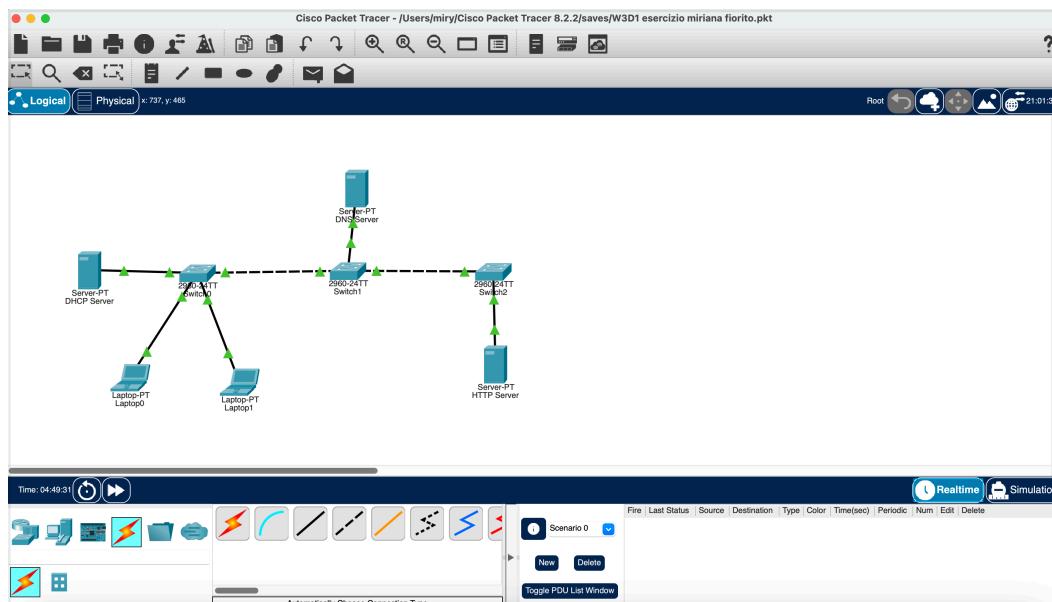


Per questo esercizio con Packet Tracer, abbiamo aggiunto ad una rete connessa tre servizi applicativi fondamentali: DHCP, DNS e HTTP.

Obiettivo: configurare due client affinché ricevessero automaticamente un indirizzo IP dal server DHCP, far sì che un nome di dominio "epicode.internal" fosse associato all'indirizzo IP del nostro server HTTP grazie al DNS, e infine verificare che tutto funzionasse correttamente facendo delle prove dai client.

Configurazione della rete:

Ho inserito nella rete dei **server** (Server-PT), uno con la funzione DHCP, uno con la funzione DNS e infine uno con la funzione HTTP. A seguire 3 switch e due laptop tutti collegati tramite cavo.



Dal DHCP apro la configurazione:

