

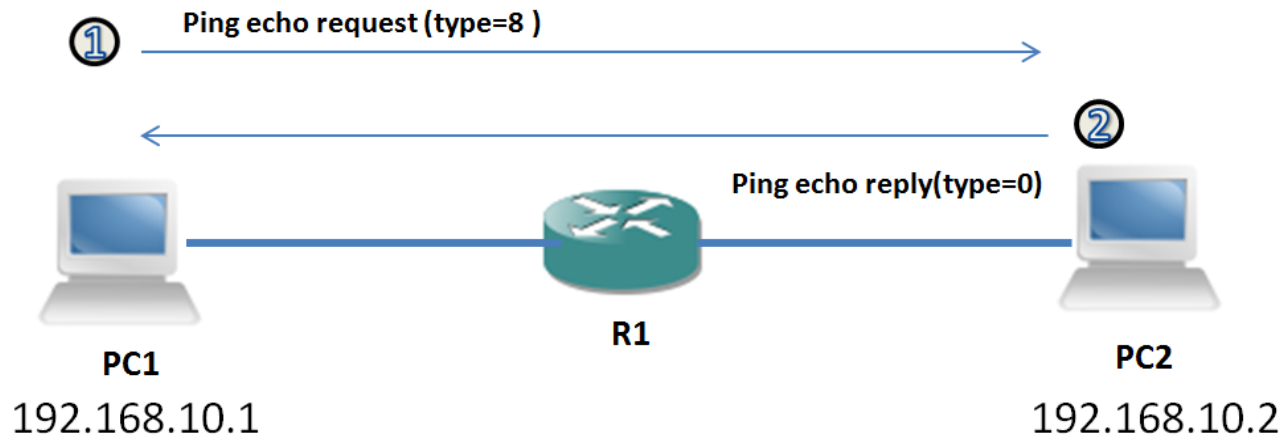
ICMP協定

ICMP協定

- **ICMP (Internet Control Message Protocol)**主要是用來回報網路狀況給管理者
- 管理者從ICMP回傳的資訊就可以判斷目前網路遭遇到的問題
- ICMP 屬於在網路層運作的協定，當作是IP的輔助協定，

Ping運作原理

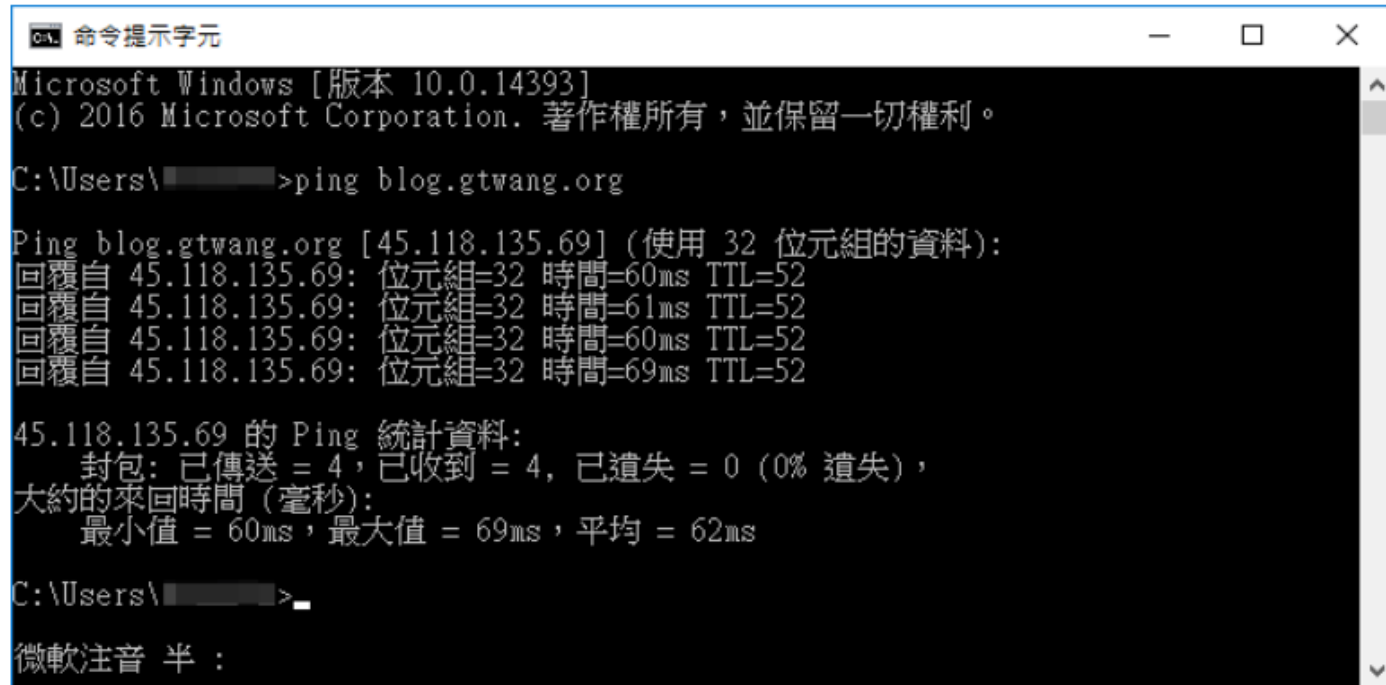
- Ping是最常用來測試網路連線的工具，主要測試來源端與目的端的設備連線是否連通
- ping的運作原理像是迴音(echo)，如果有聽到自己的迴音就代表有網路有通，若沒有則代表網路發生問題
- ping 的兩種封包-Echo Request封包與Echo Reply封包



Windows

在 Windows 中若要使用 `ping` 檢查網路，只要打開「命令提示字元」後輸入 `ping` 的指令加上主機的位址即可使用，若不加任何額外參數的話，`ping` 預設會送出 4 個 ICMP ECHO_REQUEST 封包，並統計測試結果。

```
ping blog.gtwang.org
```



```
命令提示字元
Microsoft Windows [版本 10.0.14393]
(c) 2016 Microsoft Corporation. 著作權所有，並保留一切權利。
C:\Users\>ping blog.gtwang.org

Ping blog.gtwang.org [45.118.135.69] (使用 32 位元組的資料):
回覆自 45.118.135.69: 位元組=32 時間=60ms TTL=52
回覆自 45.118.135.69: 位元組=32 時間=61ms TTL=52
回覆自 45.118.135.69: 位元組=32 時間=60ms TTL=52
回覆自 45.118.135.69: 位元組=32 時間=69ms TTL=52

45.118.135.69 的 Ping 統計資料:
    封包: 已傳送 = 4, 已收到 = 4, 已遺失 = 0 (0% 遺失),
    大約的來回時間 (毫秒):
        最小值 = 60ms, 最大值 = 69ms, 平均 = 62ms

C:\Users\>
```

