Capitolo 2

Cattura del traffico con l'ARP Spoofing

Non badare all'uomo dietro la tenda!
– Noel Langley, *Il mago di Oz*

Chiunque si sieda al tavolino di un bar e si connetta alla sua rete WiFi può intercettare e visualizzare il traffico web non crittografato degli altri utenti utilizzando una tecnica chiamata *ARP Spoofing*, che sfrutta una vulnerabilità nella progettazione del protocollo ARP (*Address Resolution Protocol*). In questo capitolo, spieghiamo come funziona il protocollo ARP, descriviamo i passi di un attacco ARP Spoofing e poi proviamo a eseguirlo.

In questo capitolo

- **Come vengono trasmessi i dati in Internet**
- **Attacchi ARP Spoofing**
- **Esecuzione di un attacco ARP Spoofing**
- **Rilevamento di un attacco ARP Spoofing**
- **Esercizi**

Come vengono trasmessi i dati in Internet

Prima di poter parlare di ARP Spoofing, dobbiamo introdurre la struttura generale di Internet. Questo paragrafo spiega come Internet trasmette i dati attraverso una rete gerarchica utilizzando pacchetti, indirizzi MAC e indirizzi IP.

Pacchetti

Tutte le informazioni, su Internet, vengono trasmesse in *pacchetti*. Un pacchetto è come una "busta" che contiene i dati che volete inviare. Come nel caso del servizio postale, questi pacchetti vengono instradati verso le loro destinazioni in base all'indirizzo specificato. La Figura 2.1 mostra alcuni parallelismi tra buste fisiche e pacchetti di dati.

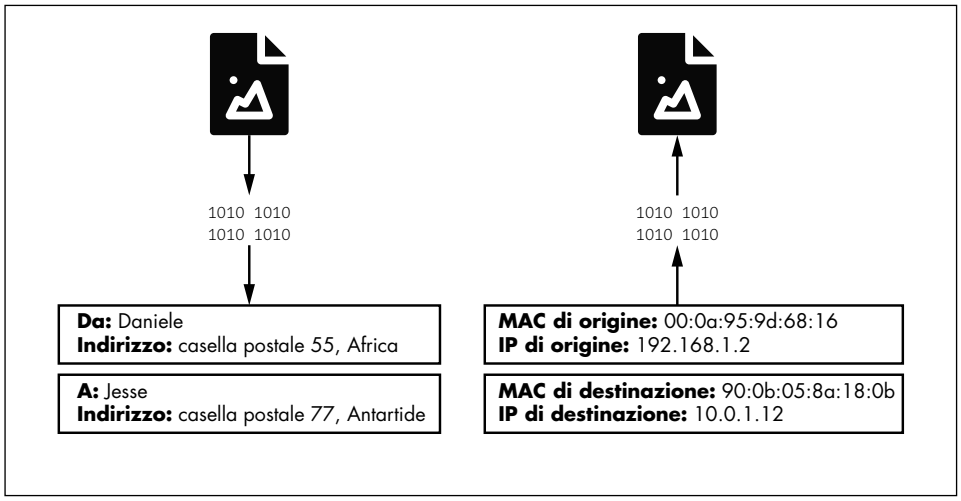

Figura 2.1 Paralleli tra buste e pacchetti.

La sezione Da su una busta contiene due informazioni fondamentali: 1) il nome della persona che invia la lettera e 2) dove vive. Allo stesso modo, i pacchetti hanno l'indirizzo MAC (*Media Access Control*) che rappresenta la macchina che invia il pacchetto e l'indirizzo IP che rappresenta la provenienza del pacchetto. Altri campi simili, le *intestazioni del pacchetto,* rappresentano la destinazione del pacchetto.
Internet utilizza dei dispositivi, detti *router*, per ordinare e inoltrare i pacchetti. I pacchetti attraversano Internet, viaggiando da un router all'altro come la posta viaggia da un ufficio postale all'altro.

Indirizzi MAC

Il vostro portatile contiene una *scheda di interfaccia di rete*, detta NIC (*Network Interface Card*) che gli consente di connettersi ai router WiFi. Questa scheda ha un indirizzo univoco, chiamato indirizzo MAC, che identifica la vostra macchina sulla rete. Quando il router vuole inviare informazioni del vostro computer, etichetta quel pacchetto con l'indirizzo MAC del vostro portatile e poi lo trasmette tramite segnali radio. Tutte le macchine connesse a quel router ricevono questo segnale radio e controllano l'indirizzo MAC del pacchetto per scoprire se il pacchetto è destinato a loro. Gli indirizzi MAC sono normalmente numeri a 48 bit scritti in esadecimale (per esempio, `08:00:27:3b:8f:ed`).

