jNetPcap

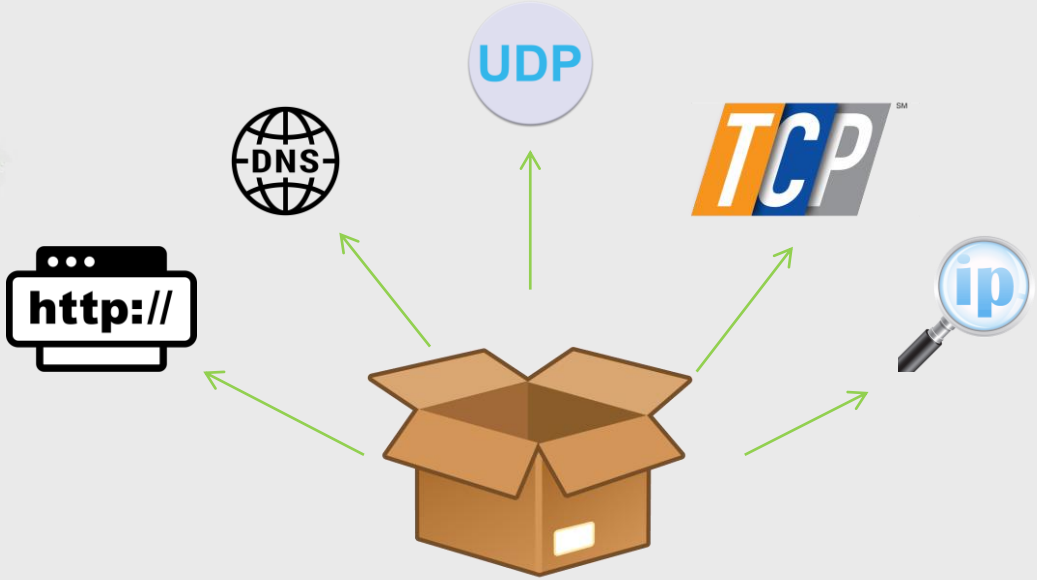
Emirhan KIYAK
Sakarya Üniversitesi
Bilgisayar Mühendisliği

jNetPcap Nedir?

jNetPcap, ağ içindeki paket alışverişinin içeriğini görüntülememizi ve bu bilgileri kendi geliştireceğimiz uygulamalarımızda kullanmamızı sağlayan açık kaynak kodlu bir Java kütüphanesidir.

Paket nedir?

- Bahsettiğimiz gibi jNetPcap, ağ trafiğinde paket yakalıyor. Peki nedir bu paket?

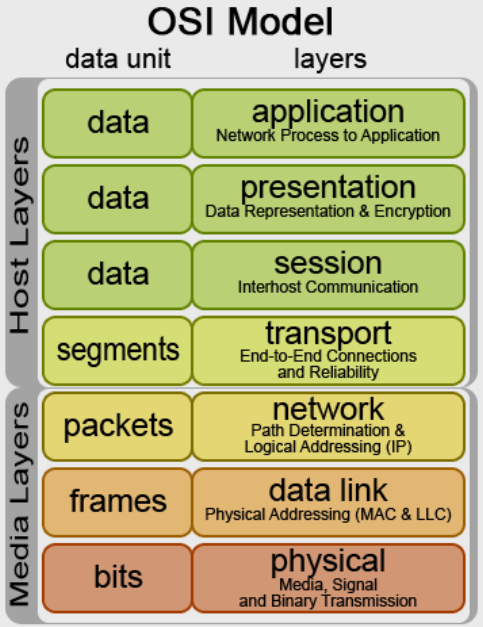


Paket Nedir?

- İçerisinde protokol bilgisi, hedef ve kaynak ip adresi, port bilgisi vs. gibi bilgileri barındıran bilgi kümesine paket denir.
- Paketler ise birleşerek asıl veriyi oluşturur.
- Bu verinin küçük bilgi kümelerine (paket) ayrılması ve iletilmesi ise OSI referans modeline göre gerçekleştirilir.

