工資系 103 級 林秉錚

H34996108

執行環境與操作：

a. OS：UBUNTU 13.10

b. compile：gcc -o arp arp_reply_attack.c

c. run：./arp

d.有小 bug，必須在 tcpdump 監聽下才會抓到正確的被攻擊者 MAC address，直接執行 arp 會抓錯。

程式解釋：

a. 程式裡面所使用的函式與自己比較不清楚的函式使用：

1. char *allocate_strmem (int len);為分配記憶體空間給 char 型態的陣列空間。
2. uint8_t * allocate_ustrmem (int len); 為分配記憶體空間給 unsigned char 型態的陣列空間。
3. inet_pton(int *af*, const char **src*, void **dst*)：將 src 字串轉成網路位址 af 的 address family 結構，並將結果複製到 dst 字串，且 af 參數需為 AF_INET 或者 AF_INET6，回傳值部份為 1 是成功轉換，0 為 src 字串非正確的 address family 結構，-1 則是 af 為不正確的 address family。

b.變數說明：

1. int sd：socket descripter。
2. char* interface：網路的介面名字。
3. char* target：接收端的 ip 或者 URL，必須在 LAN 內(link-local node)。
4. char* src_ip：傳送端的 ip。
5. uint8_t *dst_mac：接收端的 MAC 位址。
6. uint8_t *src_mac：傳送端的 MAC 位址。

7. uint8_t *ether_frame：ethernet 的框架，框架結構為 ethernet header(接收端的 MAC(6)+傳送端的 MAC(6)+ethernet type(2)) 加上 ethernet data(ARP 標頭(28))，括號數字內為 byte 數。
8. struct sockaddr_in *ipv4：IPv4 的 socket 位址結構。

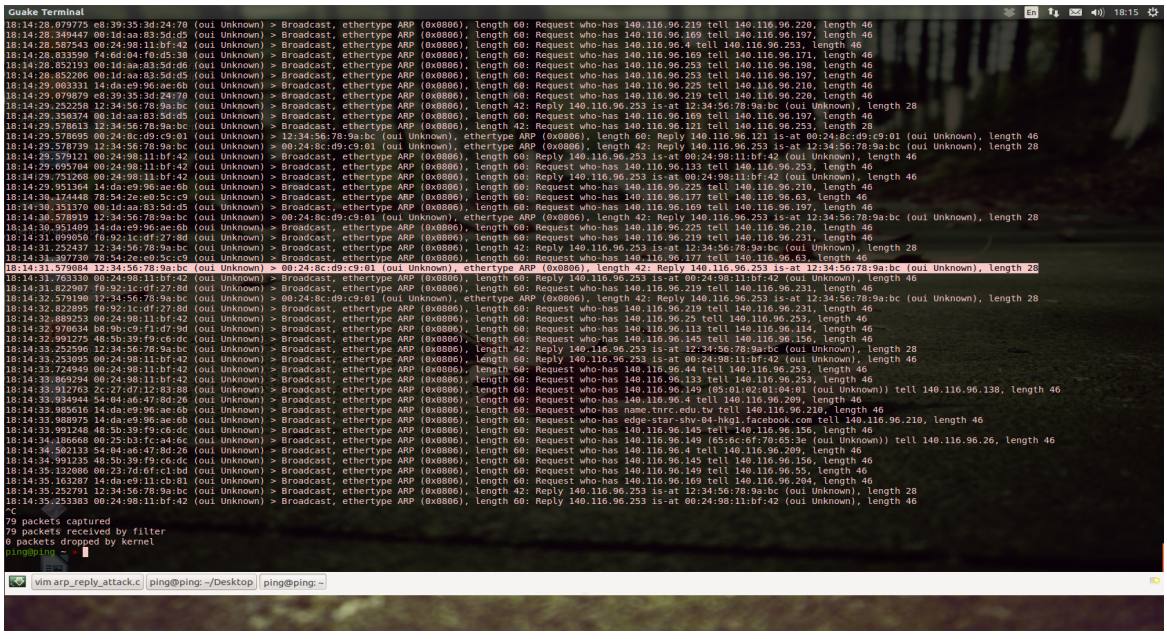
c. 程式概述：

此程式的被攻擊者為 **140.116.96.121**，其 MAC address 為 **00:24:8c:d9:c9:01**。

Main function 開始，一開始先分配記憶體給各個所需陣列，接著原本的程式是利用 socket 得到傳送端的 MAC 位址及網路介面並輸出，這邊可以將其改成任意的 MAC 位址來騙人。

之後輸出 interface 的 index，藉由 if_nametoindex(介面名子)的方式。

之後流程為設定傳送端 MAC 位址為 FF：FF：FF：FF：FF：FF(即 broadcast)的 ARP request，以取得被攻擊者的 MAC address(藉由察看 arp header 的第 22 個數字是否為 2，2 即是被攻擊者的 ARP REPLY，來確認是否封包為被攻擊者的)，之後就可以從 arp header 裏面得到被攻擊者的 MAC address 了。最後再用 ARP REPLY 的方式告知被攻擊者說 gateway 的 MAC ADDRESS 是 12:34:56:78:9A:BC 也就是假的 MAC address，以達到攻擊的目的。



附圖反白部份為最後 ARP REPLY 的部份，反白部份文字如下。

18:14:31.579084 **12:34:56:78:9a:bc** (oui Unknown) > **00:24:8c:d9:c9:01** (oui Unknown),

ethertype ARP (0x0806), length 42: Reply **140.116.96.253 is-at 12:34:56:78:9a:bc** (oui Unknown), length 28