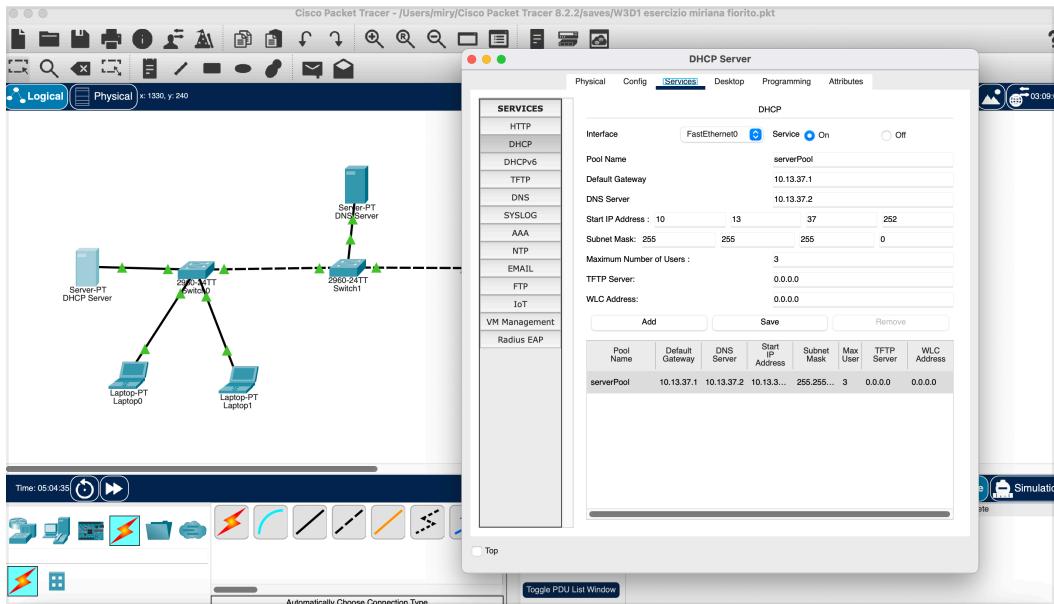
"Services" sezione "DHCP"

**Default gateway:** l'indirizzo IP Statico che ho prima assegnato al router dalla sezione --> Config. —> FastEthernet0 —> IP Configuration IPv4 Address 10.13.37.1 con Subnet Mask 255.255.255.0

**DNS server:** l'indirizzo del server DNS 10.13.37.2

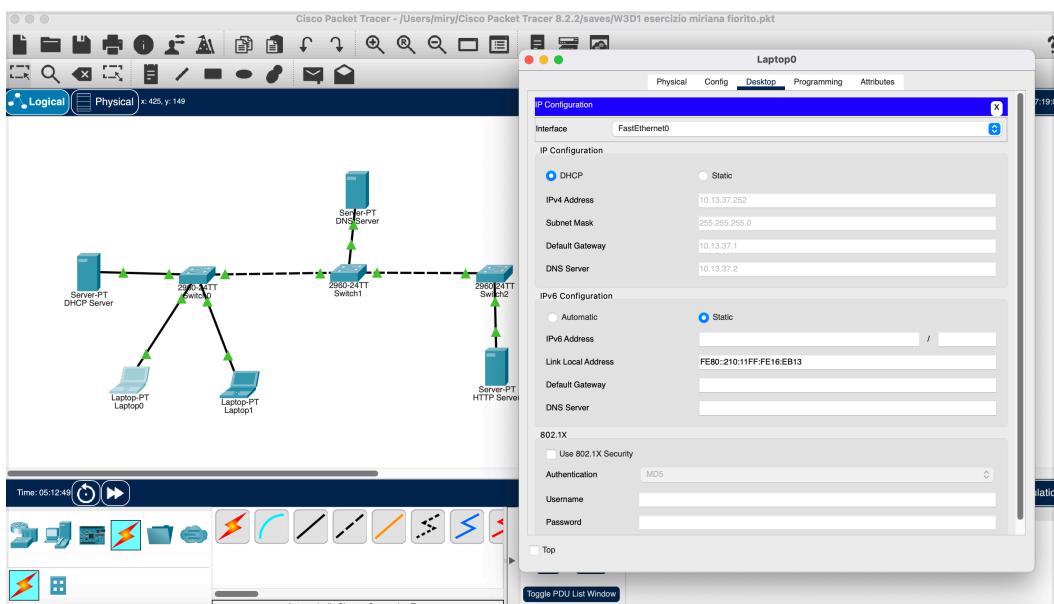
**Intervallo di IP (Start IP Address):** definisco il range di IP che il DHCP può assegnare, 10.13.37.252 , ossia 3 utenti massimi (da qui il numero 3 nel campo maximum number of users).

