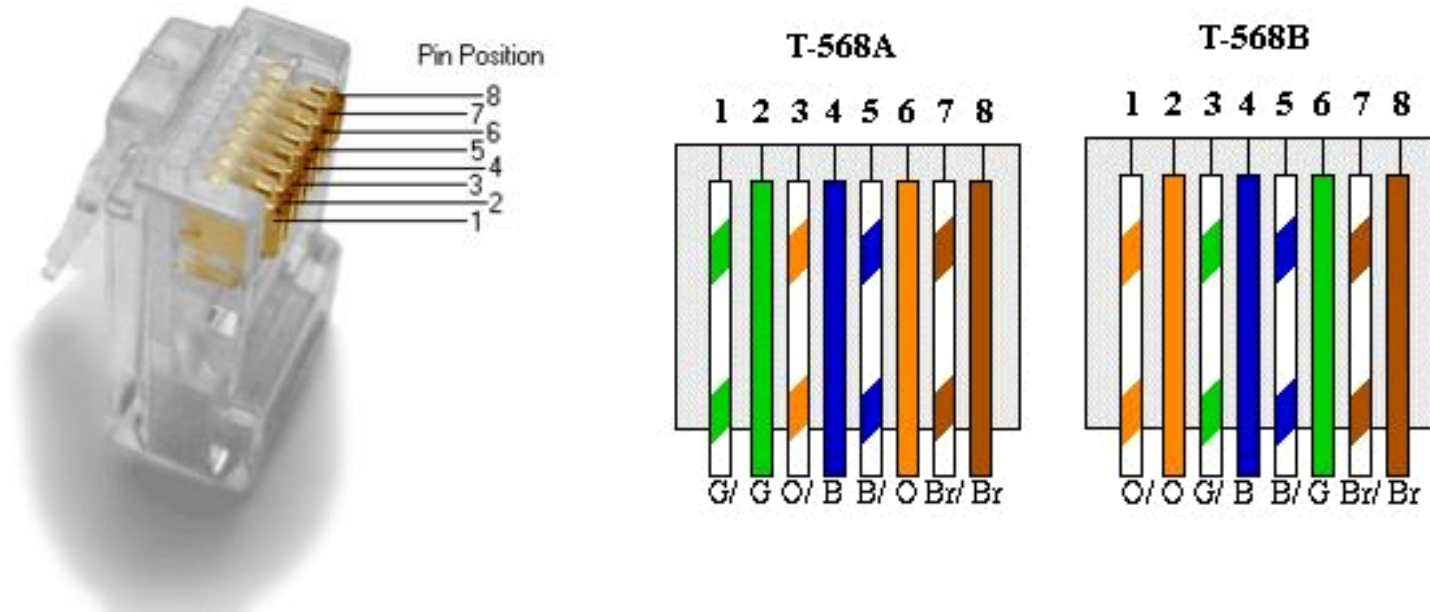


# Review on Computer Network Experiments, 6-1-2010

# Lab 1 區域網路線材製作

- 熟悉網路線製作過程
- 了解跳線與非跳線之區別與排列規則



# Lab 2 網路協定觀察與分析 (1/2)

- 了解網路封包之分層方式
- 能使用工具抓取網路封包
- 分析 **http** 之傳輸流程及封包內容

# Lab 2 網路協定觀察與分析 (2/2)

The image shows a Wireshark network traffic capture window. The main pane displays a list of captured packets with columns for No., Time, Source, Destination, Protocol, and Info. Packet 30 is highlighted in blue, showing an HTTP 200 OK response. The packet details pane below shows the structure of this packet, including Ethernet II, Internet Protocol, Transmission Control Protocol, and Hypertext Transfer Protocol layers. The raw data pane at the bottom shows the hexadecimal and ASCII representation of the packet data.

No.	Time	Source	Destination	Protocol	Info
22	2.369079	192.168.11.2	64.233.183.105	TCP	[TCP Dup ACK 1/#2] 58055 > http [ACK] Seq=94
23	2.369088	192.168.11.2	64.233.183.105	TCP	[TCP Dup ACK 1/#3] 58055 > http [ACK] Seq=94
24	2.439870	192.168.11.2	74.125.153.101	HTTP	POST /tbproxy/af/query HTTP/1.1 (text/plain)
25	2.482822	74.125.153.101	192.168.11.2	TCP	http > 57532 [ACK] Seq=1 Ack=1165 Win=62920
26	2.501212	192.168.11.2	74.125.19.101	TCP	58057 > http [SYN] Seq=0 win=8192 Len=0 MSS=
27	2.681611	74.125.19.101	192.168.11.2	TCP	http > 58057 [SYN, ACK] Seq=0 Ack=1 Win=5720
28	2.681677	192.168.11.2	74.125.19.101	TCP	58057 > http [ACK] Seq=1 Ack=1 Win=64350 Len
29	2.681843	192.168.11.2	74.125.19.101	HTTP	GET /csi?v=3&s=dictionary&action=&srt=276&t
30	2.706198	74.125.153.101	192.168.11.2	HTTP/X	HTTP/1.1 200 OK
31	2.853005	74.125.19.101	192.168.11.2	TCP	http > 58057 [ACK] Seq=1 Ack=437 Win=6432 Le
32	2.911769	192.168.11.2	74.125.153.101	TCP	57532 > http [ACK] Seq=1165 Ack=270 Win=6300
33	2.919398	74.125.19.101	192.168.11.2	HTTP	HTTP/1.1 204 No Content
34	3.121765	192.168.11.2	74.125.19.101	TCP	58057 > http [ACK] Seq=437 Ack=216 Win=6413

