**使用C#和SharpPcap开发直播App抓包工具获取直播地址**

本文将介绍如何使用[C#语言](https://so.csdn.net/so/search?q=C%23%E8%AF%AD%E8%A8%80&spm=1001.2101.3001.7020)和SharpPcap包开发一个视频直播抓包工具。通过该工具，用户可以捕获视频直播流量，并对其进行分析和监控。本文将详细阐述开发过程，包括环境搭建、包的安装和配置、代码编写等。

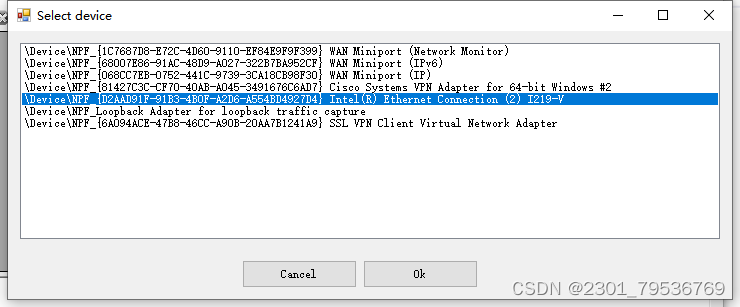
1. 环境搭建  
   在开始开发之前，需要确保已经安装了以下环境：

* Visual Studio 2019或更新版本
* .NET Framework 4.5或更新版本
* SharpPcap包

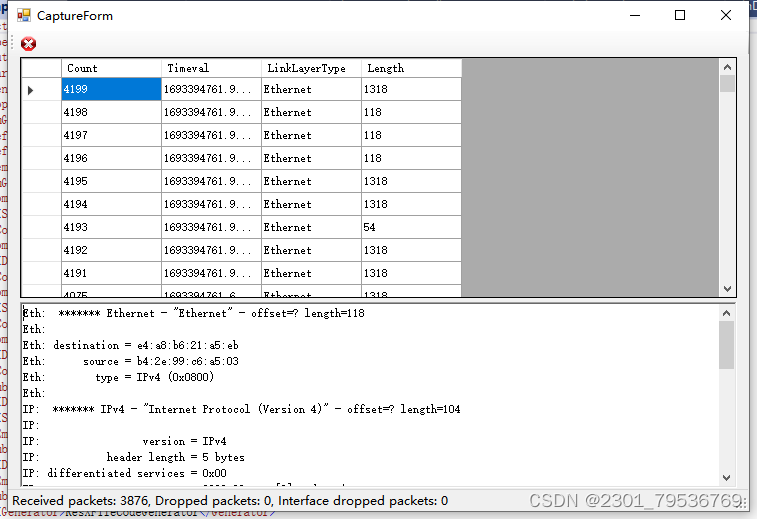
1. 安装和配置SharpPcap包  
   首先，打开Visual Studio并创建一个新的C#控制台应用程序项目。然后，通过NuGet包管理器安装SharpPcap包。在Visual Studio中，打开“工具”菜单并选择“NuGet包管理器”->“管理解决方案的NuGet程序包”。在NuGet包管理器中搜索“SharpPcap”，并安装最新版本。
2. 编写代码  
   在程序的入口点Main方法中，添加以下代码：

1. using System;
2. using SharpPcap;
3. using PacketDotNet;
5. class Program
6. {
7. static void Main(string[] args)
8. {
9. *// 获取网络设备列表*
10. var devices = CaptureDeviceList.Instance;
12. *// 选择一个网络设备*
13. var device = devices[0];
15. *// 打开设备*
16. device.Open(DeviceMode.Promiscuous);
18. *// 设置过滤器，只捕获视频直播流量*
19. device.Filter = "udp and port 1935";
21. *// 开始捕获数据包*
22. device.OnPacketArrival += new PacketArrivalEventHandler(OnPacketArrival);
23. device.StartCapture();
25. Console.WriteLine("开始捕获视频直播流量，请按任意键停止...");
26. Console.ReadKey();
28. *// 停止捕获数据包*
29. device.StopCapture();
30. device.Close();
31. }
33. *// 数据包到达事件处理程序*
34. private static void OnPacketArrival(object sender, CaptureEventArgs e)
35. {
36. var packet = Packet.ParsePacket(e.Packet.LinkLayerType, e.Packet.Data);
37. var udpPacket = (UdpPacket)packet.Extract(typeof(UdpPacket));
39. if (udpPacket != null)
40. {
41. *// 处理视频直播数据包*
42. Console.WriteLine("捕获到视频直播数据包：源IP地址={0}，目标IP地址={1}，源端口={2}，目标端口={3}",
43. udpPacket.SourceAddress, udpPacket.DestinationAddress, udpPacket.SourcePort, udpPacket.DestinationPort);
44. }
45. }
46. }

软件截图：

选择网卡：        

监控记录：



附上程序: [传送门](https://pan.baidu.com/s/1l96PwKs5n5mEi_6r87zCTQ?pwd=5r29)