命令提示字元

指定 ICMP 封包數

如果覺得 4 個 ICMP 封包太少，可以使用 `-n` 參數指定封包數，例如發送 10 個 ICMP 封包：

```
ping -n 10 blog.gtwang.org
```

持續不斷 Ping 主機

有時候在測試與檢修網路時，我們需要持續監看網路是否正常，這時候可以加上 `-t` 參數，讓 `ping` 持續不斷的 ping 特定主機，直到手動按下 `Ctrl + c` 為止：

```
ping -t blog.gtwang.org
```



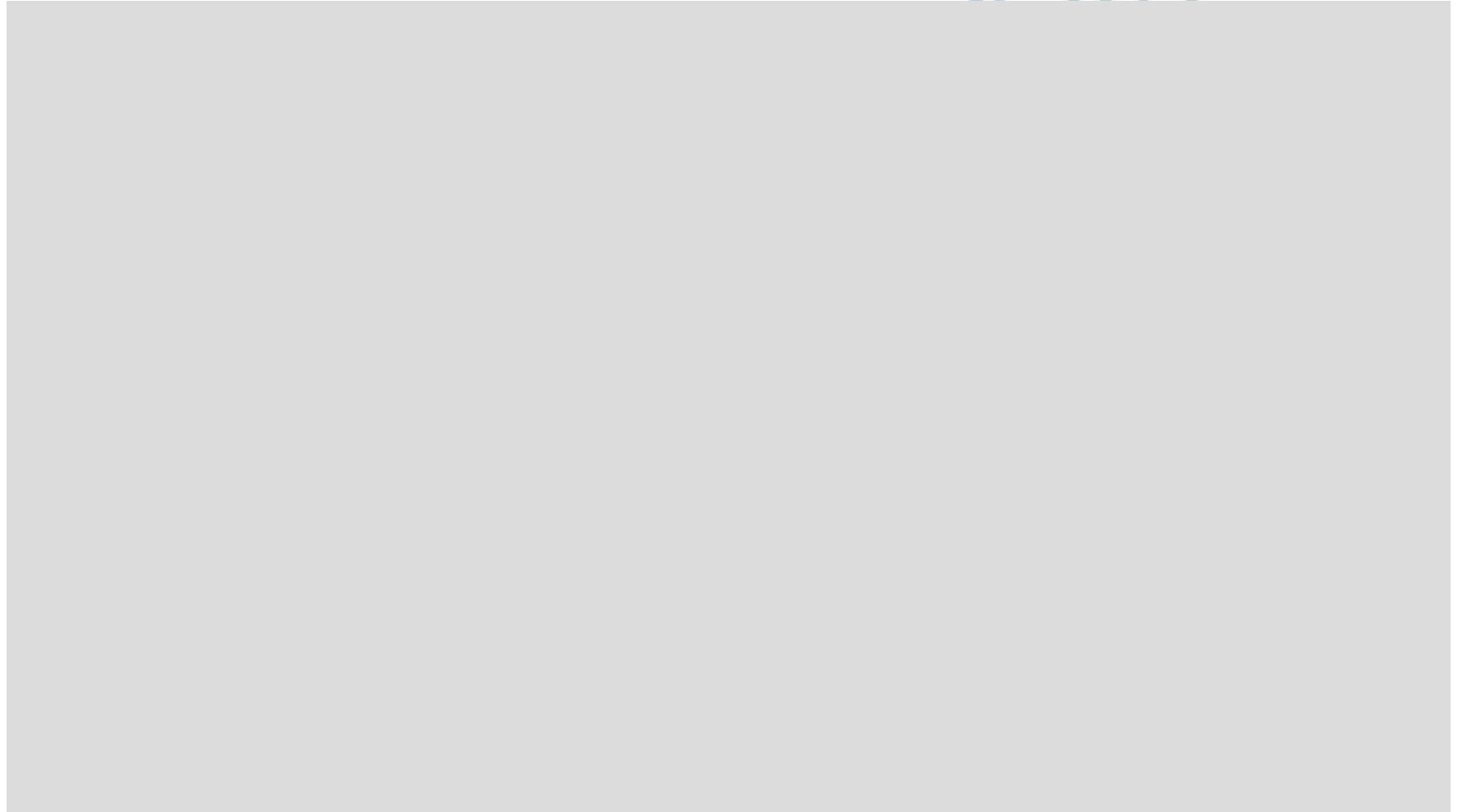
這個狀況其實很常會遇到，在網路不通時我們會檢查各種可能出問題的地方，例如防火牆是否有設定錯誤、網路卡是否正常、網路線是否脫落、路由器是否當機等等，在檢查與修正之前我們通常會先執行這種持續性的 `ping` 指令（通常這時候是沒有回應的），然後才進行各種嘗試，直到 `ping` 的結果出現回應為止。

阻斷服務攻擊

- 阻斷服務（ Denial-of-Service，簡稱DoS）攻擊
- PC0執行ping -n 100 10.10.10.1，表示PC0產生100個ping封包給PC1，當100個ping封包送給PC1，PC1就要回應100次，如果n用的很大，就成為阻斷服務攻擊



