

# 以太网帧分析实验

同济大学软件学院

# 以太网

以太网是一种计算机局域网技术。IEEE组织的IEEE 802.3标准制定了以太网的技术标准，它规定了包括物理层的连线、电子信号和介质访问层协议的内容。以太网是目前应用最普遍的局域网技术。

以太网是现实世界中最普遍的一种计算机网络。以太网有两类：第一类是经典以太网，第二类是交换式以太网，使用了一种称为交换机的设备连接不同的计算机。

# 以太网

经典以太网是以太网的原始形式，运行速度从3~10 Mbps不等；而交换式以太网正是广泛应用的以太网，可运行在100、1000和10000Mbps那样的高速率，分别以快速以太网、千兆以太网和万兆以太网的形式呈现。

以太网的标准拓扑结构为总线型拓扑，但目前的快速以太网（100BASE-T、1000BASE-T标准）为了减少冲突，将能提高的网络速度和使用效率最大化，使用交换机来进行网络连接和组织。

# 以太网

如此一来，以太网的拓扑结构就成了星型；但在逻辑上，以太网仍然使用总线型拓扑和CSMA/CD（Carrier Sense Multiple Access/Collision Detection，即载波多重访问/碰撞侦测）的总线技术。

每一个节点有全球唯一的48位地址也就是制造商分配给网卡的MAC地址，以保证以太网上所有节点能互相鉴别。由于以太网十分普遍，许多制造商把以太网卡直接集成进计算机主板。



# MAC地址

MAC地址也叫物理地址、硬件地址，由网络设备制造商生产时烧录在网卡(Network Interface Card)的EPROM(一种闪存芯片，通常可以通过程序擦写)。

MAC地址的长度为48位(6个字节)，通常表示为12个16进制数，如：00-16-EA-AE-3C-40就是一个MAC地址，其中前3个字节，16进制数00-16-EA代表网络硬件制造商的编号，它由IEEE(电气与电子工程师协会)分配，而后3个字节，16进制数AE-3C-40代表该制造商所制造的某个网络产品(如网卡)的系列号。MAC地址在世界是唯一的。

# MAC地址

MAC地址由网络其前3字节表示OUI (Organizationally Unique Identifier)，是IEEE的注册管理机构给不同厂家分配的代码，区分不同的厂家。后3字节由厂家自行分配

MAC地址最高字节 (MSB) 的低第二位 (LSb) 表示这个MAC地址是全局的还是本地的，即U/L (Universal/Local) 位，如果为0，表示是全局地址。所有的OUI这一位都是0。

MAC地址最高字节 (MSB) 的低第一位 (LSb)，表示这个MAC地址是单播还是多播。0表示单播。

# MAC 数据包格式

前导码和帧开始符

一个帧以7个字节的前导码和1个字节的帧开始符作为帧的开始。其相应的16进制表示为0x55 0x55 0x55 0x55 0x55 0x55 0x55 0xD5。

报头

报头包含源地址和目标地址的MAC地址，以太类型字段和可选的用于说明VLAN成员关系和传输优先级的IEEE 802.1Q VLAN 标签。

帧校验码

帧校验码是一个32位循环冗余校验码，以便验证帧数据是否被损坏。



# MAC 数据包格式

## 帧间距

当一个帧发送出去之后，发送方在下次发送帧之前，需要再发送至少12个octet的空闲线路状态码。

## 以太帧类型

以太帧有很多种类型。不同类型的帧具有不同的格式和MTU值。但在同种物理媒体上都可同时存在。以太网第二版称之为Ethernet II帧，DIX帧，是最常见的帧类型。并通常直接被IP协议使用。



# MAC 数据包格式

## Ethernet II

以太 II 帧 (也称作DIX以太网, 是以这个设计的主要成员, DEC, Intel和Xerox的名字命名的。把紧接在目标和源MAC地址后面的这个两字节定义为以太网帧数据类型字段。

例如, 一个0x0800的以太类型说明这个帧包含的是IPv4数据报。同样的, 一个0x0806的以太类型说明这个帧是一个ARP帧, 0x8100说明这是一个IEEE 802.1Q帧, 而0x86DD说明这是一个IPv6帧。