Packet 30 details:

- Frame 30 (323 bytes on wire, 323 bytes captured)
- Ethernet II, Src: Buffalo\_91:40:38 (00:1d:73:91:40:38), Dst: AsustekC\_72:76:05 (00:1e:8c:72:76:05)
- Internet Protocol, Src: 74.125.153.101 (74.125.153.101), Dst: 192.168.11.2 (192.168.11.2)
- Transmission Control Protocol, Src Port: http (80), Dst Port: 57532 (57532), Seq: 1, Ack: 1165, Len: 269
- Hypertext Transfer Protocol
  - HTTP/1.1 200 OK\r\n
  - Content-Type: text/xml; charset=utf-8\r\n
  - Content-Encoding: gzip\r\n
  - Date: Thu, 27 May 2010 07:34:29 GMT\r\n
  - Server: AutoFill Server\r\n
  - Cache-Control: private, x-gzip-ok=""\r\n
  - Content-Length: 67\r\n

Raw data (hex and ASCII):

```
0000 00 1e 8c 72 76 05 00 1d 73 91 40 38 08 00 45 00  ...rv... s.@8..E.  
0010 01 35 51 84 00 00 34 06 84 b2 4a 7d 99 65 c0 a8  .5Q...4. ..J}.e..  
0020 0b 02 00 50 e0 bc 76 e7 e8 16 6a 2f ce 0b 50 18  ...P..v. ..j/.P..  
0030 f5 c8 b5 0c 00 00 48 54 54 50 2f 31 2e 31 20 32  ....HT TP/1.1 2..  
0040 30 30 20 4f 4b 0d 0a 43 6f 6e 74 65 6e 74 2d 54  00 OK..C ontent-T..  
0050 79 70 65 3a 20 74 65 78 74 2f 78 6d 6c 3b 20 63  type: tex t/xml; c..  
0060 68 61 72 73 65 74 3d 75 74 66 2d 38 0d 0a 43 6f  charset=u tf-8..Co
```

List of packets

Packet layering

Raw data

# Lab 3 Linux 網路協定程式追蹤

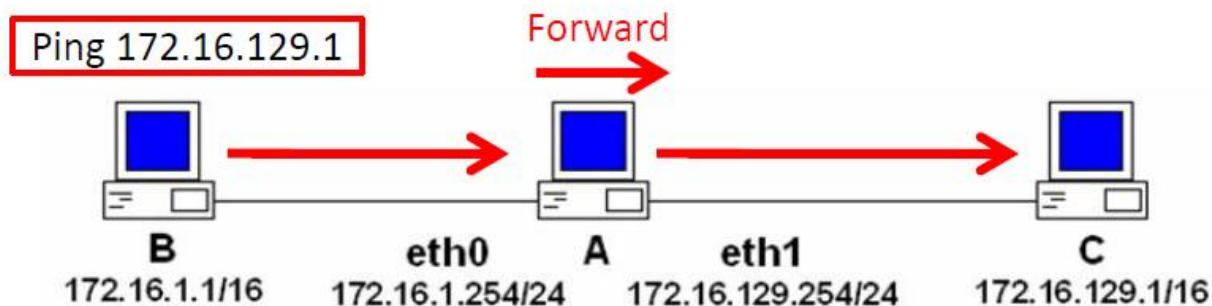
- 了解如何編譯核心
- 能搜尋，修改核心原始碼
- 追蹤開機過程中，網路模組的相關訊息

Add log  
message  
for tracing

```
int ip_fragment(struct sk_buff *skb, int (*output)(struct sk_buff *))
{
    printk(KERN_INFO "ip_output.c: ip_fragment()");
    struct iphdr *iph;
    int raw = 0;
    int ptr;
    struct net_device *dev;
    struct sk_buff *skb2;
    unsigned int mtu, hlen, left, len, ll_rs, pad;
    int offset;
    __be16 not_last_frag;
    struct rtable *rt = skb_rtable(skb);
    int err = 0;
```

# Lab 4 Linux子網域分割之設定與觀察 (1/2)

- 了解subnet運作原理與netmask的用途
- 設定ARP-Proxy



```
[root@localhost ~]# arp -i eth0 -Ds 172.16.129.1 eth0 pub
[root@localhost ~]# arp -i eth1 -Ds 172.16.1.1 eth1 pub
[root@localhost ~]# arp -n
```

Address	HWtype	HWaddress	Flags	Mask	Iface
172.16.1.1	*	*	MP		eth1
172.16.129.1	*	*	MP		eth0