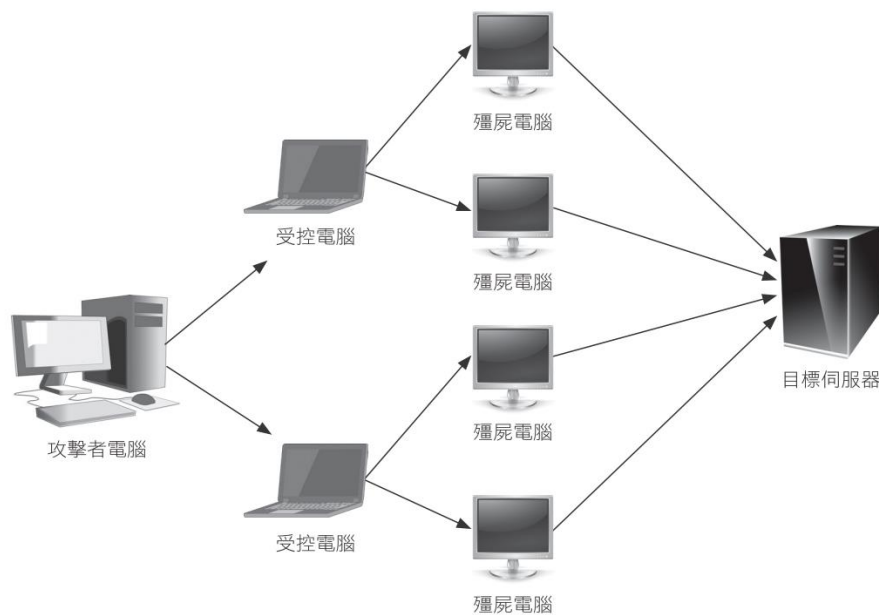
分散式阻斷服務攻擊





分散式阻斷服務攻擊

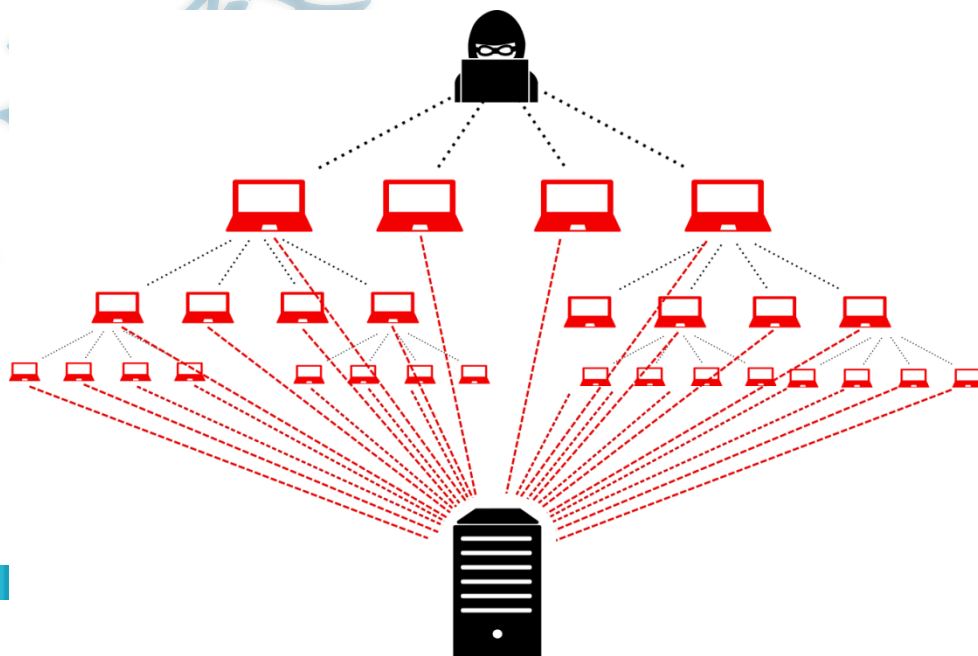
- 分散式阻斷服務攻擊(DDoS)模式是以多對一的方式，透過許多攻擊節點同時對一個目標發動攻擊，而這些發動攻擊的節點，是已經被入侵而不自知的『受控電腦』。
- 『殭屍電腦』(zombie computer)是被受控電腦安裝殭屍程式(zombie)的裝置，受控電腦可以指揮殭屍電腦。由攻擊者電腦發動攻擊命令給受控電腦，受控電腦收到命令後指揮殭屍電腦對目標伺服器發動一波一波的攻擊，目標伺服器因而耗盡資源，無法提供正常服務。





分散式阻斷服務攻擊

- 由於這種攻擊方式是以遠端遙控方式，因此不僅難以防範，更是不易追查主要的攻擊者來源，且只能消極式預防。
- 為避免自己的電腦被安裝DDoS受控制程式，系統管理者必須經常注意系統漏洞及修補(patch)漏洞，經常性的注意及掃描系統有無異常現象，確保自己的機器不被植入DDoS的受控制程式。





- ```
C:\ Administrator: Command Prompt
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\>ping -t -l 65500 127.0.0.1

Pinging 127.0.0.1 with 65500 bytes of data:
Reply from 127.0.0.1: bytes=65500 time<1ms TTL=128
Reply from 127.0.0.1: bytes=65500 time<1ms TTL=128
Reply from 127.0.0.1: bytes=65500 time<1ms TTL=128
Reply from 127.0.0.1: bytes=65500 time<1ms TTL=128
Reply from 127.0.0.1: bytes=65500 time<1ms TTL=128
Reply from 127.0.0.1: bytes=65500 time<1ms TTL=128
Reply from 127.0.0.1: bytes=65500 time<1ms TTL=128
Reply from 127.0.0.1: bytes=65500 time<1ms TTL=128
Reply from 127.0.0.1: bytes=65500 time<1ms TTL=128
Reply from 127.0.0.1: bytes=65500 time<1ms TTL=128
Reply from 127.0.0.1: bytes=65500 time<1ms TTL=128
Reply from 127.0.0.1: bytes=65500 time<1ms TTL=128
Reply from 127.0.0.1: bytes=65500 time<1ms TTL=128
Reply from 127.0.0.1: bytes=65500 time<1ms TTL=128
Reply from 127.0.0.1: bytes=65500 time<1ms TTL=128
Ping statistics for 127.0.0.1:
 Packets: Sent = 13, Received = 13, Lost = 0 (0% loss),
 Approximate round trip times in milli-seconds:
 Minimum = 0ms, Maximum = 0ms, Average = 0ms
Control-C
```