# MAC 数据包格式

## Ethernet II

当这个工业界的标准通过正式的IEEE标准化过程后，在802.3标准中以太类型字段变成了一个(数据)长度字段。(最初的以太包通过包括他们的帧来确定它们的长度，而不是以一个明确的数值。)但是包的接收层仍需知道如何解析包，因此标准要求将IEEE802.2头跟在长度字段后面，定义包的类型。多年之后，802.3x-1997标准，一个802.3标准的后继版本，正式允许两种类型的数据包同时存在。

# MAC 数据包格式

## Ethernet II

实际上，两种数据包都被广泛使用，而最初的以太数据包在以太局域网中被广泛应用，因为他的简便和低开销。

为了允许一些使用以太II版本的数据报和一些使用802.3封装的最初版本的数据包能够在同一个以太网段使用，以太类型值必须大于等于1536(0x0600)。这个值比802.3数据包的最大长度1500byte (0x05DC)要更大。



# MAC 数据包格式

## Ethernet II

因此如果这个字段的值大于等于1536，则这个帧是以太II帧，而那个字段是类型字段。否则(小于1500而大于46字节)，他是一个IEEE 802.3帧，而那个字段是长度字段。1500~1536(不包含)的数值未定义。

802.3 以太网帧结构

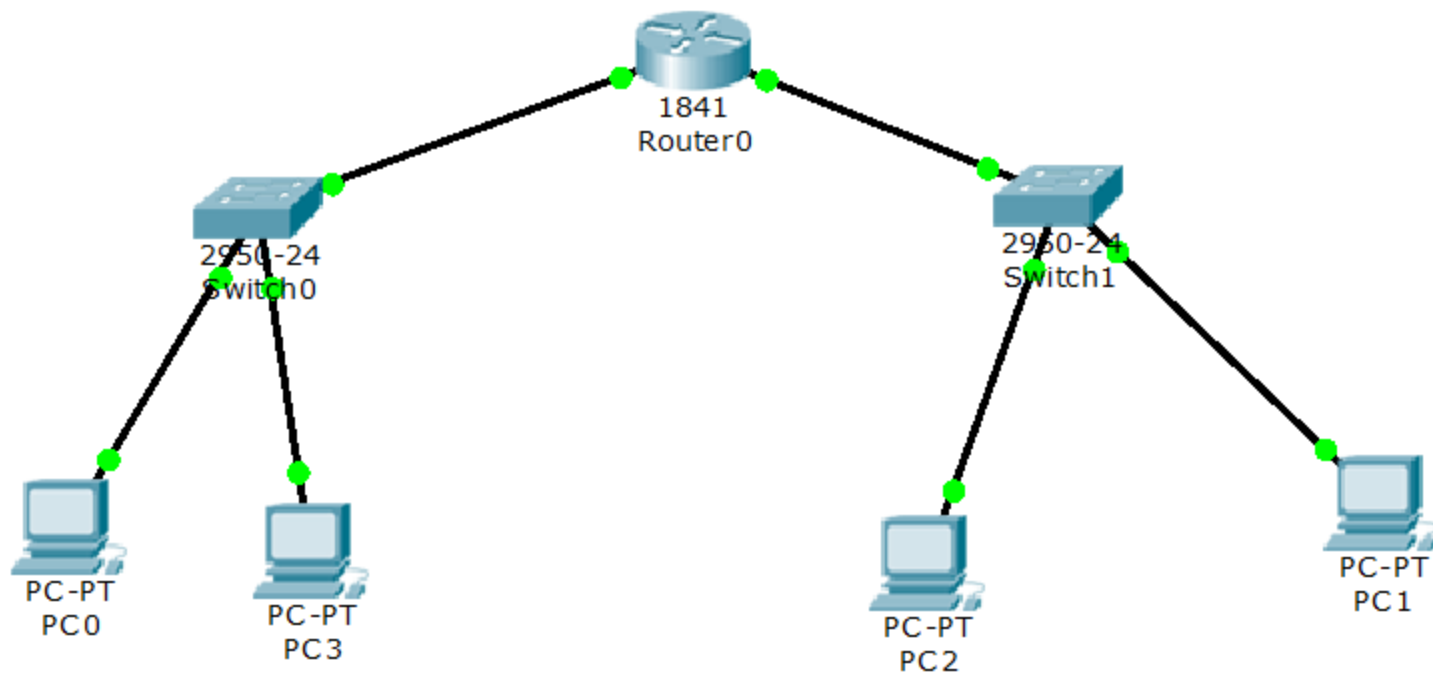
前导码	帧开始符	MAC 目标地址	MAC 源地址	802.1Q 标签 (可选)	以太类型	负载	冗余校验	帧间距
10101010 7个octet	10101011 1个octet	6 octets	6 octets	(4 octets)	2 octets	46-1500 octets	4 octets	12 octets
		64-1522 octets						
		72-1530 octets						
		84-1542 octets						



