Relazione sull'attività di cattura e analisi del traffico DNS utilizzando Wireshark su Linux

Parte 1: Cattura del traffico DNS

Step 1: Installazione di Wireshark

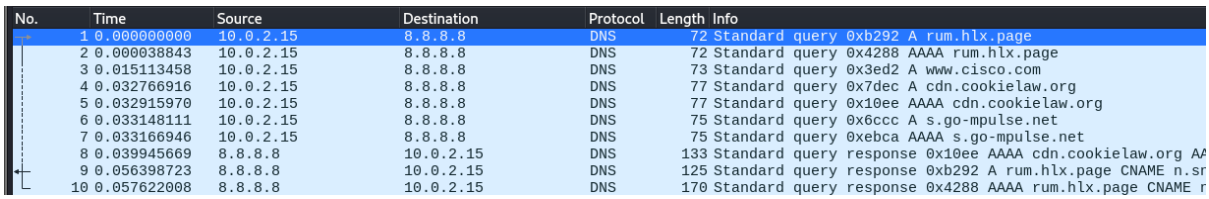
1. Per iniziare, è stato necessario installare Wireshark, uno strumento di analisi del traffico di rete.

Step 2: Cattura del traffico DNS

Prima di iniziare la cattura dei pacchetti, si è proceduto a cancellare la cache DNS per evitare che i risultati fossero influenzati da risoluzioni DNS precedenti. Su Kali Linux, il comando per verificare e cancellare la cache varia in base all'utility utilizzata per la gestione della cache DNS è stato usato:

```
sudo reboot
```

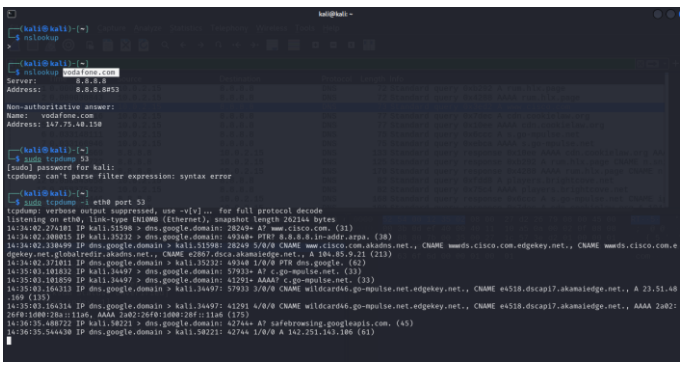
1. Avviato Wireshark, è stata selezionata l'interfaccia di rete attiva *eth0* per catturare il traffico in tempo reale.
2. Si è aperto un terminale per avviare la risoluzione dei nomi DNS tramite il comando **nslookup + il provider (nslookup vodafone.com)**.:
3. Dopo aver completato la risoluzione DNS, è stata interrotta la cattura dei pacchetti in Wireshark cliccando su "Stop capturing packets".



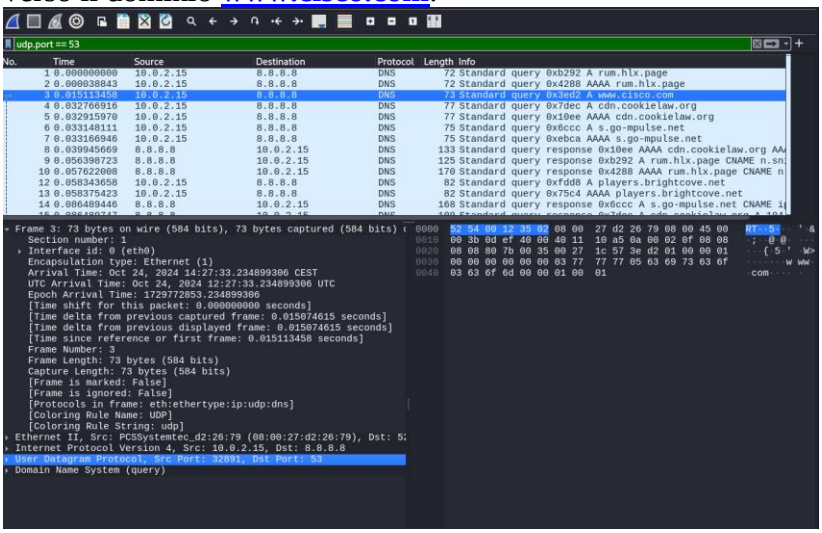
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.0.2.15	8.8.8.8	DNS	72	Standard query 0xb292 A rum.hlx.page
2	0.000038843	10.0.2.15	8.8.8.8	DNS	72	Standard query 0x4288 AAAA rum.hlx.page
3	0.015113458	10.0.2.15	8.8.8.8	DNS	73	Standard query 0x3ed2 A www.cisco.com
4	0.032766916	10.0.2.15	8.8.8.8	DNS	77	Standard query 0x7dec A cdn.cookielaw.org
5	0.032915970	10.0.2.15	8.8.8.8	DNS	77	Standard query 0x10ee AAAA cdn.cookielaw.org
6	0.033148111	10.0.2.15	8.8.8.8	DNS	75	Standard query 0x6ccc A s.go-mpulse.net
7	0.033166946	10.0.2.15	8.8.8.8	DNS	75	Standard query 0xebca AAAA s.go-mpulse.net
8	0.039945669	8.8.8.8	10.0.2.15	DNS	133	Standard query response 0x10ee AAAA cdn.cookielaw.org AA
9	0.056398723	8.8.8.8	10.0.2.15	DNS	125	Standard query response 0xb292 A rum.hlx.page CNAME n.sr
10	0.057622008	8.8.8.8	10.0.2.15	DNS	170	Standard query response 0x4288 AAAA rum.hlx.page CNAME r