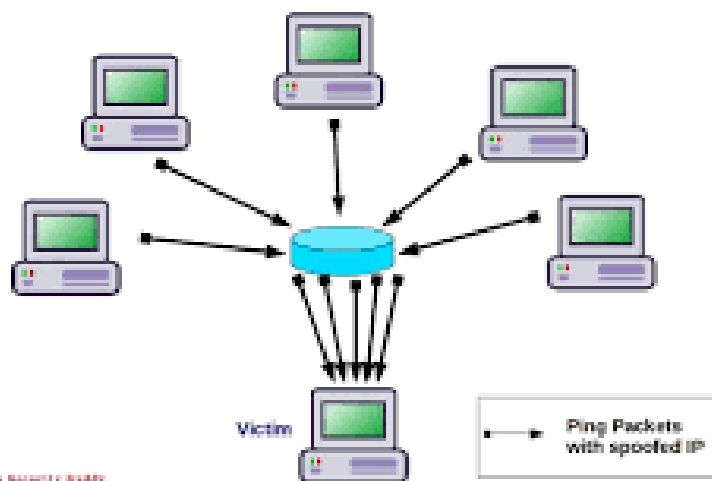


- 
- A screenshot of a Windows Administrator Command Prompt window. The title bar reads "Administrator: Command Prompt". The command prompt shows the execution of the command "ping -t -l 65500 127.0.0.1". The output displays ten successful replies from 127.0.0.1, each with a response time of approximately 0ms and TTL=128. Below the replies, it shows "Ping statistics for 127.0.0.1:" followed by summary statistics: Packets: Sent = 10, Received = 10, Lost = 0 (0% loss), Approximate round trip times in milli-seconds: Minimum = 0ms, Maximum = 0ms, Average = 0ms. The prompt ends with "Control-C".
- ```
Administrator: Command Prompt  
Microsoft Windows [Version 6.1.7601]  
Copyright (c) 2009 Microsoft Corporation. All rights reserved.  
  
C:\Users\██████>ping -t -l 65500 127.0.0.1  
  
Pinging 127.0.0.1 with 65500 bytes of data:  
Reply from 127.0.0.1: bytes=65500 time<1ms TTL=128  
Reply from 127.0.0.1: bytes=65500 time<1ms TTL=128  
Reply from 127.0.0.1: bytes=65500 time<1ms TTL=128  
Reply from 127.0.0.1: bytes=65500 time<1ms TTL=128  
Reply from 127.0.0.1: bytes=65500 time<1ms TTL=128  
Reply from 127.0.0.1: bytes=65500 time<1ms TTL=128  
Reply from 127.0.0.1: bytes=65500 time<1ms TTL=128  
Reply from 127.0.0.1: bytes=65500 time<1ms TTL=128  
Reply from 127.0.0.1: bytes=65500 time<1ms TTL=128  
Reply from 127.0.0.1: bytes=65500 time<1ms TTL=128  
  
Ping statistics for 127.0.0.1:  
    Packets: Sent = 10, Received = 10, Lost = 0 (0% loss),  
        Approximate round trip times in milli-seconds:  
            Minimum = 0ms, Maximum = 0ms, Average = 0ms  
Control-C
```



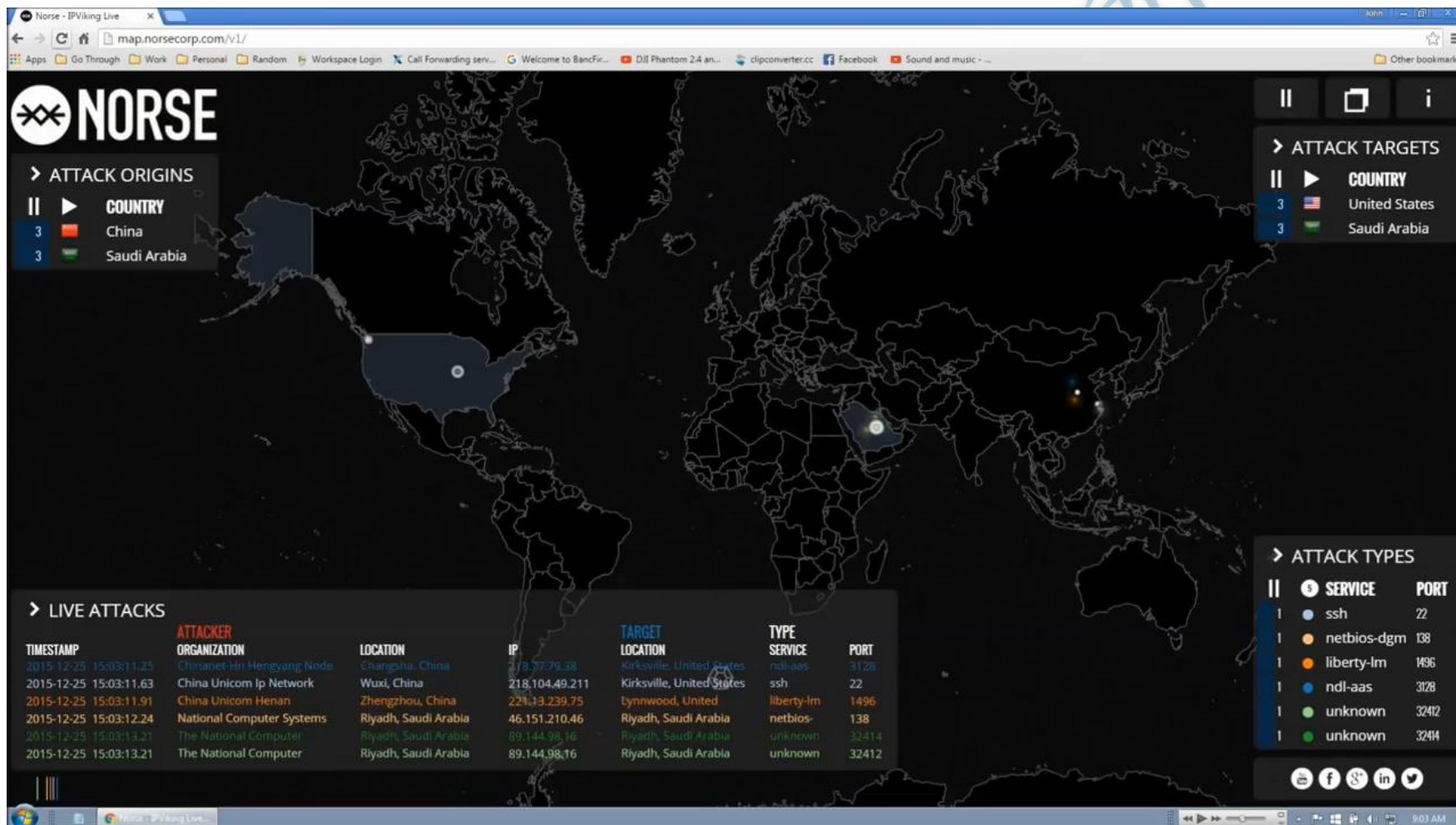
死亡之Ping 與Ping泛洪攻擊

- 早期的TCP/IP系統中，Ping of death攻擊很容易實現，曾經影響許多系統，如Unix、Linux、Mac、Windows、printers、以及routers等，這是專門針對一種協定弱點的簡單攻擊。然而大部分系統已經在1998年後修正這些漏洞，所以這個攻擊已經成為歷史名詞。
- 近年來，出現了**Ping泛洪攻擊(Ping flood)**:主要攻擊方法是發送大量Ping封包到受害主機，以致於正常封包無法送達主機接受服務。