# 配置有关仿真网络

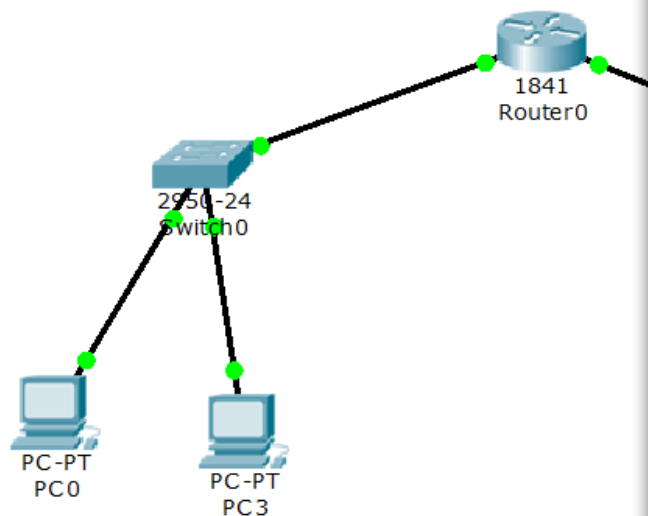
- 1 首先规划网络地址及拓扑图；
- 2 路由器接口IP地址配置；
- 3 配置DHCP之前检查PC是否存在IP地址；
- 4 在R0，配置 DHCP；
- 5 验证各个PC的IP地址。

# 实验示例图



# 查看数据包

点击模拟ICMP包，查看相关数据



PDU Information at Device: Switch0

At Device: Switch0  
Source: PC0  
Destination: PC2

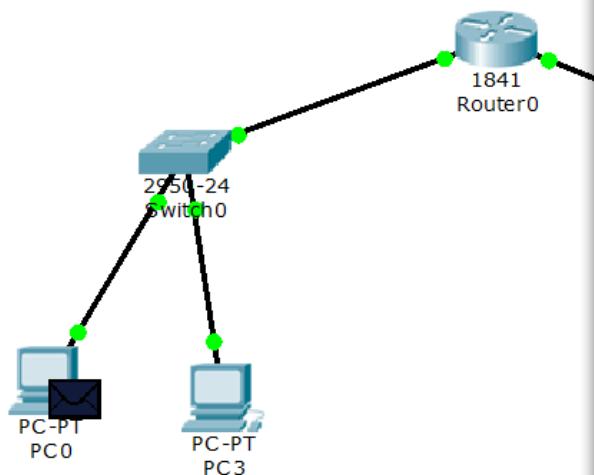
OSI Model	Inbound PDU Details	Outbound PDU Details
Layer 7:		
Layer 6:		
Layer 5:		
Layer 4:		
Layer 3:		
Layer 2: Ethernet II Header	0050.0FAB.0418 >> 0005.5E96.9C01	0050.0FAB.0418 >> 0005.5E96.9C01
Layer 1: Port FastEthernet0/2		Layer 1: Port(s): FastEthernet0/1

1. FastEthernet0/2 receives the frame.

Challenge Me << Previous Layer Next Layer >>

# 查看数据包

点击模拟ICMP包，查看相关数据



PDU Information at Device: PC0

OSI Model    Outbound PDU Details

PDU Formats

Ethernet II

0	4	8	14	19	Bytes
PREAMBLE: 101010...1011		DEST MAC: 0005.5E96.9C01		SRC MAC: 0050.0FAB.0418	
TYPE: 0x800		DATA (VARIABLE LENGTH)		FCS: 0x0	

IP

0	4	8	16	19	31	Bits
4		IHL	DSCP: 0x0		TL: 28	
ID: 0x12			0x0		0x0	
TTL: 255		PRO: 0x1		CHKSUM		
SRC IP: 192.168.1.11						
DST IP: 192.168.2.11						
OPT: 0x0					0x0	
DATA (VARIABLE LENGTH)						