Indirizzi IP

Probabilmente sapete già che anche gli indirizzi IP identificano le macchine su una rete. Allora, perché abbiamo bisogno di entrambi gli indirizzi, IP e MAC? Bene, le reti sono costituite da regioni gerarchiche, un po' come alcuni Paesi sono divisi in Stati, che a loro volta contengono regioni e città. Gli indirizzi IP seguono una struttura che consente loro di identificare la posizione di un dispositivo nella rete più ampia. Se vi trasferiste in un

altro bar, al vostro portatile verrebbe assegnato un indirizzo IP differente, per riflettere la sua nuova posizione; tuttavia, il vostro indirizzo MAC rimarrebbe sempre lo stesso.
Un indirizzo IPv4 codifica le informazioni sulla gerarchia di rete in un numero a 32 bit. Questo numero è tipicamente rappresentato da quattro sezioni separate da punti (come `192.168.3.1`). Ogni sezione rappresenta un numero binario a 8 bit. Per esempio, il `3` in `192.168.3.1` rappresenta il numero binario a 8 bit `00000011`.
Gli indirizzi IP situati nella stessa regione della gerarchia condividono gli stessi bit di livello superiore. Per esempio, tutte le macchine nel campus dell'Università della Virginia hanno indirizzi IPv4 come `128.143.xxx.xxx`. Potrete vedere che questo concetto è scritto in notazione CIDR (*Classless Inter-Domain Routing*) come `128.143.1.1/16`, a indicare che le macchine condividono gli stessi 16 bit superiori, ovvero i primi due numeri. Poiché gli indirizzi IP seguono una determinata struttura, i router possono utilizzare parti dell'indirizzo IP per decidere come instradare un pacchetto attraverso la gerarchia. La Figura 2.2 mostra un esempio semplificato di questa gerarchia di router.

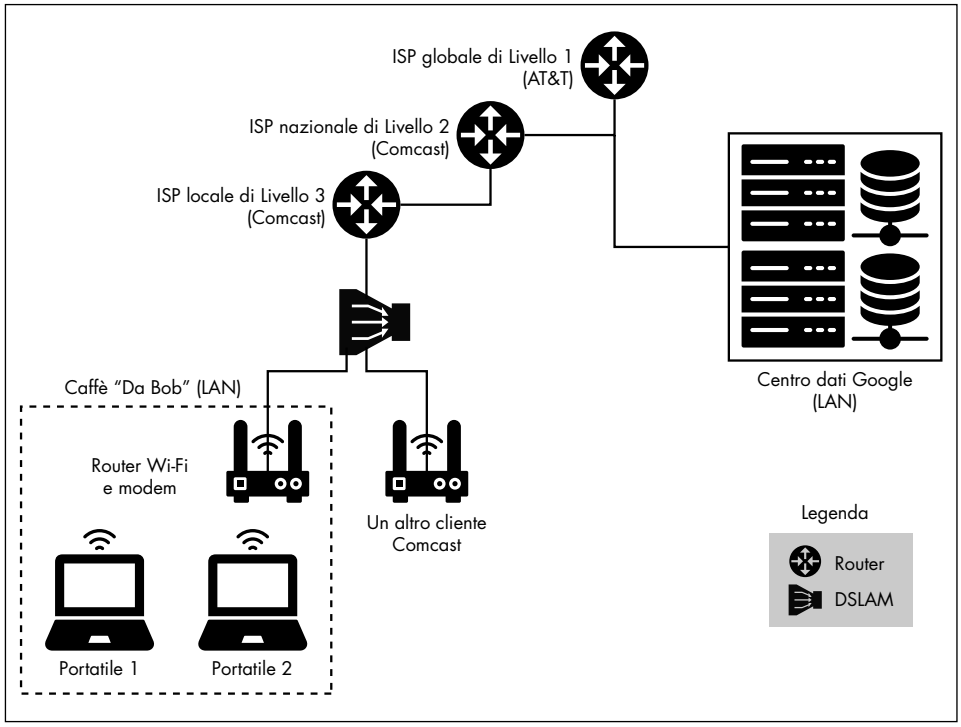

Figura 2.2 Una vista semplificata della gerarchia di rete.