Parte 2: Esplorazione del traffico di query DNS

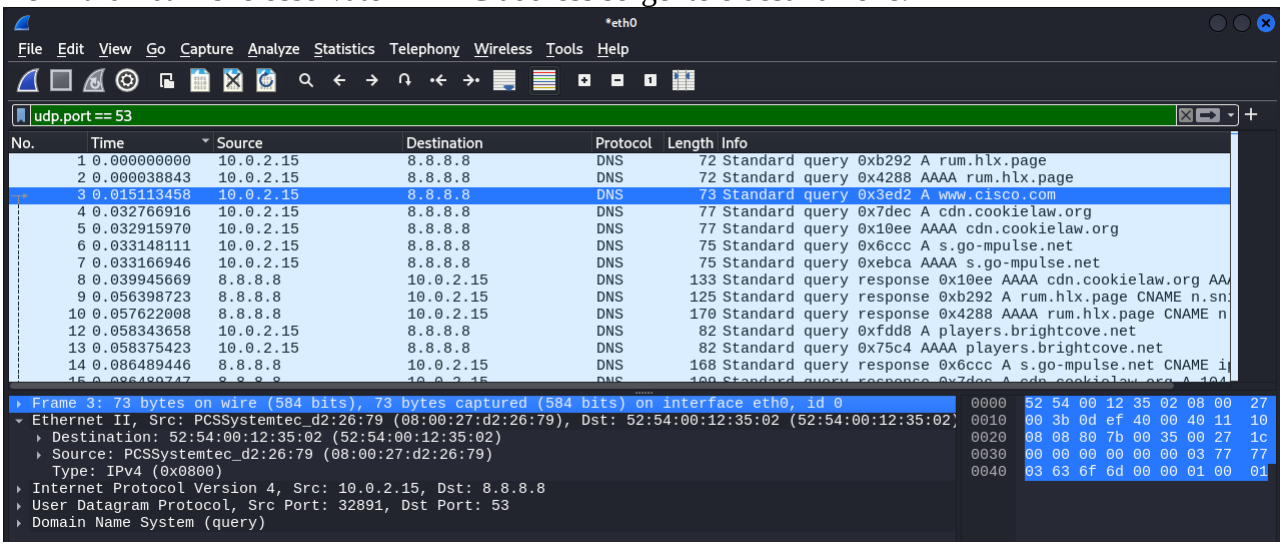
1. Una volta interrotta la cattura, si è proceduto ad analizzare i pacchetti DNS acquisiti. Per facilitare l'analisi, è stato applicato un filtro per visualizzare solo i pacchetti DNS, inserendo il seguente filtro nella barra di ricerca: `udp.port == 53`



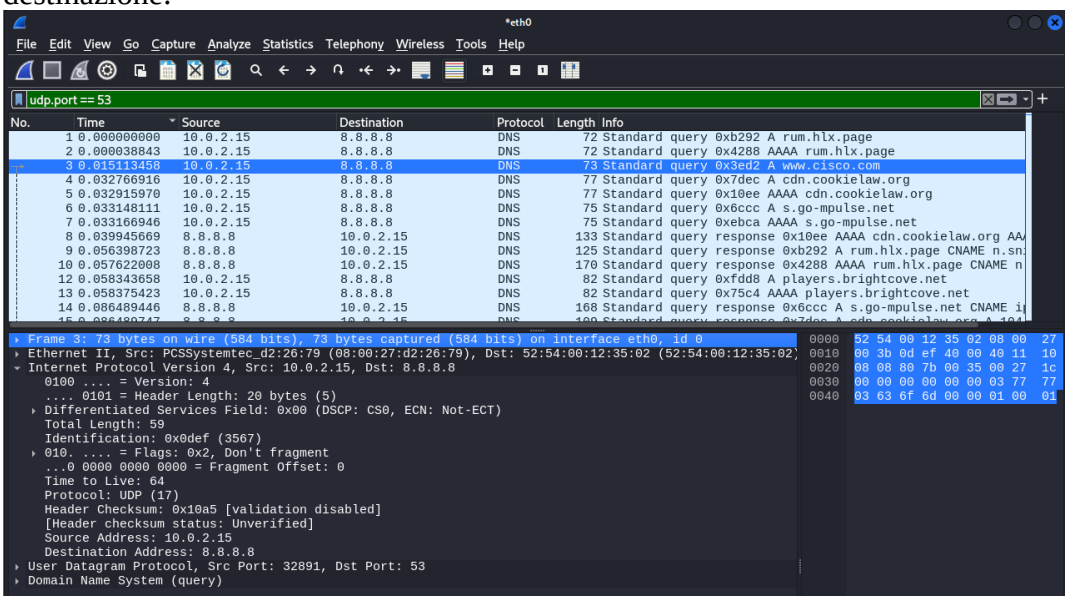
2. Nel *Packet List*, è stato selezionato un pacchetto DNS contenente una **Standard query** verso il dominio **www.cisco.com**.