ARP cache

# Lab 4 Linux子網域分割之設定與觀察 (2/2)

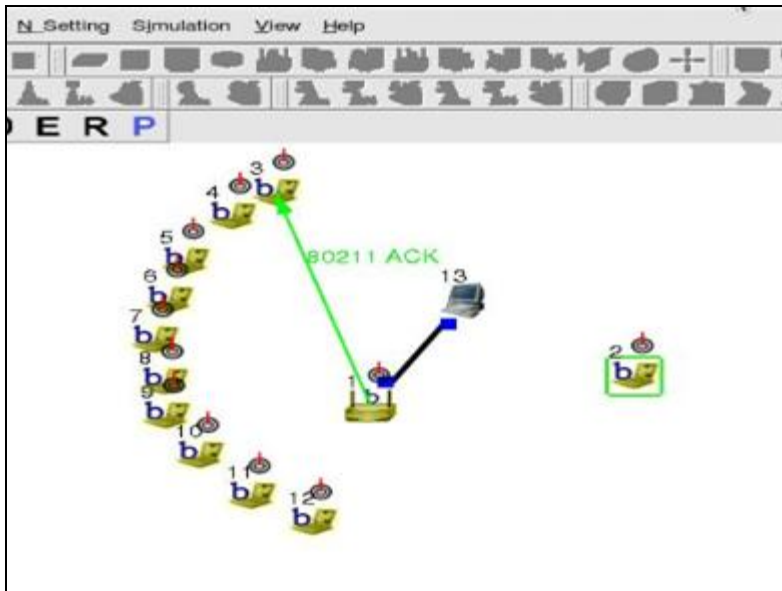
- 使用tcpdump觀察ICMP與ARP封包

```
[root@localhost ~]# tcpdump -n arp
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes
11:02:17.977096 ARP, Request who-has 172.16.1.254 tell 172.16.1.1, length 46
11:02:17.981470 ARP, Reply 172.16.1.254 is-at 08:00:27:12:10:b9, length 28
11:02:22.974058 ARP, Request who-has 172.16.1.1 tell 172.16.1.254, length 28
11:02:22.974891 ARP, Reply 172.16.1.1 is-at 08:00:27:a0:19:35, length 46
11:02:31.791721 ARP, Request who-has 172.16.129.254 tell 172.16.1.1, length 46
11:02:31.791801 ARP, Reply 172.16.129.254 is-at 08:00:27:12:10:b9, length 28
11:05:08.893929 ARP, Request who-has 172.16.129.1 tell 172.16.1.1, length 46
11:05:08.904546 ARP, Reply 172.16.129.1 is-at 08:00:27:12:10:b9, length 28
11:05:14.600907 ARP, Request who-has 172.16.1.1 tell 172.16.1.254, length 28
11:05:14.604034 ARP, Reply 172.16.1.1 is-at 08:00:27:a0:19:35, length 46
```

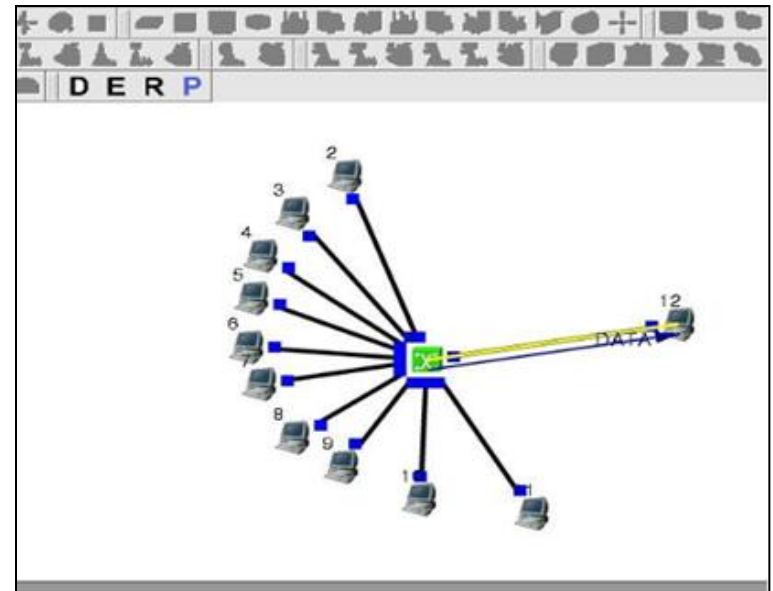
Snapshot of tcpdump

# Lab 5 媒介存取協定模擬 (1/2)

- 用NCTUns建立含wireless AP 和 router/switch 的網路拓撲



Wireless scenario (AP)