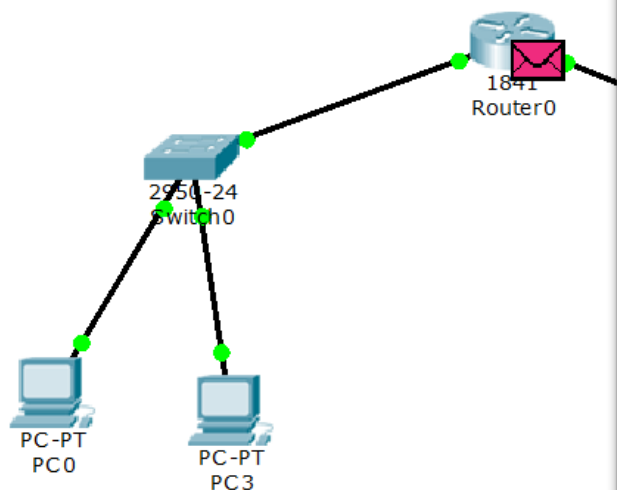
ICMP

0	8	16	31	Bits	
TYPE: 0x8		CODE: 0x0		CHECKSUM	
ID: 0x10		SEQ NUMBER: 15			



# 查看数据包

点击模拟ICMP包，查看相关数据



PDU Information at Device: Switch0

OSI Model | Inbound PDU Details | Outbound PDU Details

PDU Formats

Ethernet II

0		4		8		14		19		Bytes	
PREAMBLE: 101010...1011				DEST MAC: 0005.5E96.9C01				SRC MAC: 0050.0FAB.0418			
TYPE: 0x800		DATA (VARIABLE LENGTH)						FCS: 0x0			

IP

0		4		8		16		19		31		Bits	
4		IHL		DSCP: 0x0		TL: 28							
ID: 0x14				0x0		0x0							
TTL: 255				PRO: 0x1		CHKSUM							
SRC IP: 192.168.1.11													
DST IP: 192.168.2.14													
OPT: 0x0										0x0			
DATA (VARIABLE LENGTH)													