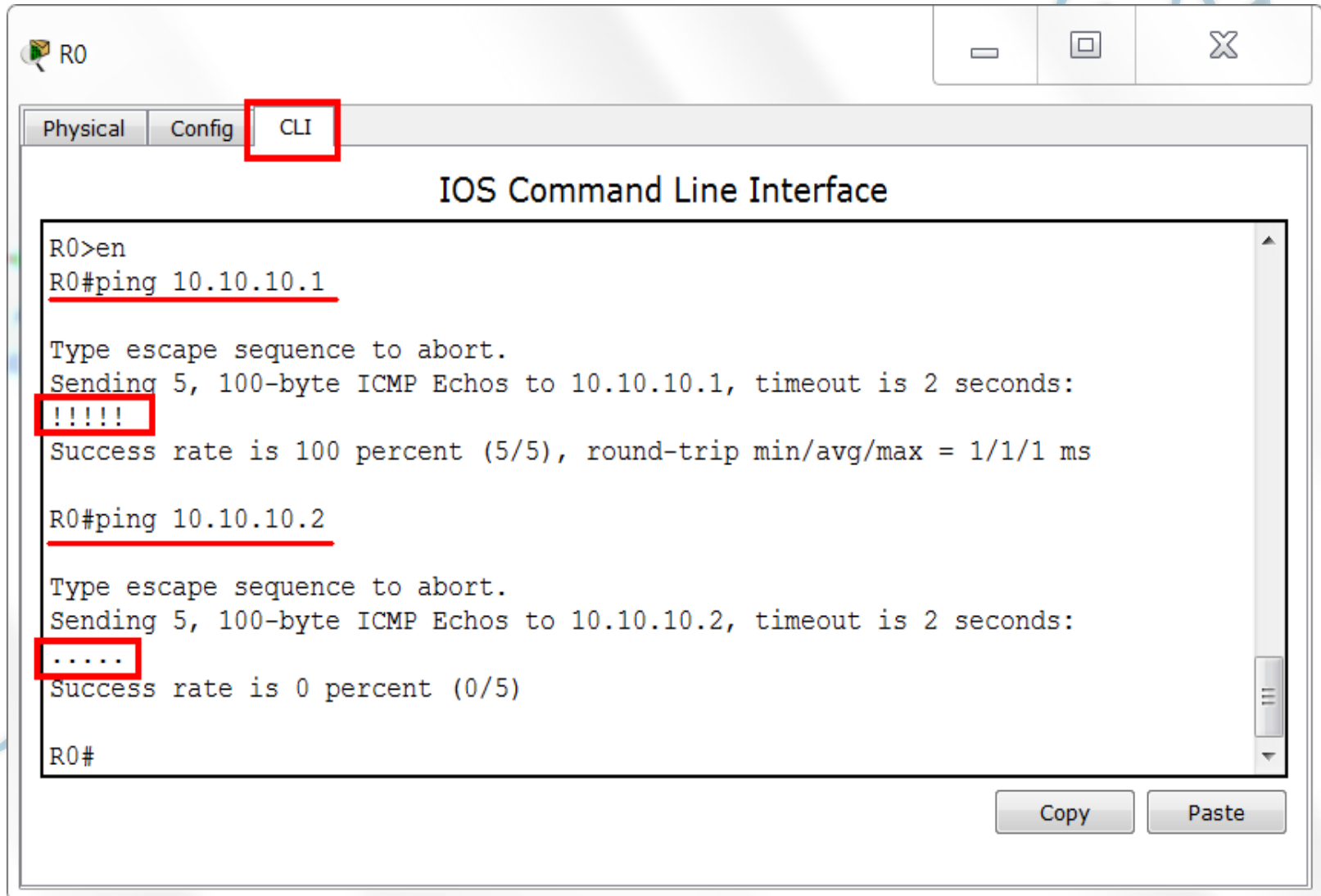




分散式阻斷服務攻擊



Router執行ping 的結果



使用Wireshark看ping封包

*區域連線 [Wireshark 1.10.0 (SVN Rev 49790 from /trunk-1.10)]

Filter: icmp Expression... Clear Apply Save

No.	Time	Source	Destination	Protocol	Length	Info
7	11.8800810	192.168.1.103	173.194.72.94	ICMP	74	Echo (ping) request id=0x0001, seq=14/3584, ttl=128 (reply in 8)
8	11.9023810	173.194.72.94	192.168.1.103	ICMP	74	Echo (ping) reply id=0x0001, seq=14/3584, ttl=51 (request in 7)
9	12.8836630	192.168.1.103	173.194.72.94	ICMP	74	Echo (ping) request id=0x0001, seq=15/3840, ttl=128 (reply in 10)

Frame 7: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface 0

- Ethernet II, Src: QuantaCo_87:89:ea (c8:0a:a9:87:89:ea), Dst: ZyxelCom_20:75:11 (c8:6c:87:20:75:11)
- Internet Protocol Version 4, Src: 192.168.1.103 (192.168.1.103), Dst: 173.194.72.94 (173.194.72.94)
 - Version: 4
 - Header length: 20 bytes
 - Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))
 - Total Length: 60
 - Identification: 0x4c4f (19535)
 - Flags: 0x00
 - Fragment offset: 0
 - Time to live: 128
 - Protocol: ICMP (1)
 - Header checksum: 0x3642 [correct]
 - Source: 192.168.1.103 (192.168.1.103)
 - Destination: 173.194.72.94 (173.194.72.94)
 - [Source GeoIP: Unknown]
 - [Destination GeoIP: Unknown]
- Internet Control Message Protocol
 - Type: 8 (Echo (ping) request)
 - Code: 0
 - Checksum: 0x4d4d [correct]
 - Identifier (BE): 1 (0x0001)
 - Identifier (LE): 256 (0x0100)
 - Sequence number (BE): 14 (0x000e)
 - Sequence number (LE): 3584 (0x0e00)
 - [\[Response frame: 8\]](#)
- Data (32 bytes)

Offset	Hex	ASCII
0000	c8 6c 87 20 75 11 c8 0a a9 87 89 ea 08 00 45 00	.l.u... ..E.
0010	00 3c 4c 4f 00 00 80 01 36 42 c0 a8 01 67 ad c2	.<LO... 6B...g.
0020	48 5e 08 00 4d 4d 00 01 00 0e 61 62 63 64 65 66	HA..MM.. ..abcdef
0030	67 68 69 6a 6b 6c 6d 6e 6f 70 71 72 73 74 75 76	ghijklmn opqrstuv

Frame (frame), 74 bytes Packets: 14 · Displayed: 8 (57.1%) · Dropped: 0 (0.0%) Profile: Default