La Figura 2.2 mostra anche un DSLAM (*Digital Subscriber Line Access Multiplexer*). Un DSLAM consente di inviare i segnali del traffico Internet su cavi originariamente previsti per la televisione via cavo. Il DSLAM distingue tra i segnali Internet e i segnali televisivi, motivo per cui è possibile collegare alla stessa presa sia il televisore sia il router.
Usiamo l'esempio del coffee shop per seguire un pacchetto mentre attraversa la gerarchia di rete. Immaginate di essere in un bar di San Francisco e di accedere alla seguente pagina web: `http://www.cs.virginia.edu`. Questa pagina web è ospitata su un server web con

l'indirizzo IP `128.143.67.11`. Nella prima parte del suo viaggio, la richiesta web attraversa la scheda NIC del vostro portatile, che poi la invia al router WiFi nel bar. Il router invia quindi la richiesta web al DSLAM, che inoltra la richiesta a un router di proprietà di un ISP (*Internet Service Provider*), come Comcast. I router Comcast confrontano l'indirizzo IP con un elenco di prefissi, fino a trovare una corrispondenza. Per esempio, il router Comcast potrebbe trovare una corrispondenza per il prefisso `128.xxx.xxx.xxx`, che indica la sua connessione a quella sezione della gerarchia. A mano a mano che la richiesta scende nella gerarchia, le corrispondenze diventeranno più specifiche. Per esempio, l'indirizzo cercherà corrispondenze con `128.143.xxx.xxx` e poi con `128.143.67.xxx`. Una volta che il pacchetto raggiunge il livello più basso della gerarchia, dove non ci sono più router, il router utilizza l'indirizzo MAC nel pacchetto per determinare la destinazione finale della richiesta. Il livello più basso della gerarchia è una rete locale, LAN (*Local Area Network*), dove tutte le macchine sono connesse tramite un singolo router.
Ora che abbiamo una panoramica generale della struttura di Internet, possiamo parlare degli attacchi che hanno luogo al livello più basso della gerarchia.

Tabelle ARP

Abbiamo stabilito che, dopo che un pacchetto ha raggiunto la sua LAN designata, la rete utilizza l'indirizzo MAC del pacchetto per determinare la sua destinazione finale. Ma come fa il router a conoscere l'indirizzo MAC della macchina avente l'indirizzo IP `128.143.67.11`? È qui che diventa utile l'ARP, il "protocollo per la risoluzione dell'indirizzo". In base all'ARP, il router invia un messaggio, chiamato *query ARP*, a tutte le macchine della rete, chiedendo alla macchina con l'indirizzo IP `128.143.67.11` di rispondere con una *risposta ARP* contenente il proprio indirizzo MAC. Il router memorizzerà così questa mappatura tra l'indirizzo IP e il MAC in un'apposita tabella, la *tabella ARP*. Memorizzando queste informazioni nella tabella ARP, il router riduce la necessità di emettere query ARP nel prossimo futuro.

La versione breve

L'indirizzo MAC identifica chi siete; l'indirizzo IP identifica dove vi trovate; la tabella ARP gestisce la mappatura tra chi siete e dove vi trovate sulla rete. In un attacco ARP Spoofing, fingiamo di essere qualcun altro.

Attacchi ARP Spoofing

Un attacco ARP Spoofing comporta due fasi. Durante la prima fase, l'hacker invia alla vittima una falsa risposta ARP, affermando che il suo indirizzo MAC è l'indirizzo IP del router. Ciò consente all'hacker di indurre la vittima a credere che la macchina dell'hacker sia il suo router. Durante la seconda fase, la vittima accetta il falso pacchetto ARP inviato dall'hacker e aggiorna la mappa nella sua tabella ARP, per riflettere il fatto che l'indirizzo MAC dell'hacker ora corrisponde all'indirizzo IP del router. Ciò significa che il traffico Internet della vittima verrà inviato alla macchina dell'hacker invece che al

router. La macchina dell'hacker potrà poi inoltrare queste stesse informazioni al router, dopo averle ispezionate.
Se l'hacker vuole intercettare anche il traffico Internet destinato alla vittima, deve ingannare anche il router, affinché gli invii il traffico della vittima. Pertanto, l'hacker deve creare un falso pacchetto ARP che indichi che l'indirizzo IP della vittima corrisponde all'indirizzo MAC dell'hacker. Ciò consente all'hacker di intercettare e ispezionare il traffico in ingresso da Internet e poi inoltrarlo alla vittima.
Possiamo spiegare le idee di base di un attacco ARP Spoofing con un semplice schema, illustrato nella Figura 2.3. Qui, Jane (l'hacker) inganna Alice (la vittima) facendo inviare a sé la posta di Alice.