ICMP

0		8		16		31		Bits			
TYPE: 0x8		CODE: 0x0		CHECKSUM							
ID: 0x12				SEQ NUMBER: 17							

# Wireshark 安装使用

查找有关安装说明，下载相关软件；

安装软件；

练习使用；

分析MAC DIX V2 帧；

# Wireshark 抓包界面 (DIX V2)

Capturing from Broadcom NetXtreme Gigabit Ethernet Driver - Wireshark

File Edit View Go Capture Analyze Statistics Telephony Tools Help

Filter:  Expression... Clear Apply

No.	Time	Source	Destination	Protocol	Info
388577	2734.57708	133.24.248.18	192.168.2.100	TCP	[TCP Fast Retransmission] [TCP segment of a reassembled PDU]
388578	2734.57714	192.168.2.100	133.24.248.18	TCP	58851 > http [ACK] Seq=608 Ack=21008625 win=262080 Len=0 SLE=21010081 SRE=21012993
388579	2734.58409	159.89.89.188	192.168.2.100	TLSv1.2	Continuation Data
388580	2734.58417	192.168.2.100	159.89.89.188	TCP	[TCP Dup ACK 388534#1] 58827 > https [ACK] Seq=1346 Ack=44068497 win=262080 Len=0 SLE=44071409 SRE=44075777
388581	2734.58419	159.89.89.188	192.168.2.100	TLSv1.2	Continuation Data
388582	2734.58423	192.168.2.100	159.89.89.188	TCP	[TCP Dup ACK 388534#2] 58827 > https [ACK] Seq=1346 Ack=44068497 win=262080 Len=0 SLE=44071409 SRE=44077233
388583	2734.71429	47.96.253.105	192.168.2.100	HTTP	Continuation or non-HTTP traffic
388584	2734.73749	192.168.2.100	40.73.105.168	TCP	57696 > http [ACK] Seq=1968 Ack=119200 win=7930 Len=0
388585	2734.76756	192.168.2.100	120.241.25.38	TCP	59082 > 36688 [ACK] Seq=10274 Ack=441 win=65096 Len=0
388586	2734.78430	121.196.50.150	192.168.2.100	TLSv1.2	Application Data
388587	2734.78451	192.168.2.100	121.196.50.150	TLSv1.2	Application Data
388588	2734.79224	121.196.50.150	192.168.2.100	TCP	https > 54493 [ACK] Seq=7453 Ack=4132 win=37453 Len=0
388589	2734.82351	120.241.25.38	192.168.2.100	TCP	[TCP Retransmission] 36688 > 59082 [PSH, ACK] Seq=421 Ack=10274 win=39936 Len=20
388590	2734.82359	192.168.2.100	120.241.25.38	TCP	[TCP Dup ACK 388585#1] 59082 > 36688 [ACK] Seq=10274 Ack=441 win=65096 Len=0 SLE=421 SRE=441
388591	2734.88237	61.129.47.29	192.168.2.100	TCP	[TCP Retransmission] [TCP segment of a reassembled PDU]
388592	2734.88251	192.168.2.100	61.129.47.29	TCP	58682 > http [ACK] Seq=505 Ack=115322241 win=66976 Len=0 SLE=115319329 SRE=115320241
388593	2734.89548	61.129.47.29	192.168.2.100	TCP	[TCP segment of a reassembled PDU]
388594	2734.89660	61.129.47.29	192.168.2.100	TCP	[TCP segment of a reassembled PDU]
388595	2734.89665	192.168.2.100	61.129.47.29	TCP	58682 > http [ACK] Seq=505 Ack=115325153 win=66976 Len=0
388596	2734.91756	192.168.2.100	47.96.253.105	TCP	57695 > http [ACK] Seq=13643 Ack=2942042 win=8103 Len=0
388597	2734.92000	61.129.47.29	192.168.2.100	TCP	[TCP segment of a reassembled PDU]
388598	2734.92111	61.129.47.29	192.168.2.100	TCP	[TCP segment of a reassembled PDU]
388599	2734.92119	192.168.2.100	61.129.47.29	TCP	58682 > http [ACK] Seq=505 Ack=115328065 win=66976 Len=0
388600	2734.92561	47.96.253.105	192.168.2.100	HTTP	Continuation or non-HTTP traffic
388601	2734.93010	133.24.248.18	192.168.2.100	TCP	[TCP Retransmission] [TCP segment of a reassembled PDU]
388602	2734.93019	192.168.2.100	133.24.248.18	TCP	58851 > http [ACK] Seq=608 Ack=21012993 win=262080 Len=0
388603	2734.93021	133.24.248.18	192.168.2.100	TCP	[TCP segment of a reassembled PDU]
388604	2734.93034	133.24.248.18	192.168.2.100	TCP	[TCP segment of a reassembled PDU]
388605	2734.93038	192.168.2.100	133.24.248.18	TCP	58851 > http [ACK] Seq=608 Ack=21015905 win=262080 Len=0
388606	2734.93508	61.129.47.29	192.168.2.100	TCP	[TCP segment of a reassembled PDU]
388607	2734.93648	61.129.47.29	192.168.2.100	TCP	[TCP segment of a reassembled PDU]
388608	2734.93656	192.168.2.100	61.129.47.29	TCP	58682 > http [ACK] Seq=505 Ack=115330977 win=66976 Len=0

Frame 21298: 54 bytes on wire (432 bits), 54 bytes captured (432 bits)

Ethernet II, Src: 10:dd:b1:c1:e6:dd (10:dd:b1:c1:e6:dd), Dst: f4:ee:14:26:84:a2 (f4:ee:14:26:84:a2)

- Destination: f4:ee:14:26:84:a2 (f4:ee:14:26:84:a2)
  - Address: f4:ee:14:26:84:a2 (f4:ee:14:26:84:a2)
    - ....0. .... = IG bit: Individual address (unicast)
    - ....0. .... = LG bit: Globally unique address (factory default)
- Source: 10:dd:b1:c1:e6:dd (10:dd:b1:c1:e6:dd)
  - Address: 10:dd:b1:c1:e6:dd (10:dd:b1:c1:e6:dd)
    - ....0. .... = IG bit: Individual address (unicast)
    - ....0. .... = LG bit: Globally unique address (factory default)

```

0000 f4 ee 14 26 84 a2 10 dd b1 c1 e6 dd 08 00 45 00 ...&... ..E.
0010 00 28 79 e3 40 00 80 06 78 69 c0 a8 02 64 65 59 .(y.@...xi...dey
0020 e0 1d e1 61 00 50 68 6c 97 56 7d 51 66 ed 50 10 ...a.PhI.V}qf.P.
0030 20 00 c1 9d 00 00
    
```

# 实验主要分析内容

1. 查看本机的MAC地址
2. 用Wireshark抓取MAC数据包。
3. 查看MAC数据包字段内容，并解读；
4. 分析在Packet tracer中模拟ICMP（ping 命令），ICMP数据包转发过程中MAC地址变化情况。