Subnet Mask: 255.255.255.0 —> "Save"

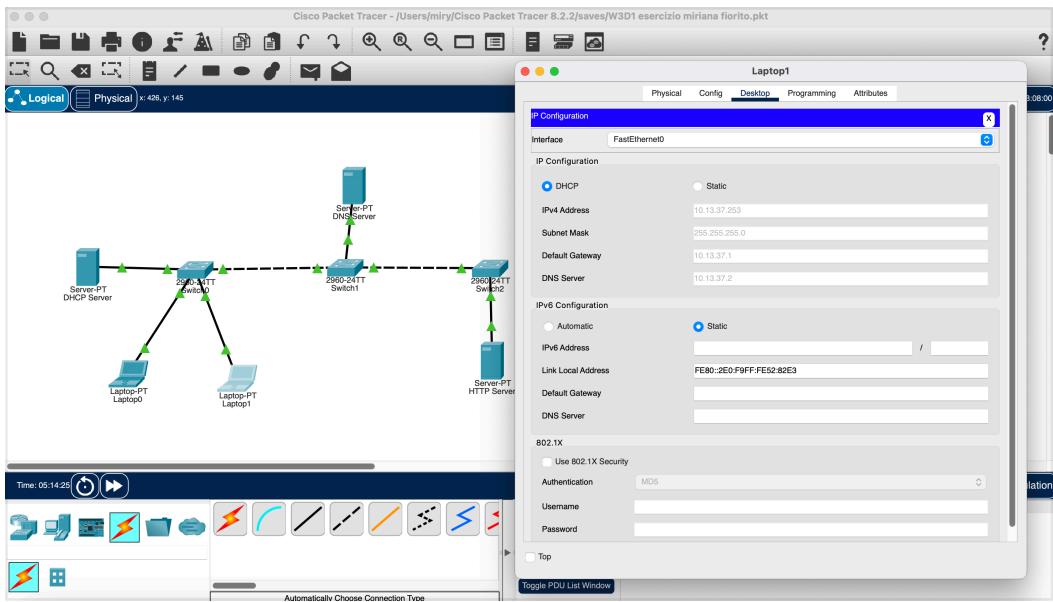


Una volta salvate queste impostazioni, ho aperto i due client della rete Laptop0 e Laptop1 → Desktop IP Configuration e ho scelto l'opzione DHCP. Dopo qualche secondo i client hanno ricevuto automaticamente un indirizzo IP all'interno del range che ho definito: segno che il server DHCP stava lavorando correttamente.