OSI REFERANS MODELİ

- OSI referans modeli, 7 katmandan oluşan yapısıyla iki ağ ortamı arasındaki veri iletişiminin nasıl olacağını tanımlayan bir çerçevedir



PROTOKOLLER

HTTP, DNS, SMTP, FTP…

ISO 8822, ISO 8823…

NFS, SMB, ISO 8326…

TCP, UDP, SCTP…

IP, IPV4, IPV6, ICMP, ARP…

ETHERNET, Wi-Fi, TokenRing…

RS-232, ISDN…

- https://upload.wikimedia.org/wikipedia/commons/2/2b/Osi-model.png
- https://tr.wikipedia.org/wiki/OSI_modeli

jNetPcap ve WinPcap Kurulumu

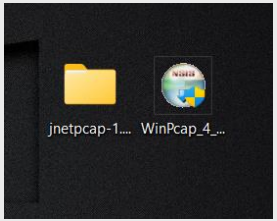
- Önce jNetPcap kütüphanesini cihazımıza indirelim.
- Daha sonra WinPcap uygulamasını cihazımıza indirelim ve kuralım.
- WinPcap, jNetPcap kütüphanesinin işlevini gerçekleştirebilmesi için bize gerekli altyapıyı sağlayacak olan, yani bu verileri kütüphaneyle kullanabilmemizi sağlayan program.

