2016-09-26 机房1017

学习dpkt文档

<http://dpkt.readthedocs.io/en/latest/>

1. 找不到ICMP？
2. 解析Mac地址

# ARP

## （地址解析协议）

地址解析协议，即ARP（Address Resolution Protocol），是根据[IP地址](http://baike.baidu.com/view/3930.htm)获取[物理地址](http://baike.baidu.com/view/883168.htm)的一个[TCP/IP协议](http://baike.baidu.com/view/7649.htm)。[主机](http://baike.baidu.com/view/23880.htm)发送信息时将包含目标IP地址的ARP请求广播到网络上的所有主机，并接收返回消息，以此确定目标的物理地址；收到返回消息后将该IP地址和物理地址存入本机ARP缓存中并保留一定时间，下次请求时直接查询ARP缓存以节约资源。地址解析协议是建立在网络中各个主机互相信任的基础上的，网络上的主机可以自主发送ARP应答消息，其他主机收到应答报文时不会检测该报文的真实性就会将其记入本机ARP缓存；由此攻击者就可以向某一主机发送伪ARP应答报文，使其发送的信息无法到达预期的主机或到达错误的主机，这就构成了一个[ARP欺骗](http://baike.baidu.com/view/155386.htm)。[ARP命令](http://baike.baidu.com/view/935158.htm)可用于查询本机ARP缓存中IP地址和[MAC地址](http://baike.baidu.com/view/69334.htm)的对应关系、添加或删除静态对应关系等。相关协议有[RARP](http://baike.baidu.com/view/32772.htm)、[代理ARP](http://baike.baidu.com/view/547486.htm)。[NDP](http://baike.baidu.com/view/1420887.htm)用于在[IPv6](http://baike.baidu.com/view/5228.htm)中代替地址解析协议。

Returns the Ethernet type.  For example, type 2048 (0x0800) is IPv4 and 34525 (0x86DD) is IPv6

Import struct

# 利用 struct 处理字节流中的mac地址

def mac2str(bi\_mac):

mac = ""

for i in bi\_mac[:-1]:

i\_str = i + "\x00\x00\x00"

tmac, = struct.unpack("i", i\_str)

mac += str(tmac) + ":"

i\_str = bi\_mac[-1] + "\x00\x00\x00"

tmac, = struct.unpack("i", i\_str)

mac += str(tmac)

return mac

### 软件环境

1. winpcpy

2. pip install dpkt

pip install pypcap

3. libpcap

http://www.tcpdump.org/release/

### To do list:

1. 解析ip地址 OK

2. 包分类存储

3. 包分类展示

4. 关键字捕捉

https://docs.python.org/2/library/socket.html

有几个地址转换的函数在使用时出现错误，还是靠stackoverflow解决了问题

pip install win-inet-pton

http://stackoverflow.com/questions/20482381/attributeerror-module-object-has-no-attribute-inet-pton

<http://grokbase.com/t/python/python-list/071wdqq146/socket-inet-ntop-and-pton-question>

上次阅读了How-to socket，学到了一些东西，遗憾的是当初的文档没有保存…..