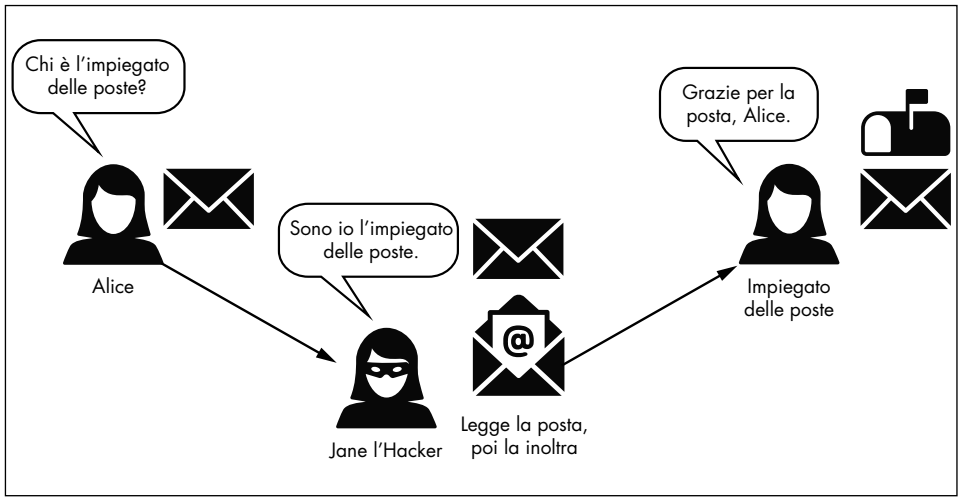

Figura 2.3 Un esempio di attacco ARP Spoofing che coinvolge un impiegato delle poste.

L'attacco ARP Spoofing è un esempio di attacco *man-in-the-middle*, perché l'hacker si frappone tra la vittima e il router.

Esecuzione di un attacco ARP Spoofing

Eseguiamo un attacco ARP Spoofing. Innanzitutto, dovete assicurarvi di aver avviato le macchine virtuali pfSense, Kali e Metasploitable prima di poter attuare questo attacco. Per sapere come farlo, consultate il Capitolo 1. Ora installiamo gli strumenti di cui avremo bisogno per eseguire l'attacco ARP Spoofing. Aprite un terminale sulla macchina virtuale Kali Linux e installate lo strumento `dsniff`. La password di default per la macchina virtuale Kali Linux è "`kali`". Iniziate eseguendo **`sudo -i`** per diventare utenti root. Dovrete anche aggiornare il gestore di pacchetti `apt-get`, eseguendo il comando **`apt-get update`**.

```
kali@kali:~$ sudo -i
kali@kali:~$ apt-get update
kali@kali:~$ apt-get install dsniff
```

Lo strumento dsniff contiene diversi strumenti utili per intercettare il traffico di rete, come arpspoof, uno strumento che esegue un attacco ARP Spoofing.
Dobbiamo scoprire gli indirizzi IP delle altre macchine della rete per *falsificarle* (ovvero per fingere di essere loro). Eseguite lo strumento netdiscover utilizzando il seguente comando:

```
kali@kali:~$ sudo netdiscover
```

netdiscover funziona sottoponendo a scansione la rete tramite query ARP. Emette query ARP per tutti i possibili indirizzi IP della sottorete e, quando una macchina sulla rete risponde, registra e visualizza l'indirizzo MAC e l'indirizzo IP della macchina. Lo strumento netdiscover deduce dall'indirizzo MAC anche il produttore della scheda NIC. Poiché tutti gli indirizzi MAC devono essere univoci, un ufficio centrale presso dell'IEEE (*Institute of Electrical and Electronics Engineers*) assegna ai produttori un intervallo di indirizzi MAC, per garantire l'univocità.
La vostra scansione dovrebbe rilevare due macchine della rete e generare l'output mostrato di seguito:

```
IP              At MAC Address      Count     Len  MAC Vendor / Hostname
-----------------------------------------------------------------------------
192.168.100.1   08:00:27:3b:8f:ed       1      60  PCS Systemtechnik GmbH
192.168.100.101 08:00:27:fe:31:e6       1      60  PCS Systemtechnik GmbH
```

Gli indirizzi IP effettivi restituiti variano a seconda della configurazione. La macchina con l'indirizzo IP più basso è normalmente il router della LAN. Per il resto di questo capitolo faremo riferimento a questo indirizzo IP come <*IP_ROUTER*>. Il secondo indirizzo IP appartiene alla macchina virtuale Metasploitable (la nostra vittima), che chiameremo <*IP_VITTIMA*>. Una volta individuate entrambe le macchine, terminate la scansione premendo CTRL-C.
Successivamente, dovrete consentire alla macchina Kali Linux di inoltrare i pacchetti per conto di altre macchine, abilitando l'inoltro IP (*IP forward*). Assicuratevi di essere utenti root su Kali Linux, quindi abilitate l'inoltro IP impostando il suo flag:

```
kali@kali:~$ echo 1 > /proc/sys/net/ipv4/ip_forward
```