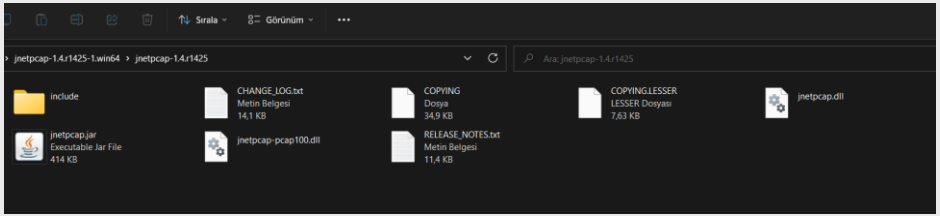
jNetPcap

https://sourceforge.net/projects/jnetpcap/

WinPcap

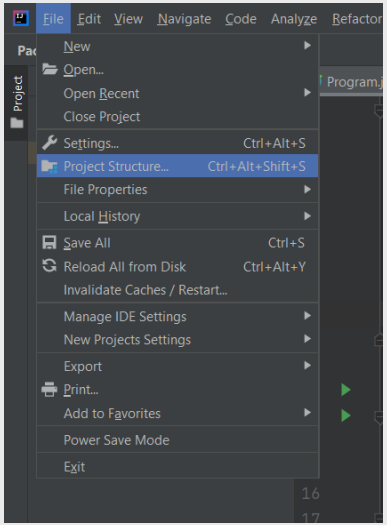
https://www.winpcap.org/install/





jNetPcap' in Java'ya Entegrasyonu

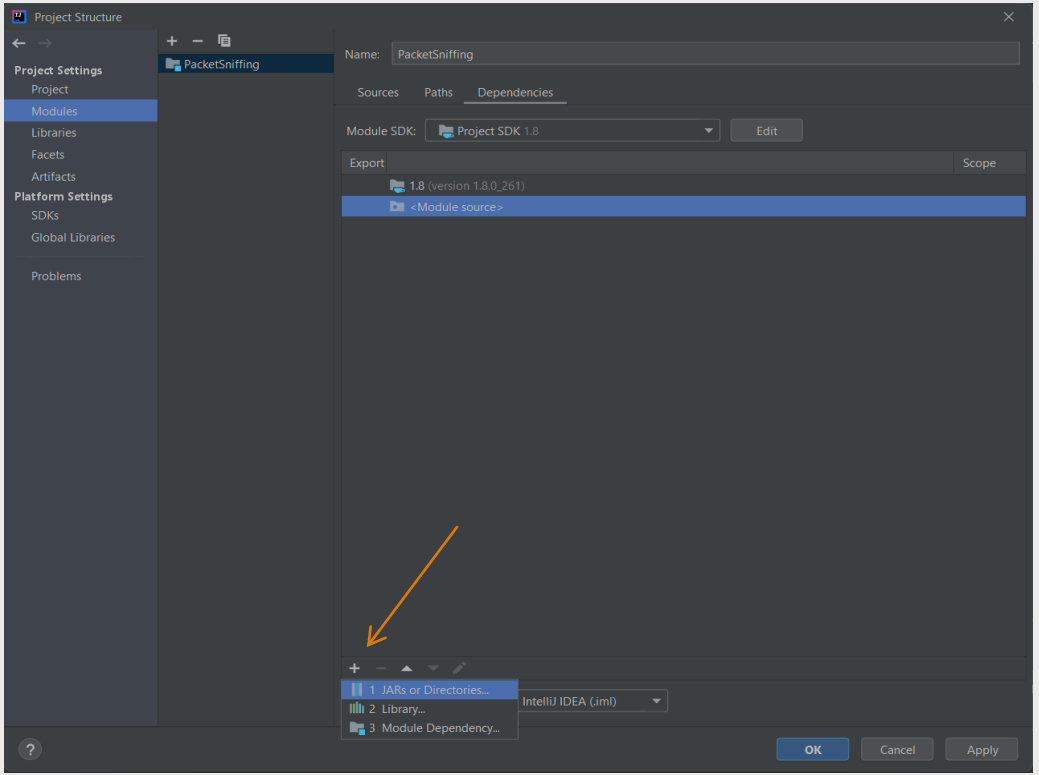
- Proje ekranında File kısmından Project Structure' ı açalım



Not : Kütüphane ekleme işlemi IntellijIdea' da gerçekleştirilmiştir. Farklı geliştirme ortamlarında da benzer mantıkta eklenebilir.