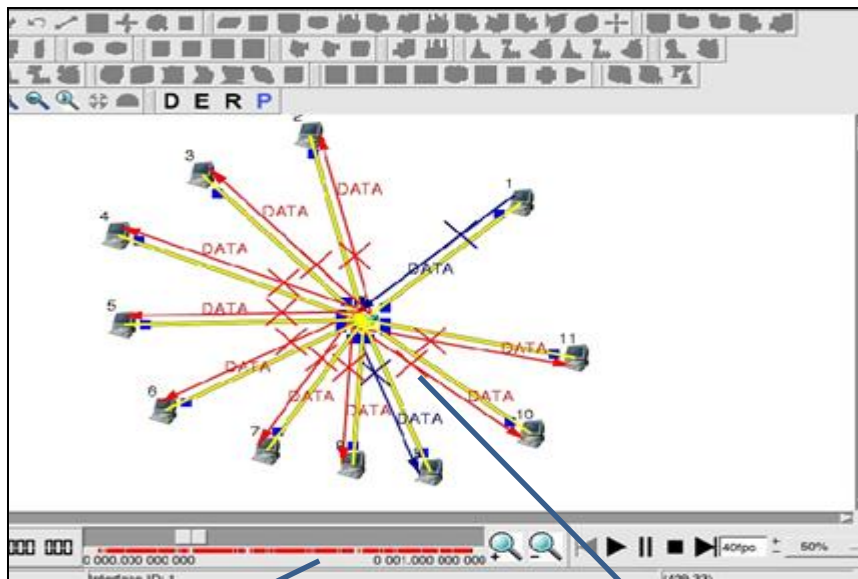


Wired scenario (Switch)



