| Kali | Meta | Scan |
|---|---|---|
| 192.168.50.100 | 192.168.50.101 | TPC |
| 192.168.50.100 | 192.168.50.101 | SYN |
| 192.168.50.100 | 192.168.50.101 | Switch -A |

# GELOSA MATTEO ESERCIZIO W9D1

## SCANSIONE TCP

Scan porta 80 : Prendendo in considerazione lo scan sulla porta 80(porta aperta) ho notato che avviene uno scambio di pacchetti 4 volte.

1. La prima richiesta viene fatta da Kali , contiene un pacchetto SYN ed il numero di sequenza  per avviare la connessione con Meta .
2. La seconda richiesta viene fatta da Meta, più che una richiesta è una risposta, essendo la porta aperta e quindi pronta ad accettare la richiesta di connessione avremo come risposta un SYN ACK con il numero di sequenza iniziale.
3. La terza sessione viene iniziata da Kali mandando un pacchetto ACK, contiene il numero di sequenza successivo per poter procedere.
4. L'ultima sessione completa il 3-Handshake , le tue macchine possono comunicare tra loro .

Questa scansione è più lenta ed "invasiva" perchè per ogni richiesta il client cerca di creare una connessione con il destinatario, a differenza della scansione SYN.
Infine ho notato che quando una porta risulta chiusa Whireshark risponde con un RST che appunto corrisponde alla mancata connessione.

## SCANSIONE SYN

Nella scansione SYN , prendendo in considerazione la porta 80 ho notato che Kali inviaun solo un pacchetto , non completando tutto l'intero ciclo di Handshake.
Se la porta risulta aperta meta risponde con SYN ACK.
Questo metodo risulta molto più veloce appunto perché non avviene uno scambio di 4 pacchetti come nel metodo TCP.

# Scansione -A

In questa scansione ho notato come venga fatta una scansione dettagliata su tutte le porte aperte di meta.
Prendendo sempre in considerazione la porta 80 posso notare ovviamente che la porta è aperta, usa un protocollo http, quindi non sicuro.
Posso vedere che su questa porta è attivo un server Apache 2.2.8 (Ubuntu).