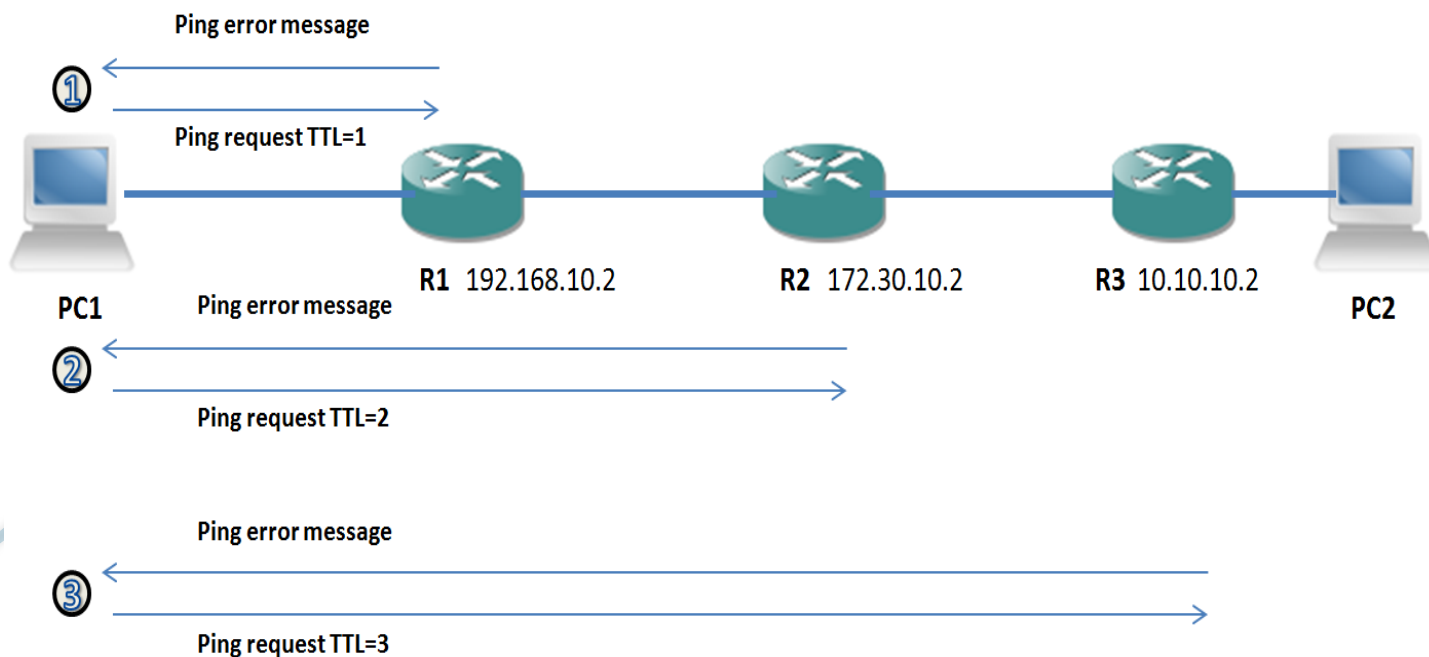
網路模型

ICMP type代表意思

Type 欄位值	ICMP 封包類型
0	Echo Reply*
3	Destination Unreachable*
4	Source Quench*
5	Redirect*
8	Echo Request*
11	Time Exceeded for a Datagram*
12	Parameter Problem on a Datagram
13	Timestamp Request
14	Timestamp Reply

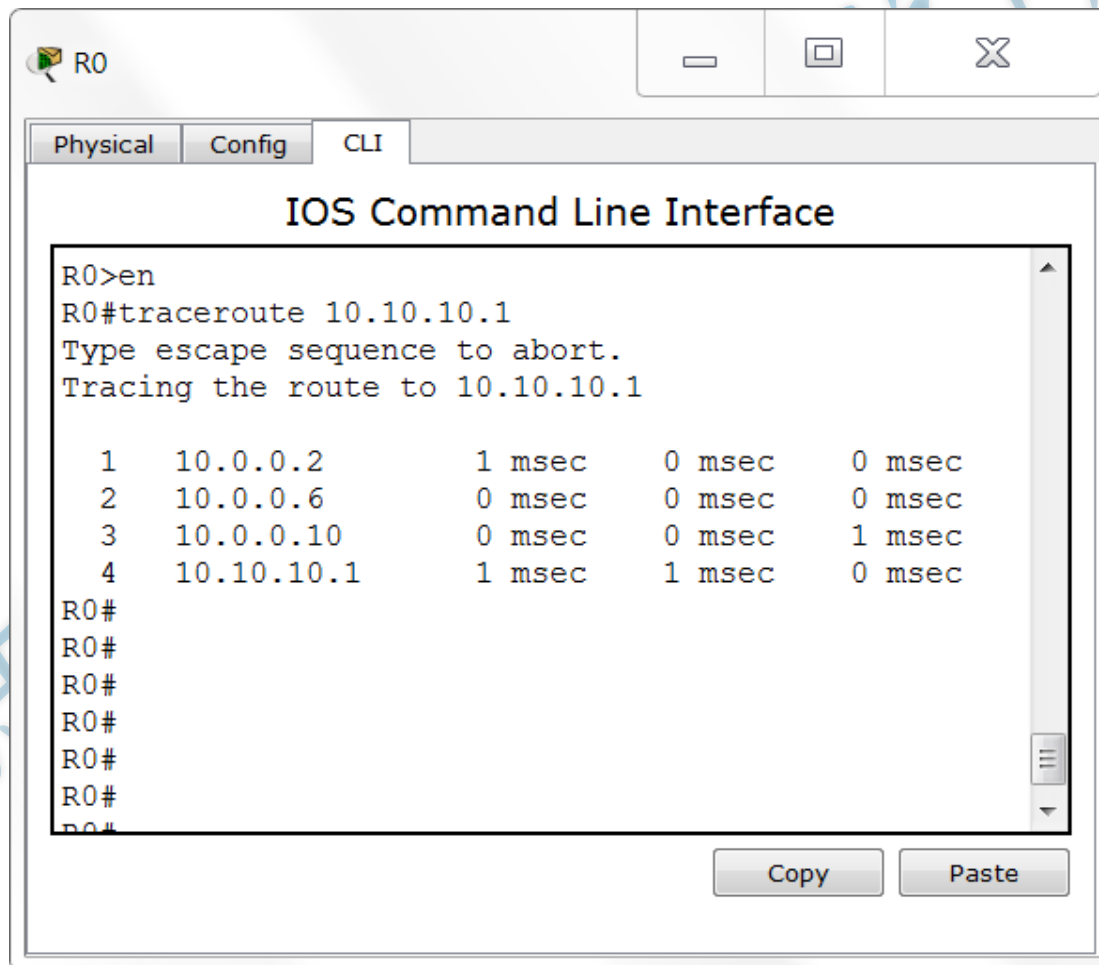
Traceroute 作用

- Ping只會顯示來源端與目的端的網路連線結果
- **Traceroute** 會顯示經過的路由器的IP
- Traceroute如何做到顯示經過的路由器，這也是使用ping的機制