# Lab 5 媒介存取協定模擬 (2/2)

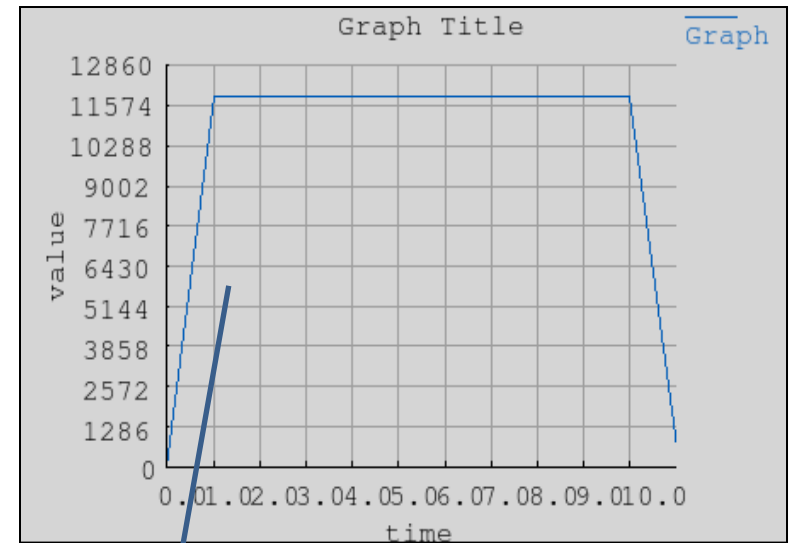
- 設定與執行網路流量模擬



Control bar

Collisions

Throughput

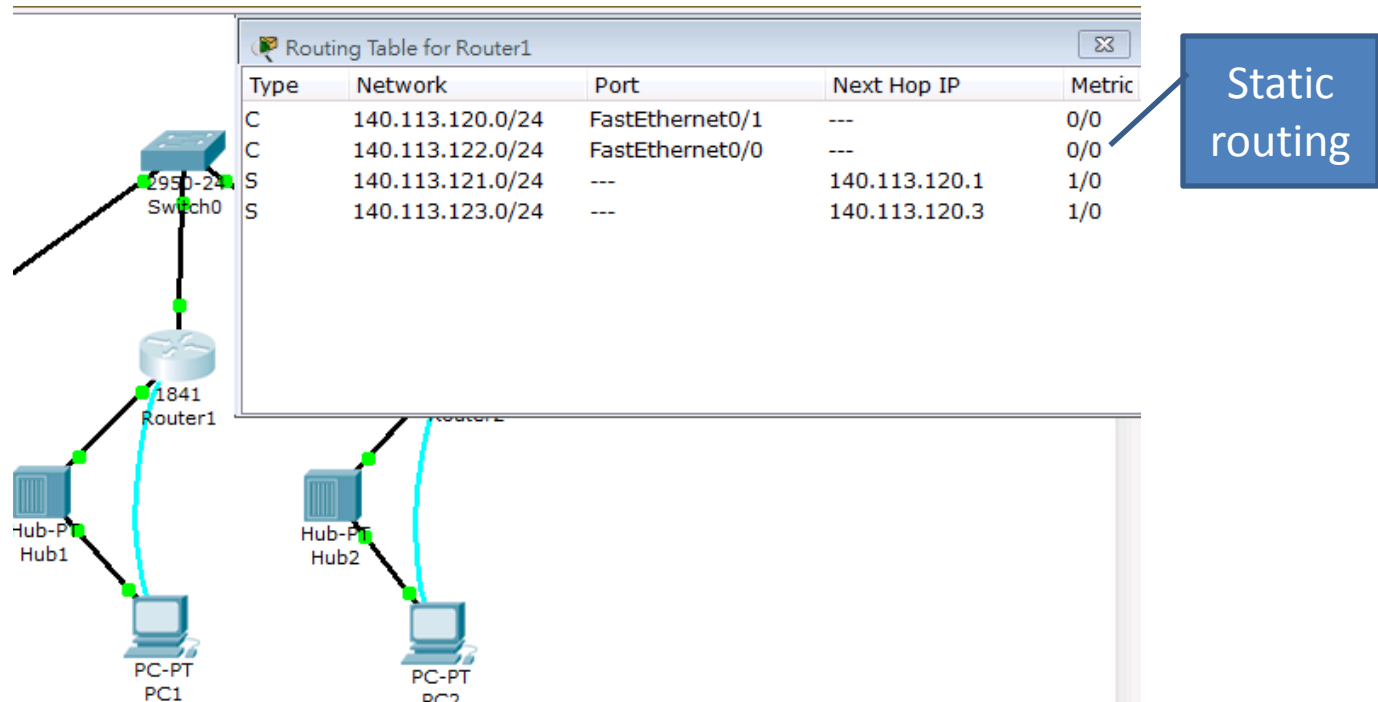


Simulation result

- 改變node之間連線的capacity

# Lab 6 路由器操作設定 (1/2)

- 建立一含 router 的基本網路拓撲
- 設定 static routing 使不同 subnet 的電腦能互相連通



# Lab 6 路由器操作設定 (2/2)

```
Router>enable
Router#configure
Configuring from terminal, memory, or network [terminal]?
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface fastEthernet 0/0
Router(config-if)#ip address 140.113.121.1 255.255.255.0
Router(config-if)#exit
Router(config)#ip route 140.113.123.0 255.255.255.0 140.113.120.3
```

- Set IP address and add static routing

# Lab 7 Linux 路由器之建構與追蹤 (1/2)

- 在 kernel 中加上 kdb 的 patch
- 使用 kdb 追蹤 kernel function
- 設定 quagga 之 static routing 及 RIP

```
Entering kdb (current=0xc0769e80, pid 0) on processor 0 due to Keyboard Entry
[0]kdb> bp ip_rt_ioctl

Instruction(i) BP #0 at 0xc0509960 (ip_rt_ioctl)
    is enabled globally adjust 1#addr at 00000000c0509960, hardtype=0, forcehw=0
, installed=0, hard=c08bb818

[0]kdb> go
```

Set breakpoint

# Lab 7 Linux 路由器之建構與追蹤 (2/2)

```
root@ubuntu:~# route del default
Instruction(i) breakpoint #0 at 0xc0509960 (adjusted)
0xc0509960 ip_rt_ioctl:      int3

Entering kdb (current=0xf01657f0, pid 836) on processor 0 due to Breakpoint @ 0xc0509960
[0]kdb> ssb

0xc0509961 ip_rt_ioctl+0x1:      mov     %esp,%ebp
0xc0509963 ip_rt_ioctl+0x3:      sub     $0xd0,%esp
0xc0509969 ip_rt_ioctl+0x9:      mov     %eax,0xffffffff40(%ebp)
0xc050996f ip_rt_ioctl+0xf:      mov     %gs:0x14,%eax
0xc0509975 ip_rt_ioctl+0x15:     mov     %eax,0xfffffffff0(%ebp)
0xc0509978 ip_rt_ioctl+0x18:     xor     %eax,%eax
0xc050997a ip_rt_ioctl+0x1a:     lea    0xffff76f5(%edx),%eax
0xc0509980 ip_rt_ioctl+0x20:     cmp     $0x1,%eax
0xc0509983 ip_rt_ioctl+0x23:     mov     %ebx,0xfffffffff4(%ebp)
0xc0509986 ip_rt_ioctl+0x26:     mov     %edx,%ebx
0xc0509988 ip_rt_ioctl+0x28:     mov     %esi,0xfffffffff8(%ebp)
0xc050998b ip_rt_ioctl+0x2b:     mov     %ecx,%esi
0xc050998d ip_rt_ioctl+0x2d:     mov     %edi,0xfffffffffc(%ebp)
0xc0509990 ip_rt_ioctl+0x30:     mov     $0xffffffe0,%edi
0xc0509995 ip_rt_ioctl+0x35:     ja     0xc05099b0 ip_rt_ioctl+0x50
[0]kdb> _
```

- Trigger break point function
- trace its execution flow

# Lab 8 網路探測：路徑、延遲與流量統計 (1/2)

- 找出封包在網路上傳遞的路徑
  - 路徑探測工具的原理



Map

Routing path with RTT

Information of a node

VisualRoute



# Lab 9 建置網路安全閘道器 (1/2)

- iptables 操作
- 使用 squid 過濾 http 連線
- 以 openvpn 建立 site-to-site vpn

iptables rule for dropping packet

```
root@ubuntu:~# iptables -A FORWARD -d 140.113.88.160 -o eth0 -j DROP
root@ubuntu:~# iptables -L -v
Chain INPUT (policy ACCEPT 158 packets, 49432 bytes)
 pkts bytes target    prot opt in      out     source         destination
Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target    prot opt in      out     source         destination
    0    0 DROP     all  --  any     eth0     anywhere       140.113.88.160
Chain OUTPUT (policy ACCEPT 10 packets, 728 bytes)
 pkts bytes target    prot opt in      out     source         destination
```