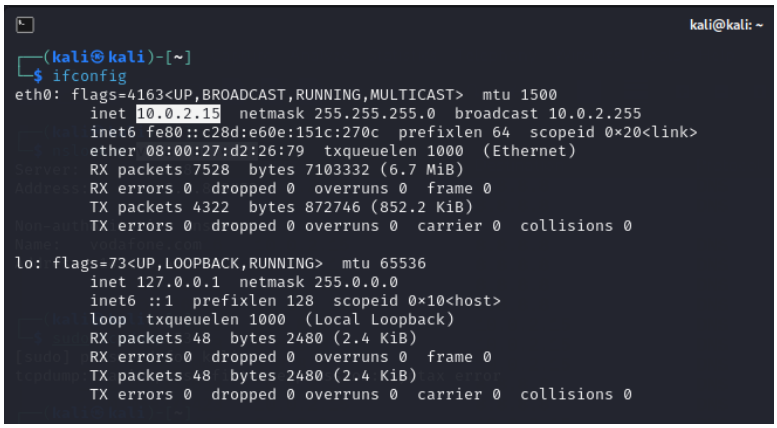
3. Nell'*Ethernet II* si è osservato il MAC address sorgente e destinazione.



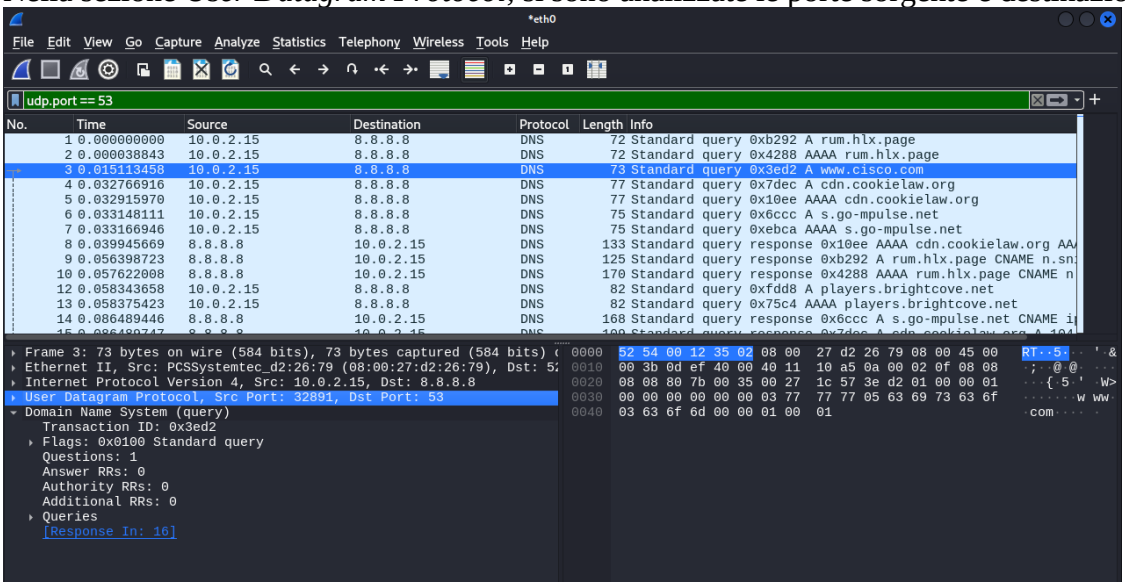
4. Nella sezione *Internet Protocol Version 4*, sono stati esaminati gli indirizzi IPv4 sorgente e destinazione.