Ora che avete abilitato l'inoltro IP, dovrete indurre la vittima a credere che voi siate il router. Potete farlo emettendo false risposte ARP, affermando che il vostro indirizzo MAC è mappato all'indirizzo IP del router. La Figura 2.4 mostra un esempio di questa fase dell'attacco.
Potete anche generare più risposte ARP false, lanciando il seguente comando:

```
arpspoof -i eth0 -t <IP_VITTIMA> <IP_ROUTER>
```

Il flag -t specifica l'obiettivo e il flag -i rappresenta l'interfaccia. La scheda NIC supporta diversi modi di connessione alla rete. Per esempio, WLAN rappresenta una LAN wireless (una connessione WiFi) ed eth0 rappresenta una connessione Ethernet. In questo ambiente virtuale, le macchine sono virtualmente connesse tramite Ethernet, quindi utilizzerete eth0 per la vostra interfaccia. Nell'ambiente del coffee shop, l'interfaccia sarebbe impostata su wlan.

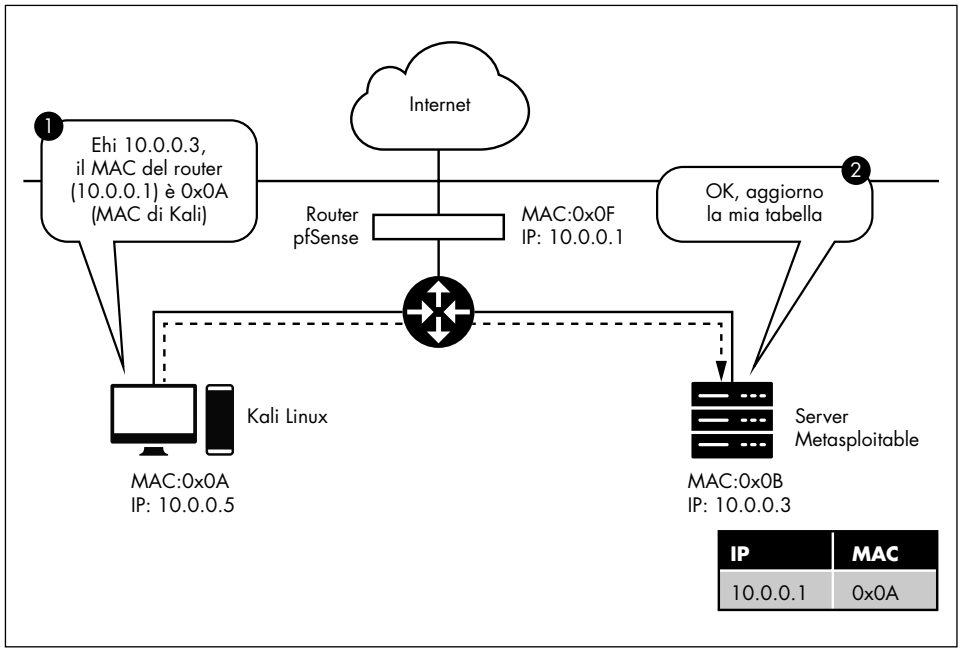

Figura 2.4 La prima fase di un attacco ARP Spoofing.

Il frammento di codice seguente mostra il risultato dell'esecuzione di `arpspoof`. Dovrete generare più risposte ARP false per assicurarvi che la tabella sia sempre aggiornata con le informazioni errate. Ma lo strumento genererà automaticamente più pacchetti per voi, quindi dovete eseguirlo una sola volta.

```
kali@kali:~$ sudo arpspoof -i eth0 -t 192.168.100.101 192.168.100.1
[sudo] password for kali:
8:0:27:1f:30:76 8:0:27:fe:31:e6 0806 42: arp reply 192.168.100.1 is-at 8:0:27:1f:30:76 ❶
8:0:27:1f:30:76 8:0:27:fe:31:e6 0806 42: arp reply 192.168.100.1 is-at 8:0:27:1f:30:76
```

Esaminiamo l'output del comando, prestando particolare attenzione alla prima riga, ❶. Questa riga rappresenta un riepilogo delle informazioni contenute nel pacchetto appena inviato. Il sommario è composto da cinque parti.