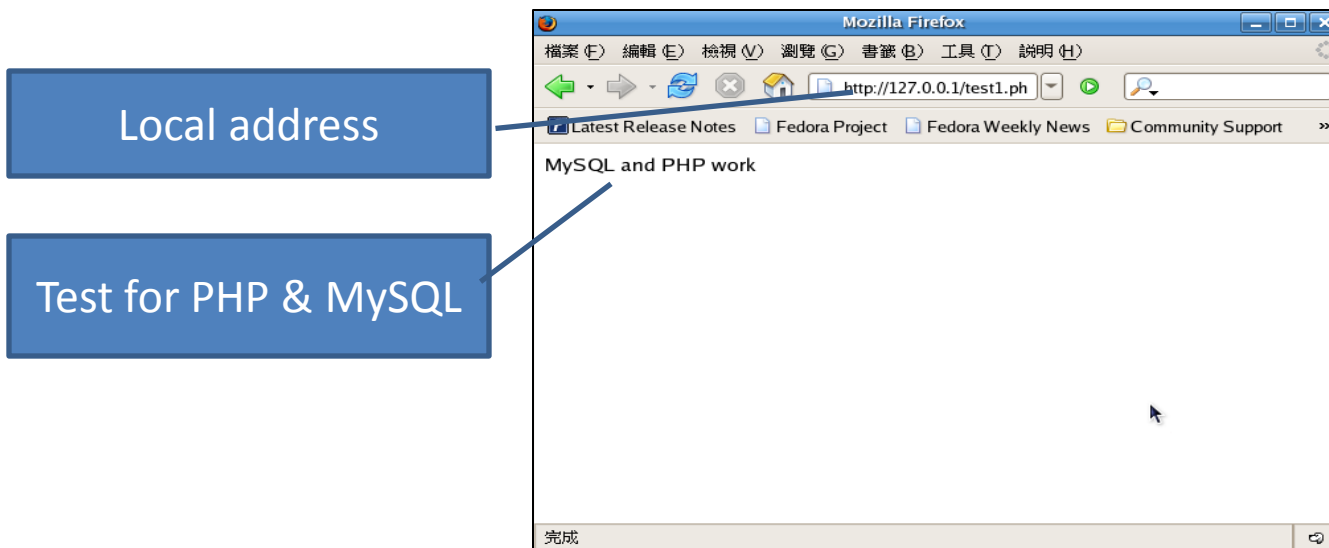


# Lab 9 建置網路安全閘道器 (2/2)

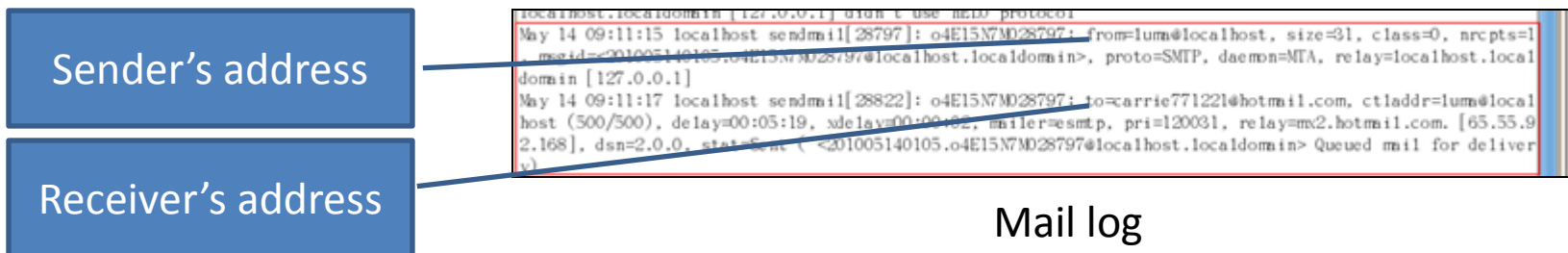
- Squid ACL rules
  - `acl EXEFile urlpath_regex/*.exe`
  - `http_access deny EXEFile`
  
  - `acl Badsite url_regex dstdomain www.sex.com`
  - `http_access deny Badsite`

# Lab 10 以Linux架設Internet/Intranet 伺服器 (1/2)

- 架設Web Server (Apache/PHP/MySQL)



- 架設Mail Server (Qmail)



Mail log

# Lab 10 以Linux架設Internet/Intranet 伺服器 (2/2)

- 架設 FTP server (vsftpd)
- 架設 Samba server



Files in samba server

# Lab 11 建置防範病毒信及廣告信之 郵件伺服器 (1/2)

- 安裝設定Postfix, Amavisd-new, SpamAssassin, 和 ClamAV
- 廣告信件之攔截

```
Return-Path: <vt2@nctuswimteam.twbbs.org>  
X-Original-To: vt@mailshead.twbbs.org  
Delivered-To: vt@mailshead.twbbs.org  
Received: from ShEaD.Dorm13.NCTU.edu.tw (localhost [127.0.0.1])  
by mailshead.twbbs.org (Postfix) with ESMTTP id 295CF7314E  
for <vt@mailshead.twbbs.org>: Mon, 24 May 2010 01:58:17 +0800 (CST)  
X-Quarantine-ID: <VoGf3aiQT0ms>  
X-Virus-Scanned: amavisd-new at twbbs.org  
X-Spam-Flag: YES  
X-Spam-Score: 1005.793  
X-Spam-Level: *****  
X-Spam-Status: Yes, score=1005.793 tagged_above=-999 required=6
```

