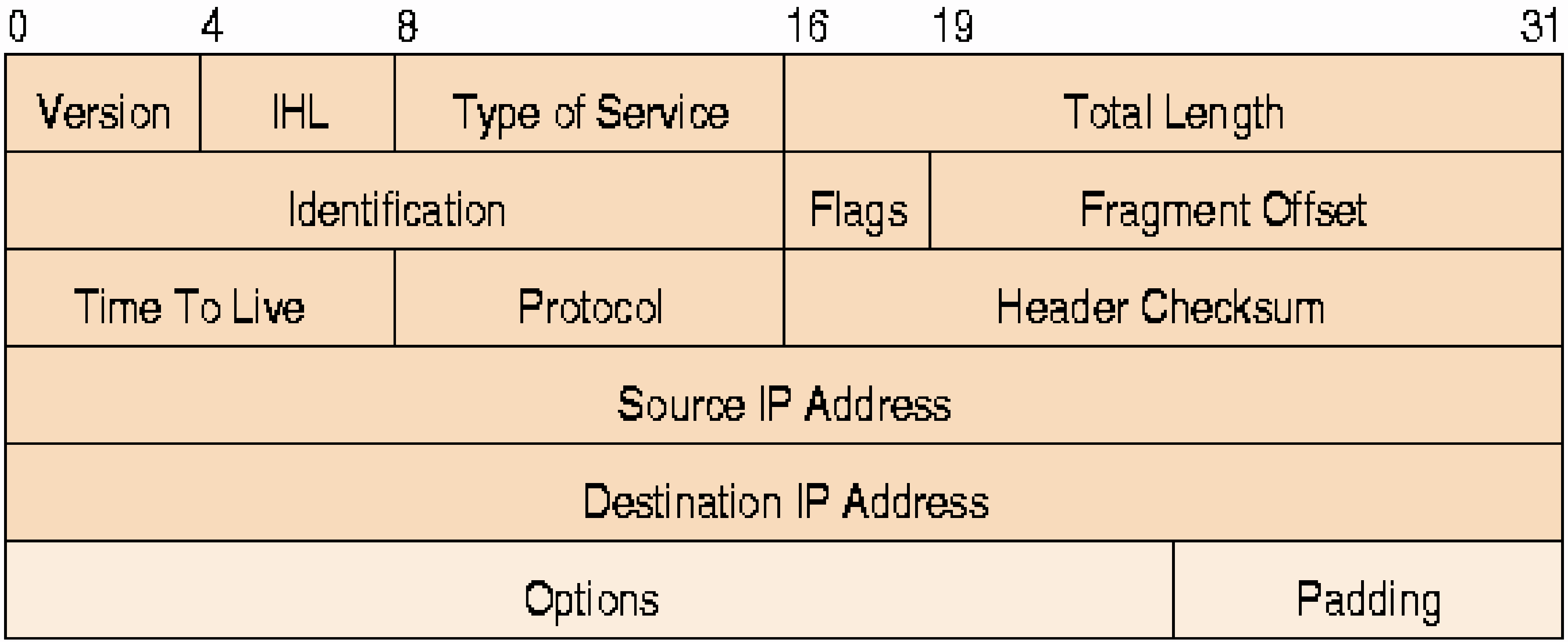
文件描述符（File descriptor）

> In Unix and related computer operating systems, a file descriptor (FD, less frequently fildes) is an abstract indicator (handle) used to access a file or other input/output resource, such as a pipe or network socket. File descriptors form part of the POSIX application programming interface. A file descriptor is a non-negative integer, generally represented in the C programming language as the type int (negative values being reserved to indicate "no value" or an error condition).

```c
int fd;
int fd=open("file1.c",O_RDWR);
write(fd,"Hello world!",sizeof("Hello world!"));
```

IP数据包

!