### Laptop0

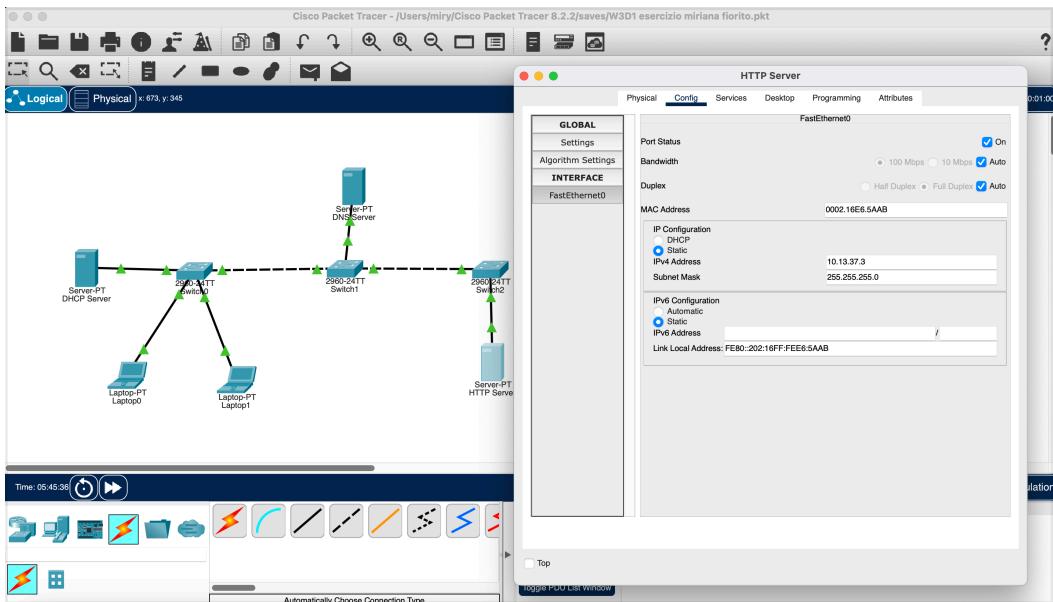


### Laptop1



Ho proseguito la configurazione del server HTTP.

→ Config. → FastEthernet0 ed ho assegnato un indirizzo IP statico IPv4:  
10.13.37.3 → Subnet Mask 255.255.255.0 → ON

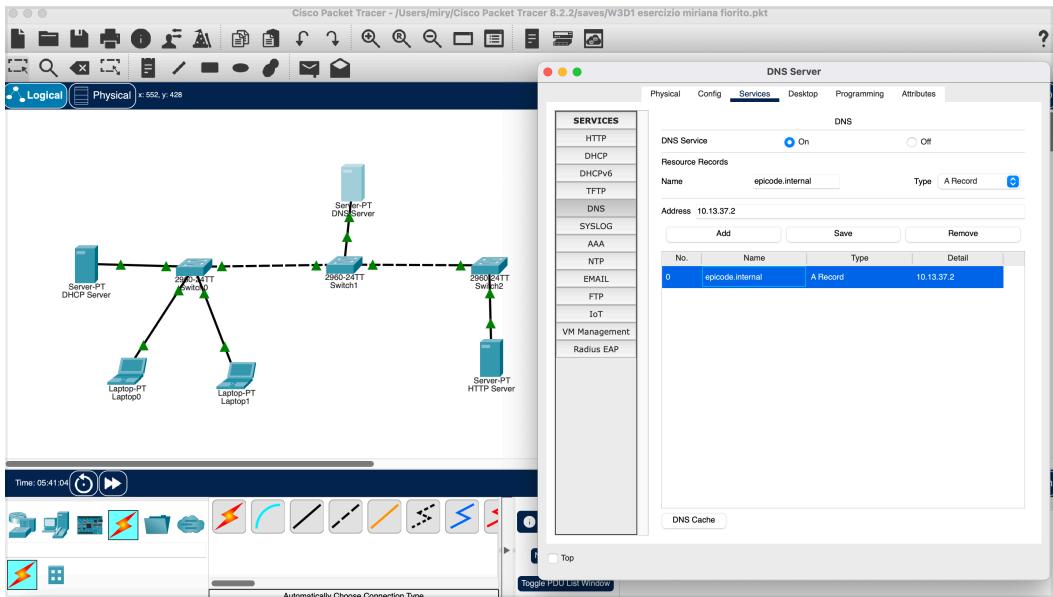


Infine ho effettuato la configurazione del server DNS, che traduce i nomi di dominio in indirizzi IP. Ho assegnato anche qui un indirizzo ip statico