Score of SpamAssassin

# Lab 11 建置防範病毒信及廣告信之 郵件伺服器 (2/2)

- 病毒信件之攔截

```
Content-Type: message/rfc822; x-spam-type=original; name="message"  
Content-Disposition: attachment; filename="message"  
Content-Transfer-Encoding: 7bit  
Content-Description: Original message  
  
Return-Path: <test@MailA.dorm13.nctu.edu.tw>  
Received: from MailA.dorm13.nctu.edu.tw (MailA.dorm13.nctu.edu.tw [192.168.3.1])  
    by MailB.dorm13.nctu.edu.tw (Postfix) with ESMTTP id 1CA2A1100EB  
    for <test@MailB.dorm13.nctu.edu.tw>; Mon, 24 May 2010 18:21:27 +0800 (CST)  
Received: from MailA.dorm13.nctu.edu.tw (localhost [127.0.0.1])  
    by MailA.dorm13.nctu.edu.tw (Postfix) with ESMTTP id 1A6DD1100F4  
    for <test@MailB.dorm13.nctu.edu.tw>; Mon, 24 May 2010 18:09:59 +0800 (CST)  
X-Virus-Scanned: amavisd-new at MailA.dorm13.nctu.edu.tw  
X-Amavis-Alert: INFECTED. message contains virus: Eicar-Test-Signature  
Received: from MailA.dorm13.nctu.edu.tw ([127.0.0.1])  
    by MailA.dorm13.nctu.edu.tw (MailA.dorm13.nctu.edu.tw [127.0.0.1]) (amavisd-new, port 10024)  
    with ESMTTP id nT5Vie8EgPtb for <test@MailB.dorm13.nctu.edu.tw>;  
    Mon, 24 May 2010 18:09:58 +0800 (CST)  
Received: from nctub2399a3737 (XP1.dorm13.nctu.edu.tw [192.168.1.1])  
    by MailA.dorm13.nctu.edu.tw (Postfix) with SMTP id 3CB6C10E45B  
    for <test@MailB.dorm13.nctu.edu.tw>; Mon, 24 May 2010 18:09:58 +0800 (CST)  
Message-ID: <7730DC315DF94C179E80D8E2B711CFBB@nctub2399a3737>
```

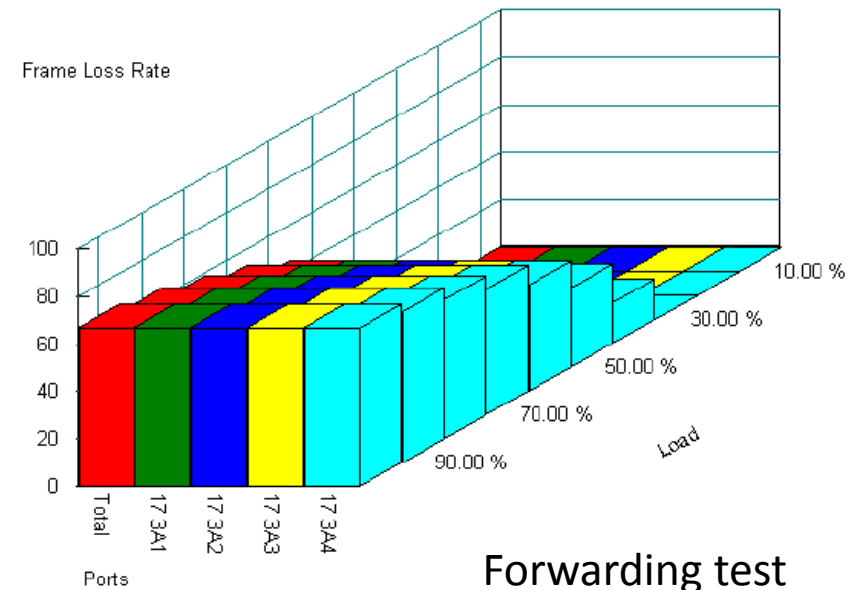
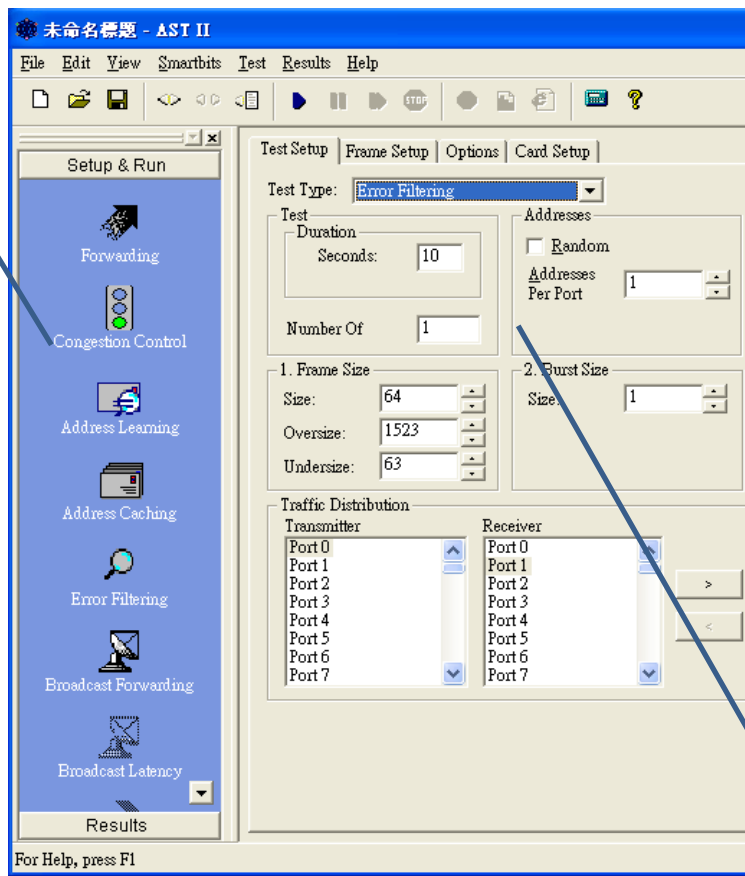
Virus detected by ClamAV