Router traceroute

- DOS的指令是**tracert**，而路由器是**traceroute**，兩這指令的用法都是一樣



The screenshot shows a Cisco IOS Command Line Interface (CLI) window for router R0. The window has tabs for Physical, Config, and CLI. The CLI tab is active, and the title bar says "IOS Command Line Interface". The command prompt is "R0>". The user has entered "en" to enter enable mode, and then "traceroute 10.10.10.1". The output shows the route to 10.10.10.1, with four hops. The first hop is 10.0.0.2, the second is 10.0.0.6, the third is 10.0.0.10, and the fourth is 10.10.10.1. The output also shows the time taken for each hop: 1 msec, 0 msec, 0 msec, and 1 msec respectively. The window also has a "Copy" button and a "Paste" button.

```
R0>en
R0#traceroute 10.10.10.1
Type escape sequence to abort.
Tracing the route to 10.10.10.1

 1  10.0.0.2          1 msec    0 msec    0 msec
 2  10.0.0.6          0 msec    0 msec    0 msec
 3  10.0.0.10         0 msec    0 msec    1 msec
 4  10.10.10.1        1 msec    1 msec    0 msec
R0#
R0#
R0#
R0#
R0#
R0#
R0#
```

使用Wireshark 觀察tracert

Capturing from 區域連線 [Wireshark 1.10.0 (SVN Rev 49790 from /trunk-1.10)]

File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help

Filter: icmp Expression... Clear Apply Save

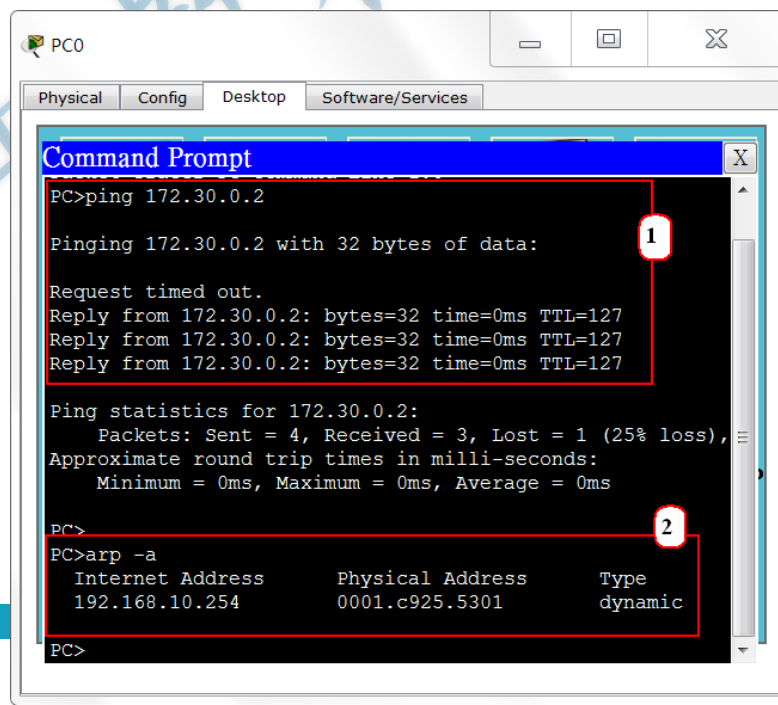
No.	Time	Source	Destination	Protocol	Length	Info
3	0.52992400	192.168.1.103	74.125.31.105	ICMP	106	Echo (ping) request id=0x0001, seq=112/28672, ttl=1
4	0.53035400	192.168.1.1	192.168.1.103	ICMP	134	Time-to-live exceeded (Time to live exceeded in transit)
5	0.53128900	192.168.1.103	74.125.31.105	ICMP	106	Echo (ping) request id=0x0001, seq=113/28928, ttl=1
6	0.53165100	192.168.1.1	192.168.1.103	ICMP	134	Time-to-live exceeded (Time to live exceeded in transit)
7	0.53263100	192.168.1.103	74.125.31.105	ICMP	106	Echo (ping) request id=0x0001, seq=114/29184, ttl=1
8	0.53294100	192.168.1.1	192.168.1.103	ICMP	134	Time-to-live exceeded (Time to live exceeded in transit)
9	1.53451700	192.168.1.103	74.125.31.105	ICMP	106	Echo (ping) request id=0x0001, seq=115/29440, ttl=2
10	1.55103100	168.95.98.254	192.168.1.103	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
11	1.55206300	192.168.1.103	74.125.31.105	ICMP	106	Echo (ping) request id=0x0001, seq=116/29696, ttl=2
12	1.56875500	168.95.98.254	192.168.1.103	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
13	1.56973000	192.168.1.103	74.125.31.105	ICMP	106	Echo (ping) request id=0x0001, seq=117/29952, ttl=2
14	1.58647400	168.95.98.254	192.168.1.103	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
17	2.57260300	192.168.1.103	74.125.31.105	ICMP	106	Echo (ping) request id=0x0001, seq=118/30208, ttl=3
18	2.58891900	168.95.94.14	192.168.1.103	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
19	2.58990900	192.168.1.103	74.125.31.105	ICMP	106	Echo (ping) request id=0x0001, seq=119/30464, ttl=3
20	2.60619100	168.95.94.14	192.168.1.103	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
21	2.60712100	192.168.1.103	74.125.31.105	ICMP	106	Echo (ping) request id=0x0001, seq=120/30720, ttl=3
22	2.62341200	168.95.94.14	192.168.1.103	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
23	3.60956600	192.168.1.103	74.125.31.105	ICMP	106	Echo (ping) request id=0x0001, seq=121/30976, ttl=4
24	3.62565700	220.128.8.209	192.168.1.103	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
25	3.62679100	192.168.1.103	74.125.31.105	ICMP	106	Echo (ping) request id=0x0001, seq=122/31232, ttl=4
26	3.64308400	220.128.8.209	192.168.1.103	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
27	3.64392600	192.168.1.103	74.125.31.105	ICMP	106	Echo (ping) request id=0x0001, seq=123/31488, ttl=4
28	3.66020700	220.128.8.209	192.168.1.103	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
29	4.64660800	192.168.1.103	74.125.31.105	ICMP	106	Echo (ping) request id=0x0001, seq=124/31744, ttl=5
30	4.66462700	211.22.226.5	192.168.1.103	ICMP	70	Time-to-live exceeded (Time to live exceeded in transit)
31	4.66536500	192.168.1.103	74.125.31.105	ICMP	106	Echo (ping) request id=0x0001, seq=125/32000, ttl=5

Frame (frame), 106 bytes Packets: 186 · Displayed: 51 (27.4%) Profile: Default

2.9 ARP 協定

ARP 協定

- ARP(Address Resolution Protocol)協定功能就是給定一個同網路的電腦IP位址，這個協定會找出有該IP的電腦MAC，ARP的運作有三個步驟：
 1. 來源電腦發送廣播資料，含要尋找目的IP。
 2. 同一網路下的電腦收到廣播資料後，進行比對本身的IP與廣播資料中要尋找目的IP。
 3. 比對正確的電腦回應本身的MAC給來源電腦。