<table>
<tr><th>0</th><th>4</th><th>8</th><th>16</th><th>19</th><th>31</th></tr>
<tr><td>Version</td><td>IHL</td><td>Type of Service</td><td colspan="3">Total Length</td></tr>
<tr><td colspan="3">Identification</td><td>Flags</td><td colspan="2">Fragment Offset</td></tr>
<tr><td colspan="2">Time To Live</td><td>Protocol</td><td colspan="3">Header Checksum</td></tr>
<tr><td colspan="6">Source IP Address</td></tr>
<tr><td colspan="6">Destination IP Address</td></tr>
<tr><td colspan="4">Options</td><td colspan="2">Padding</td></tr>
</table>

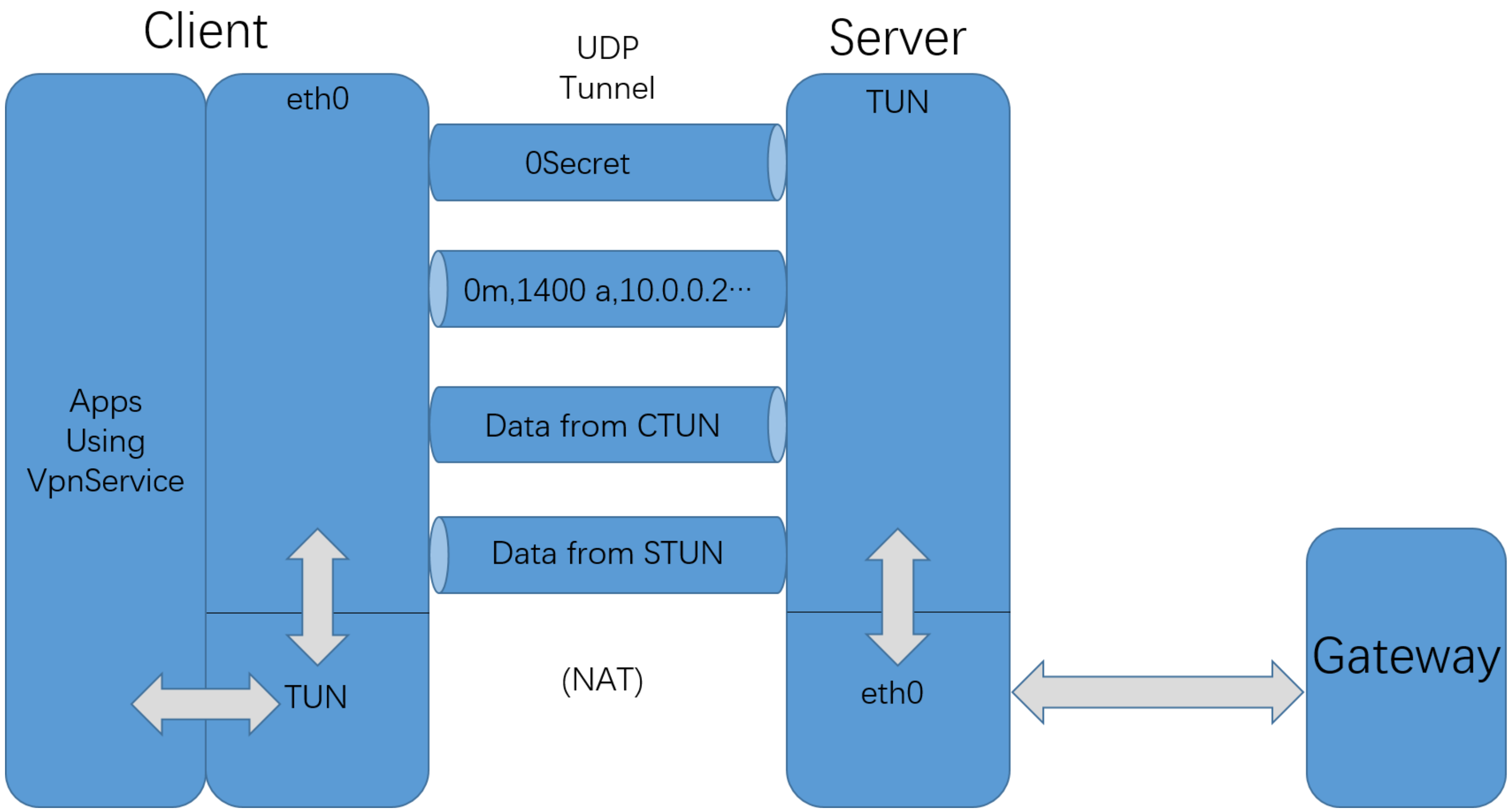
使用**VpnService**的 *套路*

1. VpnService.prepare()
2. Builder builder = new Builder()
3. mInterface = builder.setSession().addAddress().addDnsServer() .addRoute().establish()
4. FileInputStream in = new FileInputStream(mInterface.getFileDescriptor())
5. FileOutputStream out = new FileOutputStream(mInterface.getFileDescriptor());
6. in.read()
7. out.write()