1. `8:0:27:1f:30:76` è l'indirizzo MAC della macchina Kali Linux (l'hacker), che ha creato il pacchetto.
2. `8:0:27:fe:31:e6` è l'indirizzo MAC della macchina (vittima) che riceverà il pacchetto.
3. `0806` è un campo del tipo, che indica che un pacchetto ARP è contenuto all'interno del frame Ethernet trasmesso.
4. `42` rappresenta il numero totale di byte associati al frame Ethernet.
5. La sezione rimanente, `arp reply 192.168.100.1 is-at 8:0:27:1f:30:76`, è un riepilogo della risposta ARP, che afferma, falsamente, che l'indirizzo IP del router (`192.168.100.1`) è associato all'indirizzo MAC della macchina Kali Linux (`8:0:27:1f:30:76`).

Dovete indurre anche il router a credere che voi siate la vittima, in modo da poter intercettare anche il traffico Internet in ingresso e diretto alla vittima. Aprite un nuovo terminale e lanciate il comando che segue. Notate che *<IP_ROUTER>* e *<IP_VITTIMA>* ora sono invertiti. Questo perché ora state generando dei pacchetti per indurre il router a credere che voi siate la vittima:

```
kali@kali:~$ arpspoof -i eth0 -t <IP_ROUTER> <IP_VITTIMA>
```

Ora che avete ingannato la vittima e il router, che cosa potete fare con i pacchetti intercettati? Ispezioniamo i pacchetti che abbiamo intercettato ed estraiamo i loro URL. Questo ci consentirà di generare un elenco dei siti web visitati dalla vittima. Estraete gli URL lanciando il seguente comando in un nuovo terminale:

```
kali@kali:~$ urlsnarf -i eth0
```

Potete anche generare del traffico Internet sulla macchina della vittima. Fate login alla macchina virtuale Metasploitable utilizzando `msfadmin` sia per il nome utente sia per la password, quindi immettete il seguente comando per generare una richiesta web a `google.com`:

```
msfadmin@metasploitable:~$ wget http://www.google.com
```

La Figura 2.5 presenta una panoramica di ciò che accade in questa fase.

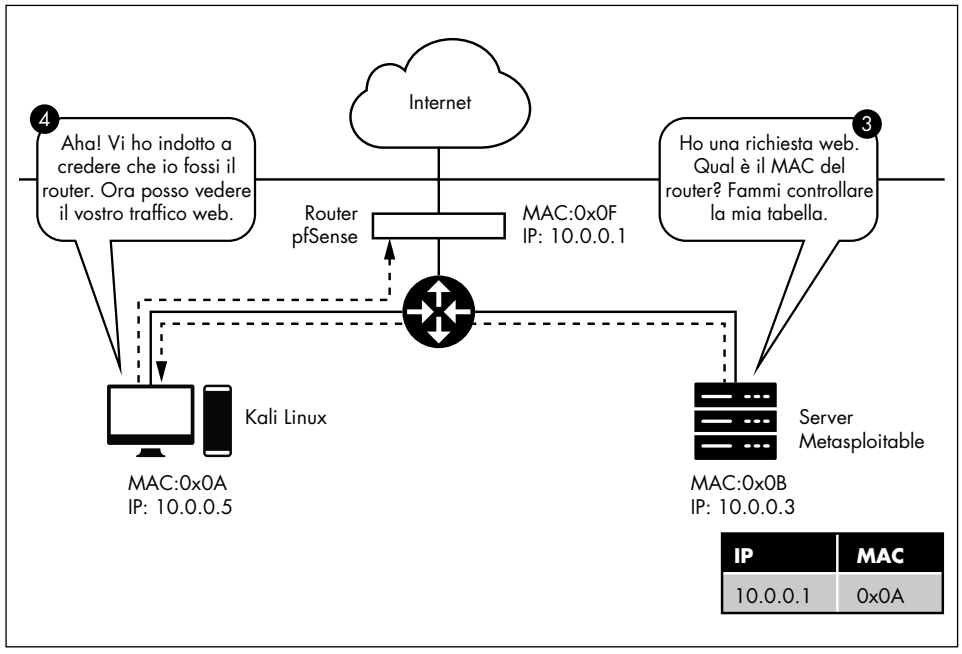

Figura 2.5 La seconda fase dell'attacco ARP Spoofing, in cui la vittima utilizza la tabella ARP manipolata per indirizzare i pacchetti.