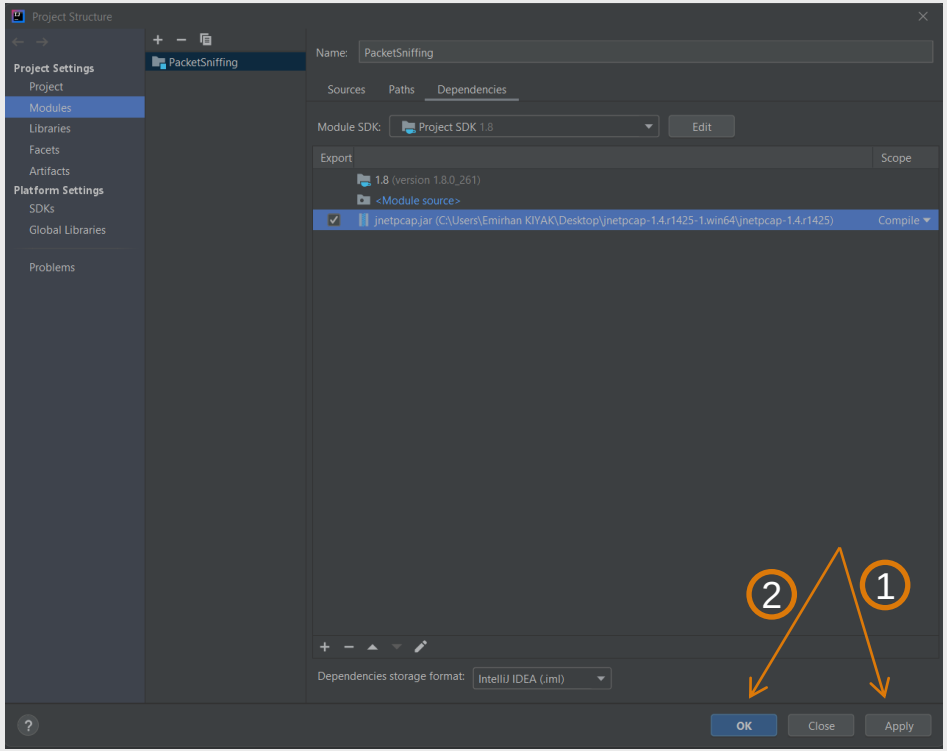
jNetPcap' in Java'ya Entegrasyonu

- Daha sonra Modules->Dependencies kısmına girelim ve jNetPcap klasörü içinde bulunan jnetpcap.jar dosyasını buraya ekleyelim



jNetPcap' in Java'ya Entegrasyonu

- Ekledikten sonra kütüphaneyi işaretleyerek ardından apply diyerek pencereyi kapatalım. Böylece Kütüphane kurulumu tamamlanmış oldu.



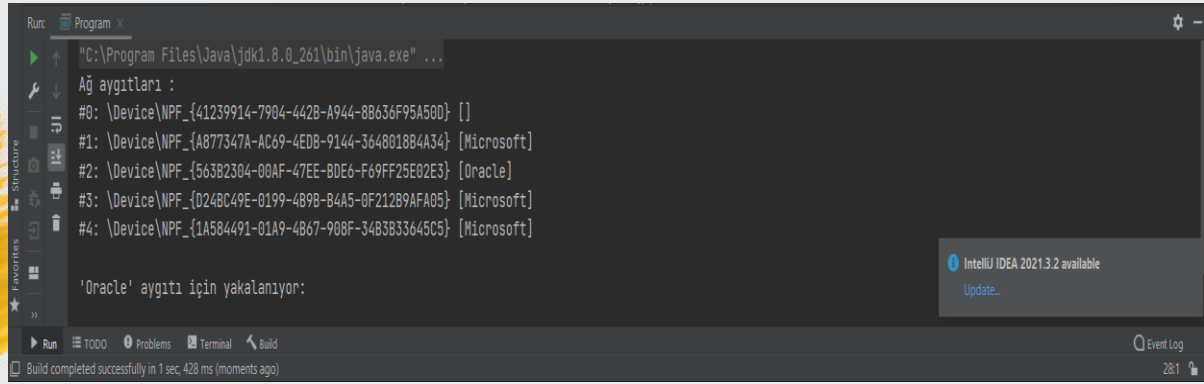
NOT : Eğer kütüphane işlev görmüyorsa, kütüphane klasörünü Windows gelişmiş sistem ayarları kısmından PATH olarak eklemeniz gerekebilir.

jNetPcap Kullanımı

- Öncelikle Ağ aygıtlarını listeleyerek başlayalım :

```
List<PcapIf> alldevs = new ArrayList<PcapIf>(); // aygıt listesi
StringBuilder errbuf = new StringBuilder(); // herhangi bir hata mesajı için
int r = Pcap.findAllDevs(alldevs,errbuf);
if(r == -1 || alldevs.isEmpty()){
    System.err.printf("Aygıt listesi okunamadı, hata -> %s",errbuf.toString());
    return;
}
System.out.println("Ağ aygıtları : ");

int i=0;
for (PcapIf device:alldevs){
    String description = (device.getDescription() != null) ? device.getDescription() : "Açıklama yok";
    System.out.printf("#%d: %s [%s]\n",i++, device.getName(),description);
}

PcapIf device = alldevs.get(4); // parametre olarak girilen aygıtı seç ve taramaya başla
System.out.printf("\n'%s' aygıtı için yakalanıyor:\n", (device.getDescription() != null) ? device.getDescription()
: device.getName());
int snaplen = 64*1024; // kesme olmadan tüm paketleri yakala
int flags = Pcap.MODE_PROMISCUOUS; // tüm paketleri yakala
int timeout = 2*1000;
Pcap pcap = Pcap.openLive(device.getName(),snaplen,flags,timeout,errbuf);

if(pcap == null){
    System.err.printf("Paket yakalama için ağ aygıtı açılırken hata oluştu : " + errbuf.toString());
    return;
}
```