什么是**ToyVpn**

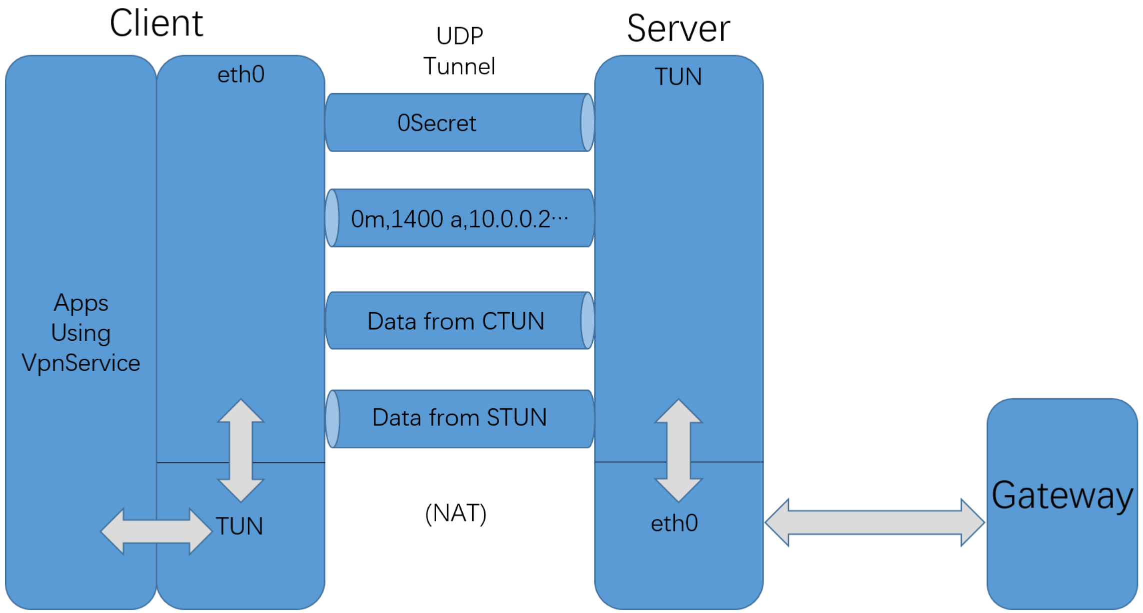
- 安卓官方的例程
- 演示如何用VpnService class构建VPN client
- 实现**IP over UDP**的隧道
- 包含用Java实现的客户端和C实现的服务器
- 使用VPN服务的app流量通过隧道由服务器代理