Se avete svolto tutto correttamente, l'URL associato alla richiesta web apparirà nel terminale dopo un paio di minuti. Siate pazienti: ci vuole tempo per analizzare i pacchetti:

```
kali@kali:~$ sudo urlsnarf -i eth0
urlsnarf: listening on eth0 [tcp port 80 or port 8080 or port 3128]
192.168.100.101 - - "GET http://www.google.com/HTTP/1.0"
```

Provate a esaminare questo output. Anche se qui mostriamo solo l'URL, la macchina dell'hacker sta catturando tutti i pacchetti che la vittima invia e riceve da Internet. Ciò significa che l'hacker può vedere qualsiasi informazione non crittografata che la vittima invia attraverso la Rete. Significa anche che un malintenzionato può modificare i pacchetti per iniettare codice dannoso sulla macchina.
Una volta che avete finito di eseguire le vostre azioni dannose, non lasciate le tabelle ARP nello stato manipolato. Una volta che l'hacker ha lasciato il bar, la vittima non sarà più in grado di connettersi a Internet, e così sospetterà qualcosa. È necessario ripristinare le tabelle ARP alle loro configurazioni originali prima di terminare l'attacco. Per fortuna, `arpspoof` lo fa per noi. Concludete l'attacco premendo CTRL-C in entrambi i terminali che eseguono `arpspoof`.

Proteggersi dall'ARP Spoofing

Sebbene sia difficile prevenire un attacco ARP Spoofing, la crittografia del traffico Internet aiuta a proteggere le informazioni, che non potranno essere sottratte o modificate. Tutto il traffico inviato tramite una connessione HTTPS è crittografato. Tuttavia, è noioso controllare manualmente che ogni URL che visitate utilizzi HTTPS, quindi l'Electronic Frontier Foundation (*eff.org*) ha creato un'estensione dei browser web (per Chrome, Edge, Firefox e Opera) chiamata *HTTPS Everywhere* che garantisce che tutto il vostro traffico web attraversi una connessione HTTPS. L'installazione di questo plug-in è un ottimo modo per proteggersi.

Rilevamento di un attacco ARP Spoofing

In questo paragrafo, scriveremo un programma Python per rilevare euristicamente un attacco ARP Spoofing. Costruiremo la nostra tabella ARP usando un dizionario e poi controlleremo se il pacchetto che riceviamo ha cambiato una voce. Supporremo che qualsiasi pacchetto che modifica lo stato della nostra tabella sia dannoso.
Inizieremo selezionando una libreria in grado di intercettare e analizzare i pacchetti che attraversano la nostra scheda NIC. Scapy è un pacchetto Python che ci consente di leggere e inviare pacchetti di dati. Prima di poter utilizzare Scapy, dovrete installarlo con `pip3`. Usate i seguenti comandi per ottenere sia `pip3` sia Scapy:

```
kali@kali:~$ sudo apt-get install python3-pip
kali@kali:~$ pip3 install --pre scapy[basic]
```

Una volta installato Scapy, potrete importare la libreria `sniff`, che ci consente di catturare e ispezionare i pacchetti che attraversano la nostra scheda NIC. Copiate e incollate il seguente programma Python (`arpDetector.py`) in Mousepad o nell'editor di codice di vostra scelta. Per avviare Mousepad, eseguite **`mousepad &`**.

```
from scapy.all import sniff
IP_MAC_Map = {}

def processPacket(packet):
    src_IP = packet['ARP'].psrc
    src_MAC = packet['Ether'].src
    if src_MAC in IP_MAC_Map.keys():
        if IP_MAC_Map[src_MAC] != src_IP :
            try:
                old_IP = IP_MAC_Map[src_MAC]
            except:
                old_IP = "unknown"
            message = ("\n Possible ARP attack detected \n "
                       + "It is possible that the machine with IP address \n "
                       + str(old_IP) + " is pretending to be " + str(src_IP)
                       +"\n ")
            return message
        else:
            IP_MAC_Map[src_MAC] = src_IP

sniff(count=0, filter="arp", store = 0, prn = processPacket)  ❶
```

La funzione `sniff()` ❶ della libreria Scapy accetta diversi parametri opzionali. In questa implementazione, usiamo il parametro `count` per indicare il numero di pacchetti da "annusare". Un valore di `count` pari a `0` significa che la libreria deve controllare tutti i pacchetti. Usiamo anche il parametro `filter`, che specifica il tipo di pacchetto da catturare. Poiché siamo interessati solo ai pacchetti ARP, specifichiamo un valore di filtro `"arp"`. Il parametro `store` indica il numero di pacchetti da memorizzare. Impostiamo il parametro a `0`, perché non vogliamo sprecare memoria memorizzando i pacchetti. Infine, il parametro `prn` punta alla funzione da richiamare ogni volta che viene ricevuto un pacchetto. Prende un singolo parametro, che rappresenta il pacchetto ricevuto, come input.

```
kali@kali:~$ sudo python3 arpDetector.py
```