The screenshot shows a PC0 window with a Command Prompt. The window has tabs for Physical, Config, Desktop, and Software/Services. The Command Prompt shows the following commands and output:

```
PC>ping 172.30.0.2

Pinging 172.30.0.2 with 32 bytes of data:

Request timed out.
Reply from 172.30.0.2: bytes=32 time=0ms TTL=127
Reply from 172.30.0.2: bytes=32 time=0ms TTL=127
Reply from 172.30.0.2: bytes=32 time=0ms TTL=127

Ping statistics for 172.30.0.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

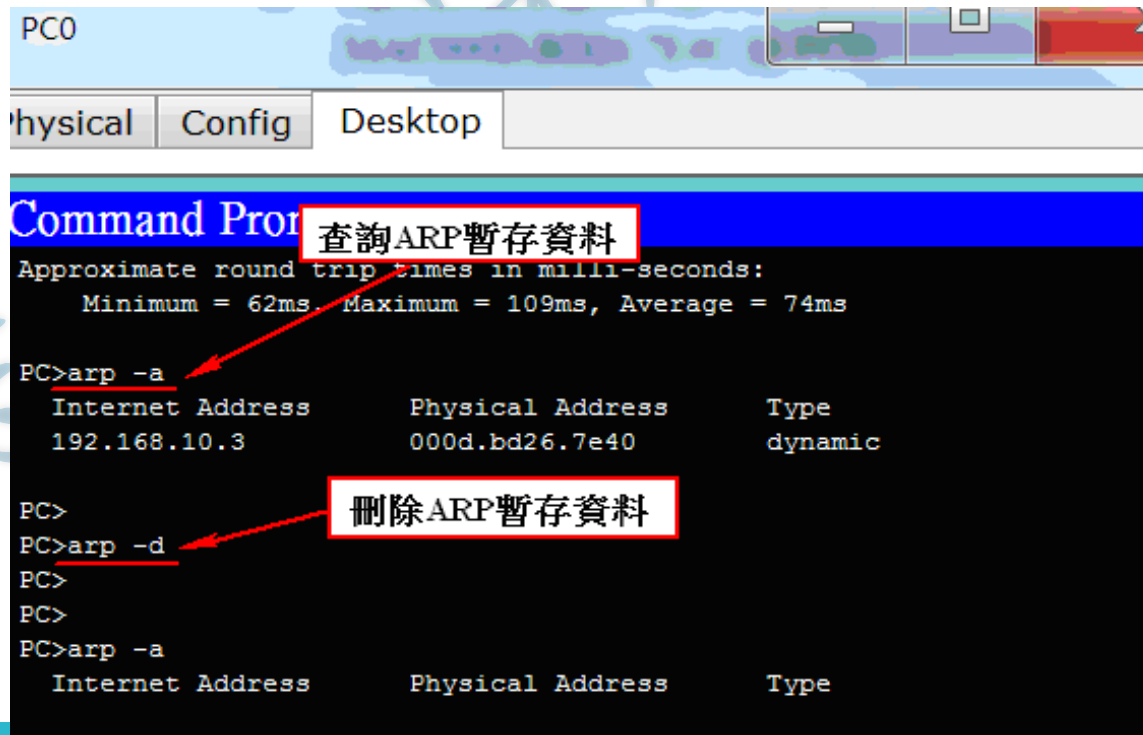
PC>
PC>arp -a
```

Internet Address	Physical Address	Type
192.168.10.254	0001.c925.5301	dynamic

The output of the `arp -a` command is displayed in a table format. The table has three columns: Internet Address, Physical Address, and Type. The first row shows the IP address 192.168.10.254, the MAC address 0001.c925.5301, and the type dynamic.

ARP暫存查詢與刪除

- 當ARP已經有詢問目的IP的MAC位址後，會將目的IP對應的MAC暫存起來
- 在電腦的DOS模式使用arp -a 來查詢ARP暫存資料，PC0剛剛有使用ARP詢問192.168.10.3的MAC位址，因此PC0會暫存此筆對應如下圖所示
- 使用arp -d為刪除ARP暫存資料。



The screenshot shows a PC0 interface with a 'Command Prompt' window open. The window title is 'PC0'. The interface has tabs for 'physical', 'Config', and 'Desktop'. The Command Prompt displays the following text:

```
Approximate round trip times in milli-seconds:
  Minimum = 62ms, Maximum = 109ms, Average = 74ms

PC>arp -a
Internet Address      Physical Address      Type
192.168.10.3          000d.bd26.7e40        dynamic

PC>
PC>arp -d
PC>
PC>
PC>arp -a
Internet Address      Physical Address      Type
```

Two red arrows point to the commands in the Command Prompt:

- A red arrow points from the text '查詢ARP暫存資料' (Query ARP temporary storage) to the command 'arp -a'.
- A red arrow points from the text '刪除ARP暫存資料' (Delete ARP temporary storage) to the command 'arp -d'.

路由器的ARP暫存資料

- 路由器也會執行ARP協定來詢問目的IP的MAC，所以路由器會有ARP的暫存資料
- 使用IOS的show arp指令來查詢，查詢結果如下圖所示

```
o up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to
o up

Router>
Router>en
Router#show arp
Router#show arp
Protocol  Address          Age (min)  Hardware Addr  Type   Interface
Internet  172.30.0.1        0          000D.BD26.7E40  ARPA   FastEthernet0/1
Internet  172.30.0.2        6          0005.5E95.1232  ARPA   FastEthernet0/1
Internet  172.30.0.254      -          0001.C925.5302  ARPA   FastEthernet0/1
Internet  192.168.10.1      6          000D.BD93.ACE1  ARPA   FastEthernet0/0
Internet  192.168.10.2      0          0001.6340.0CD3  ARPA   FastEthernet0/0
Internet  192.168.10.254    -          0001.C925.5301  ARPA   FastEthernet0/0
Router#
```

ARP攻擊

- 駭客會使用ARP運作的缺點來攻擊網路或是竊取資料，最常見的攻擊手法就是將電腦中ARP暫存資料，讓目的IP對應到假的MAC位址
- 如此就無法將資料送到目的IP電腦，造成網路無法連結，這就稱作ARP攻擊
- 如果要避免ARP攻擊，可以在Switch啟動Dynamic ARP inspection (DAI) 功能，DAI會檢查合法的IP與合法MAC對應，如此就可以檢查出攻擊者的造假的IP與MAC對應關係，DAI屬於CCNP課程範圍。

ARP攻擊

有