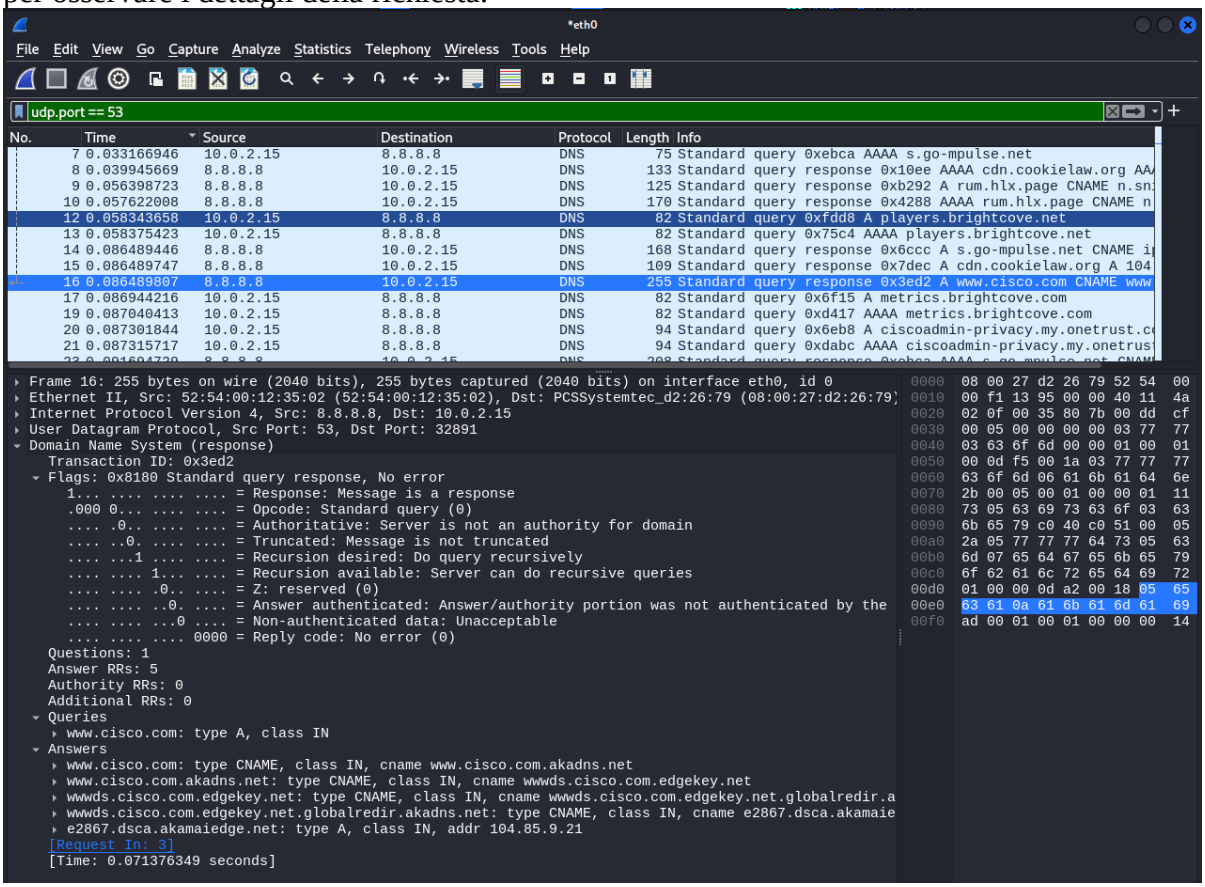
Successivamente abbiamo messo l'**ifconfig** in un terminale.



5. Nella sezione *User Datagram Protocol*, si sono analizzate le porte sorgente e destinazione.



6. Infine, nella sezione *Domain Name System (query)*, si sono espansi i campi **Flags** e **Queries** per osservare i dettagli della richiesta.



Parte 3: Esplorazione del traffico di risposta DNS

1. Si è passati a osservare i pacchetti DNS di risposta, selezionando un pacchetto che conteneva una **Standard query response** per il dominio **www.cisco.com**.
2. Nella sezione *Domain Name System (response)*, si sono espansi i campi **Flags**, **Queries** e **Answers**.
3. Nei dettagli della risposta, si è potuto osservare il **CNAME** e gli **A record** che forniscono l'indirizzo IP associato al dominio richiesto.

Conclusioni

Questa attività ha permesso di acquisire una comprensione approfondita del traffico DNS e del modo in cui i pacchetti vengono scambiati tra un client e un server DNS. L'uso di Wireshark su un ambiente Linux ha facilitato la cattura e l'analisi dettagliata delle richieste e risposte DNS. In particolare, l'analisi del contenuto dei pacchetti ha fornito informazioni preziose sugli indirizzi IP, MAC e sui record DNS (A e CNAME), cruciali per comprendere le dinamiche di risoluzione dei nomi di dominio su una rete.