Mentre il programma è in esecuzione, aprite un altro terminale Kali ed eseguite un attacco ARP Spoofing.
Poi, concludete l'attacco premendo CTRL-C. Ciò farà in modo che `arpspoof` emetta dei pacchetti che ripristinano la tabella ARP. Quando il vostro programma Python rileva questi pacchetti, vedrete un messaggio simile al seguente:

```
Possible ARP attack detected
It is possible that the machine with IP address
192.168.0.67 is pretending to be 192.168.48.67
```

Esercizi

Approfondite la vostra comprensione dello spoofing e dell'inoltro ARP provando i seguenti esercizi, proposti in ordine di difficoltà crescente. Il primo esercizio richiede l'esecuzione di un solo comando; il secondo è più impegnativo, perché richiede di scrivere un programma Python e approfondire la comprensione della libreria Scapy. L'ultimo esercizio vi chiede di applicare le nozioni fondamentali apprese in questo capitolo a un nuovo attacco.

Ispezionare le tabelle ARP

Ispezionate le tabelle ARP della macchina virtuale Metasploitable lanciando questo comando:

```
msfadmin@metasploitable:~$ sudo arp -a
```

Confrontate lo stato delle tabelle ARP sul server Metasploitable prima e dopo l'attacco ARP Spoofing. Notate differenze? In caso affermativo, quali voci sono state modificate?

Implementare uno spoofer ARP in Python

In questo capitolo, abbiamo visto come eseguire un attacco ARP Spoofing. Per questo esercizio, scriverete un programma Python che vi permetta di eseguire un attacco ARP Spoofing con un singolo comando, mostrato di seguito:

```
kali@kali:~$ sudo python3 arpSpoof.py <IP_VITTIMA> <IP_ROUTER >
```

Per fare ciò, dovrete scrivere un programma che esegua i passi trattati in questo capitolo. Il vostro programma dovrebbe generare pacchetti ARP contraffatti e inviarli sia alla vittima sia al router. Una volta completato l'attacco, il programma dovrebbe ripristinare le tabelle ARP al loro stato originale. Scrivete il vostro programma (`arpSpoof.py`) in Python e usate la libreria Scapy per costruire e inviare i pacchetti. Di seguito trovate del "codice scheletro":

```
from scapy.all import *
import sys

def arp_spoof(dest_ip, dest_mac, source_ip):  ❶
    pass

def arp_restore(dest_ip, dest_mac, source_ip, source_mac):  ❷
    packet=ARP(op="is-at", hwsrc=source_mac,  ❸
               psrc= source_ip, hwdst= dest_mac, pdst= dest_ip)
    send(packet, verbose=False)  ❹

def main():
    victim_ip= sys.argv[1]
    router_ip= sys.argv[2]
```

```
    victim_mac = getmacbyip(victim_ip)
    router_mac = getmacbyip(router_ip)

    try:
        print("Sending spoofed ARP packets")
        while True:
            arp_spoof(victim_ip, victim_mac, router_ip)
            arp_spoof(router_ip, router_mac, victim_ip)
    except KeyboardInterrupt:
        print("Restoring ARP Tables")
        arp_restore(router_ip, router_mac, victim_ip, victim_mac)
        arp_restore(victim_ip, victim_mac, router_ip, router_mac)
        quit()

main()
```

Implementate la funzione `arp_spoof()` ❶. Questa funzione dovrebbe essere molto simile ad `arp_restore()` ❷, che ripristina le tabelle ARP al loro stato originale. Potete usare `arp_restore()` come guida. All'interno di tale funzione, creiamo un nuovo pacchetto ARP. La funzione `ARP()` ❸ accetta diverse opzioni (`op`). L'opzione `"is-at"` rappresenta una risposta ARP e l'opzione `"who-has"` rappresenta una richiesta ARP. Potreste anche vedere queste opzioni elencate rispettivamente come i numeri 2 e 1. Infine, inviamo il pacchetto che abbiamo creato ❹.

MAC Flooding

La memoria CAM (*Content Addressable Memory*) è la memoria hardware utilizzata sia nei router sia negli switch. Negli switch, questa memoria mappa gli indirizzi MAC alle relative porte. Pertanto, la memoria CAM può memorizzare solo un numero limitato di voci. Se la memoria CAM dello switch è piena, trasmetterà un messaggio su tutte le porte. Gli hacker possono forzare questo comportamento inviando allo switch dei pacchetti aventi indirizzi MAC casuali. Scrivete un programma Scapy che svolga questo attacco.