ToyVpn的通信过程



其他问题

- NAT
- 处理超时

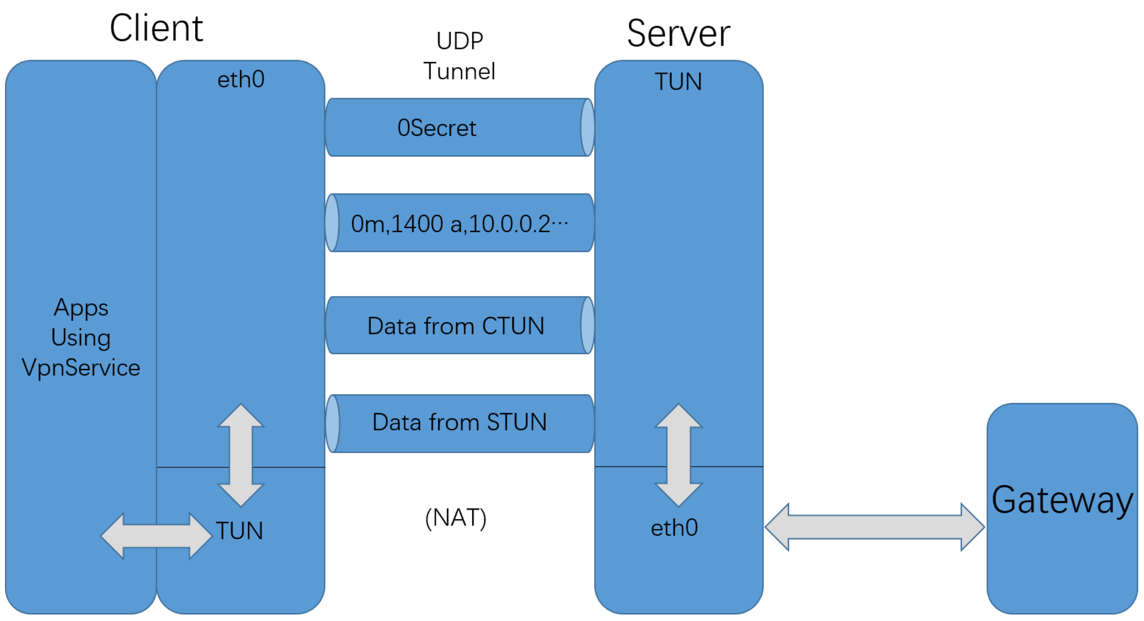


NAT(Network Address Translation)

在IP数据包通过路由器或防火墙时重写来源IP地址或目的IP地址的技术

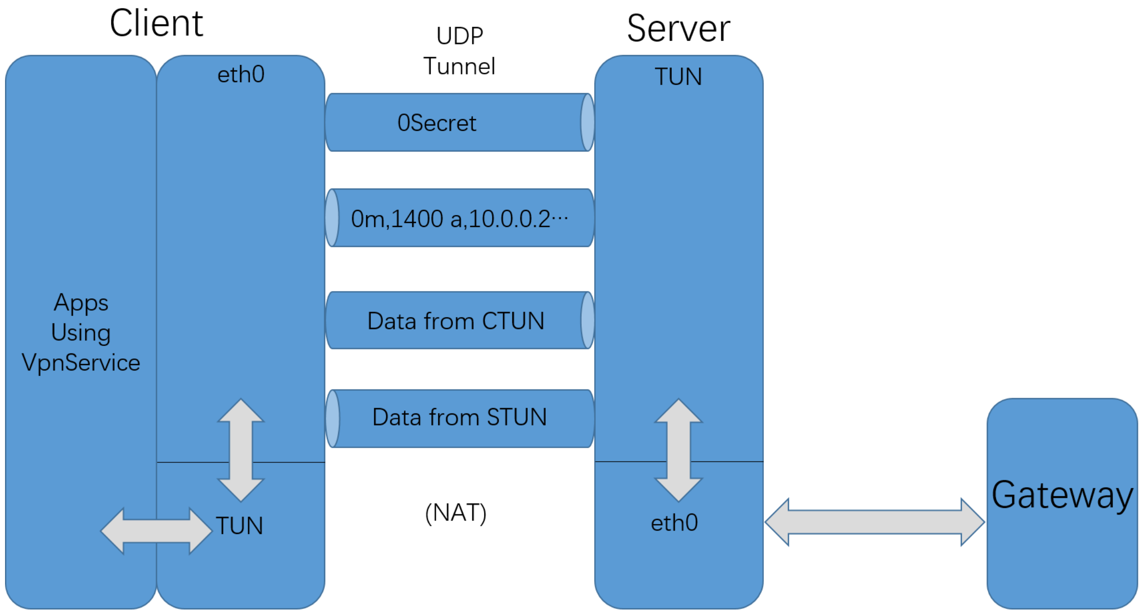
```
iptables -t nat -A POSTROUTING -s 10.0.0.0/8 -o eth0 -j MASQUERADE
```

不是所有的网卡名都是eth0



客户端的超时问题

- lastSendTime
- lastReceiveTime
- （UDP Tunnel）太久不发，则发0
- （UDP Tunnel）太久收不到数据，认为连接中断



服务器的超时问题

- timer表征收发状态，同时表征收发间隔
- timer += (timer > 0) ? 100 : -100;
- （UDP Tunnel）太久（timer < -16000）不发，则发0
- （UDP Tunnel）太久(timer > 20000)收不到数据，认为连接中断

谢谢🙏！！