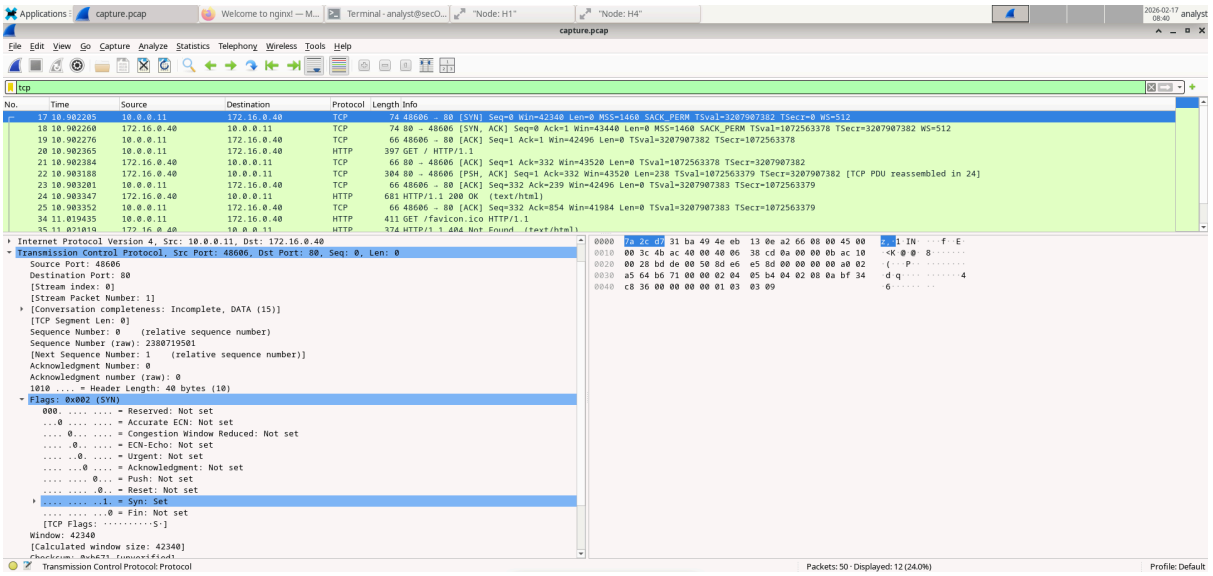
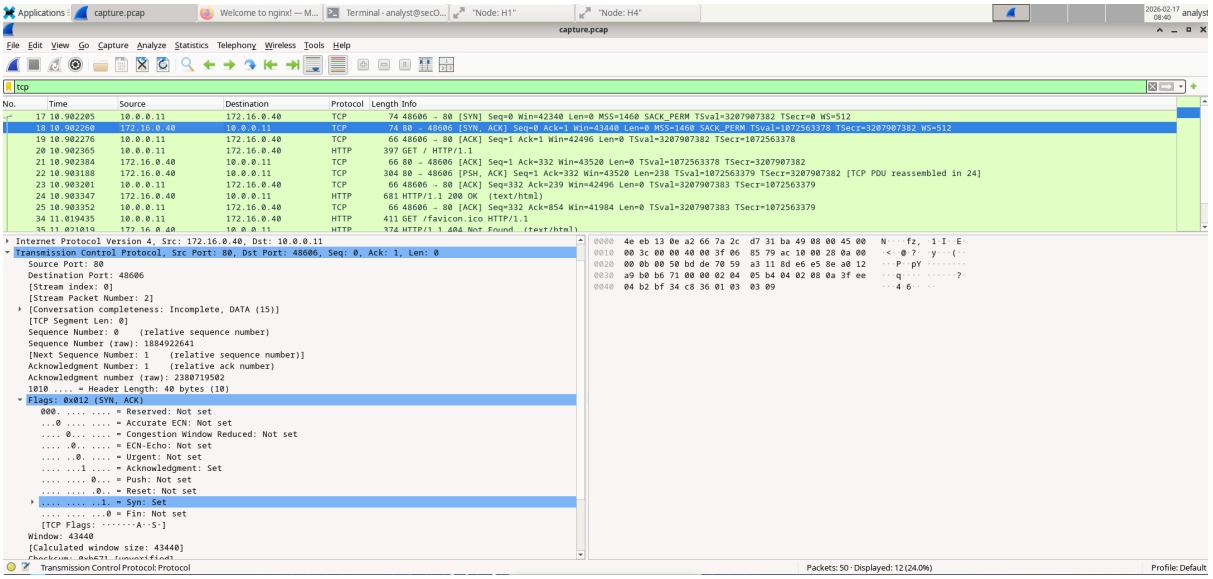
Analisi Handshake TCP a 3 Vie

PACCHETTO 1 (Frame 17 - SYN)