Ağ Aygıtları Listesi

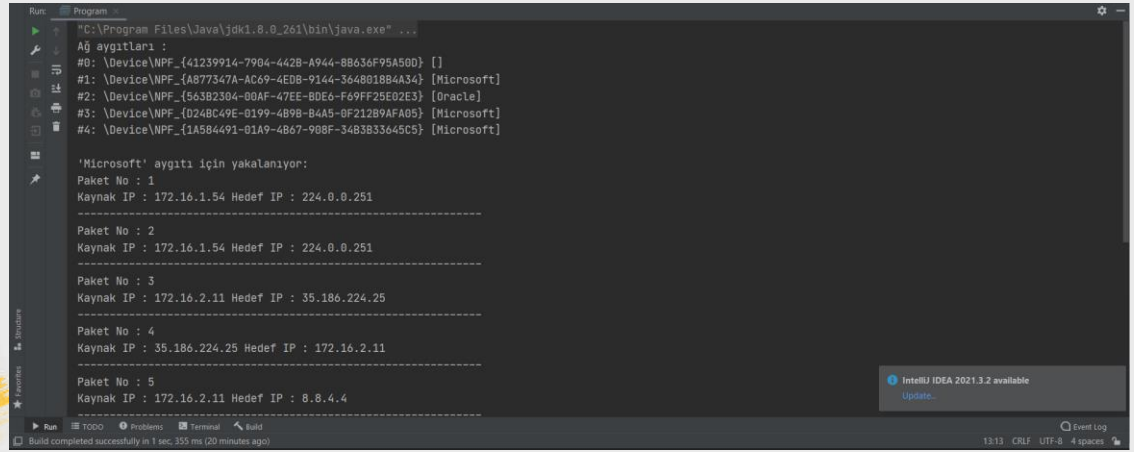


Örnek Uygulama

Kaynak ve Hedef Ip Adreslerini Bulma

```
PcapPacketHandler<String> jpacketHAndler = new PcapPacketHandler<String>() {
    int paketSayac = 1;
    @Override
    public void nextPacket(PcapPacket pcapPacket, String user) {
        byte[] data = pcapPacket.getByteArray(0,
         pcapPacket.size()); // paket verisi
        byte[] sIP = new byte[4];
        byte[] dIP = new byte[4];

        Ip4 ip = new Ip4();
        if(pcapPacket.hasHeader(ip) == false) return; // ip paketi yok

        sIP = ip.source();
        dIP = ip.destination();

        // JnetPcap format yardımcı araçlarını kullanma

        String sourceIP = org.jnetpcap.packet.format.FormatUtils.ip(sIP);
        String destinationIP = org.jnetpcap.packet.format.FormatUtils.ip(dIP);
        System.out.printf("Paket No : %d\n",paketSayac++);
        System.out.println("Kaynak IP : " + sourceIP +
                " Hedef IP : " + destinationIP +
        System.out.println("---------------------------------------------------------------------");
    }
};
pcap.loop(-1,jpacketHAndler, "jNetPcap"); // ne kadar paket yakalanacağını belirleriz. İlk parametre paket sayısını ifade eder. ‘ -1 ‘ verdiğimizde sınırsız olarak paket yakalama işlemine devam eder.
pcap.close();
```

Uygulama Ekran Çıktısı



Not : Diğer protokoller de aynı şekilde, Ip nesnesinin oluşturulup kullanılma şekliyle programda kullanılabilir.

KAYNAKÇA

Kullanılan ikonlar

- https://uxwing.com/http-icon/
- https://www.pngegg.com/en/search?q=DNS
- https://www.pngwing.com/en/free-png-saxdr/download
- https://commons.wikimedia.org/wiki/File:Osi-model.png