# Lab 12 建置入侵偵測防禦系統及弱點偵測掃描系統

- 了解 bridge 用處及如何建立 bridge
- 熟悉 snort rule 格式
- `reject tcp any any -> 192.168.1.21 22 (sid:1001; content:"/bin/sh"; msg:"Possible SSH buffer overflow"; )`

# Lab 13 以SmartBits來測試Layer 2/3交換器 (1/2)

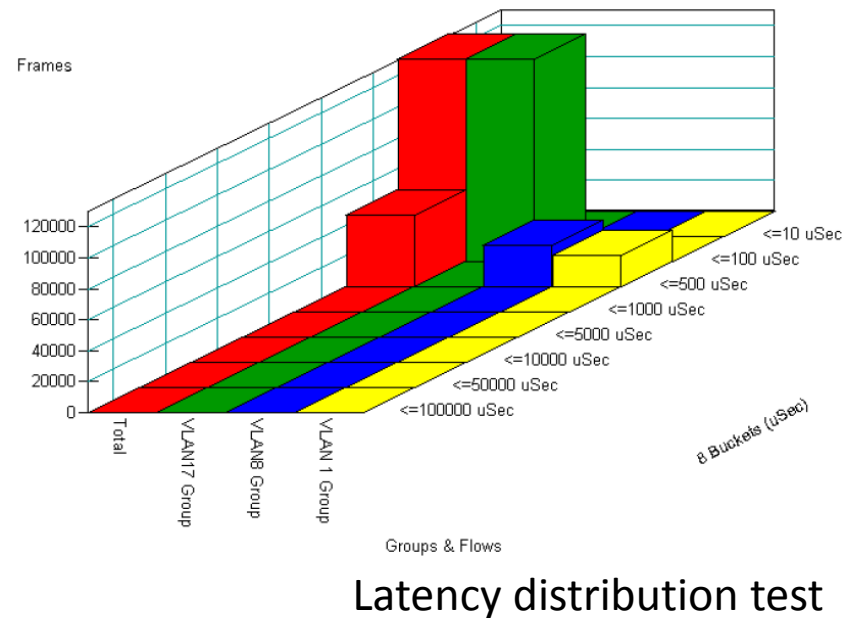
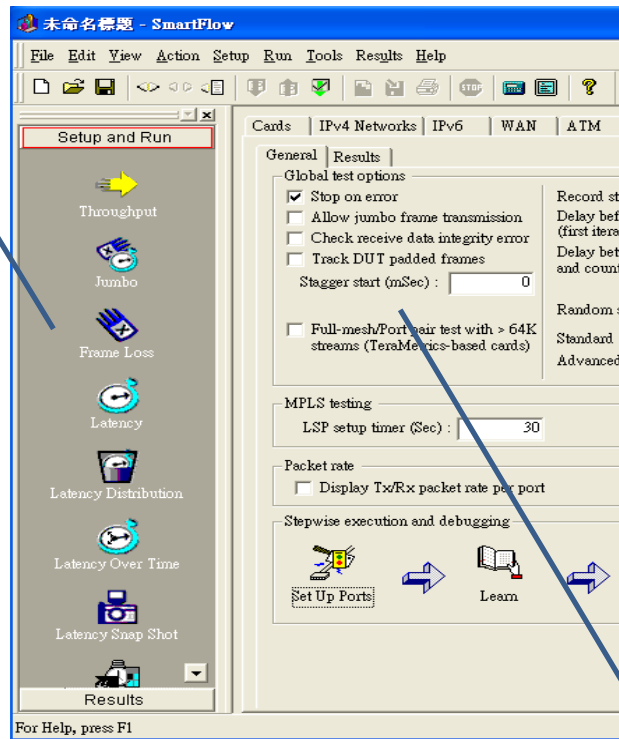
- 利用AST II測試Layer2 switch



Configurations

# Lab 13以SmartBits來測試Layer 2/3交換器 (2/2)

- 利用SmartFlow測試Layer3 router



Configurations