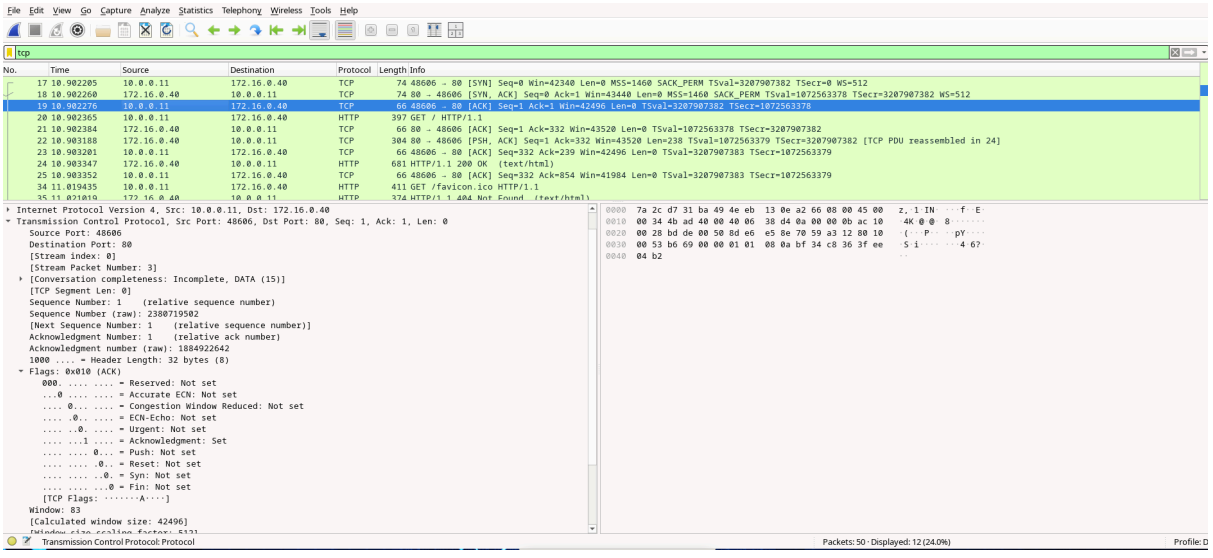
- Qual è il numero di porta TCP di origine?
  - 48606
- Come classificheresti la porta di origine?
  - Porta dinamica/effimera (range 49152-65535, usata dai client per connessioni temporanee)
- Qual è il numero di porta TCP di destinazione?
  - 80
- Come classificheresti la porta di destinazione?
  - Porta well-known (porta standard HTTP per server web)
- Quale flag è impostato?
  - SYN (Synchronize - richiesta di sincronizzazione per iniziare la connessione)
- A quale valore è impostato il numero di sequenza relativo?
  - 0 (numero di sequenza iniziale relativo)

PACCHETTO 2 (Frame 18 - SYN, ACK)



- Quali sono i valori delle porte di origine e destinazione?
    - Porta di origine: 80 (server web)
    - Porta di destinazione: 48606 (client - le porte sono invertite rispetto al pacchetto 1)
- Quali flag sono impostati?
    - SYN e ACK (Synchronize + Acknowledgment - il server accetta la connessione e invia il suo numero di sequenza)
- A quali valori sono impostati i numeri relativi di sequenza e acknowledgment?
    - Sequence number (relativo): 0
    - Acknowledgment number (relativo): 1 (conferma la ricezione del SYN del client)

PACCHETTO 3 (Frame 19 - ACK)



● Quale flag è impostato?

- ACK (Acknowledgment - il client conferma la ricezione del SYN-ACK del server)

● A quali valori sono impostati i numeri relativi di sequenza e acknowledgment?

- Sequence number (relativo): 1
- Acknowledgment number (relativo): 1

Risultato: La connessione TCP è ora stabilita e la comunicazione tra client (10.0.0.11) e server web (172.16.0.40) può iniziare.

Tre filtri utili per un amministratore di rete:

1. `ip.addr == 192.168.1.100` - Filtra tutto il traffico da/verso un IP specifico per diagnosticare problemi su un host
2. `tcp.flags.reset == 1` - Identifica connessioni TCP resettate, utile per trovare problemi di connessione o attacchi
3. `http.request.method == "POST"` - Monitora richieste POST HTTP per analizzare upload di dati o possibili esfiltrazione

Altri utilizzi di Wireshark in una rete di produzione:

- Troubleshooting di problemi di rete: latenza, packet loss, retransmissioni
- Analisi delle prestazioni: individuare bottleneck e ottimizzare il traffico
- Sicurezza: rilevare attacchi (port scanning, DoS, man-in-the-middle)
- Verifica configurazioni: controllo VLAN, QoS, routing
- Analisi protocolli applicativi: verificare corretto funzionamento di protocolli specifici
- Documentazione: creare baseline del traffico normale per confronti futuri

---