→ Config. → FastEthernet0 → IPv4: 10.13.37.2 → Subnet Mask:  
255.255.255.0

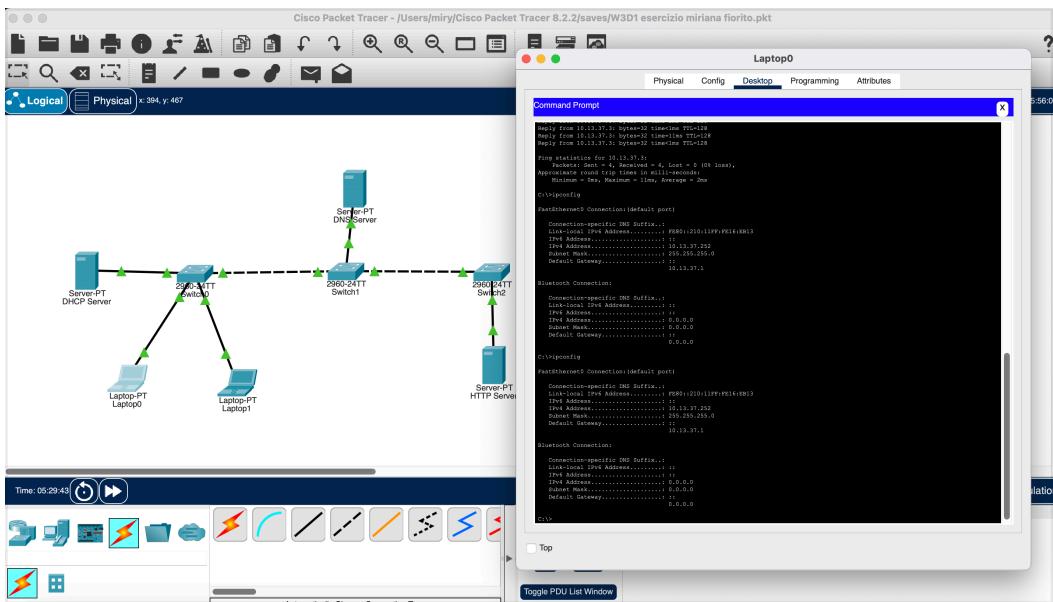
Creo un nuovo **record A** da → Services → DNS ON

- Name: episode.internal
- Type: A Record
- Address: l'IP del server HTTP (10.13.37.2)

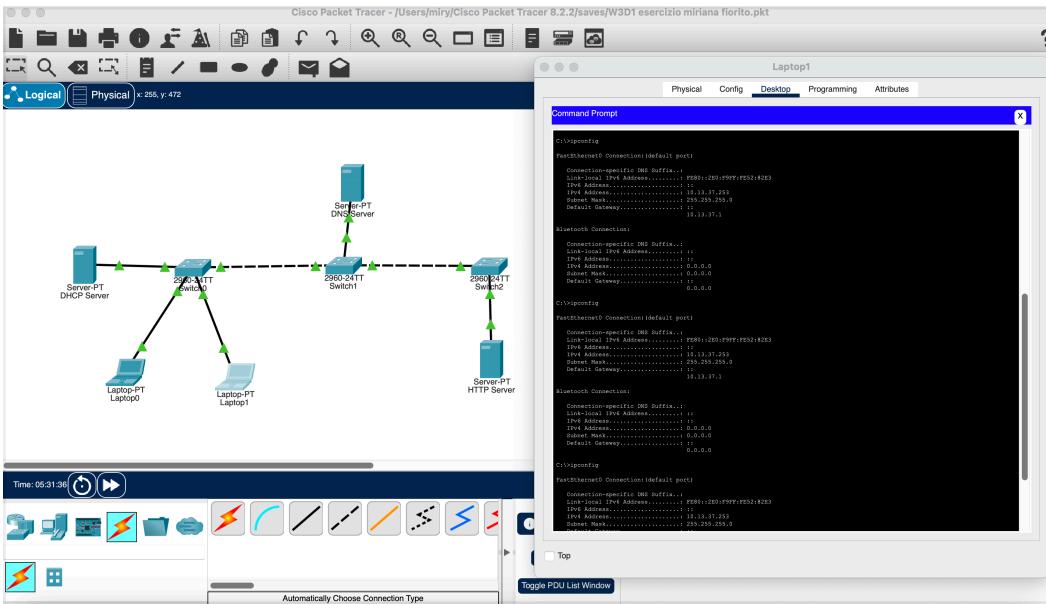


A questo punto ho effettuato la verifica. Sui client ho aperto il prompt dei comandi e digitato ipconfig, per controllare che l'indirizzo IP fosse stato assegnato dal server DHCP

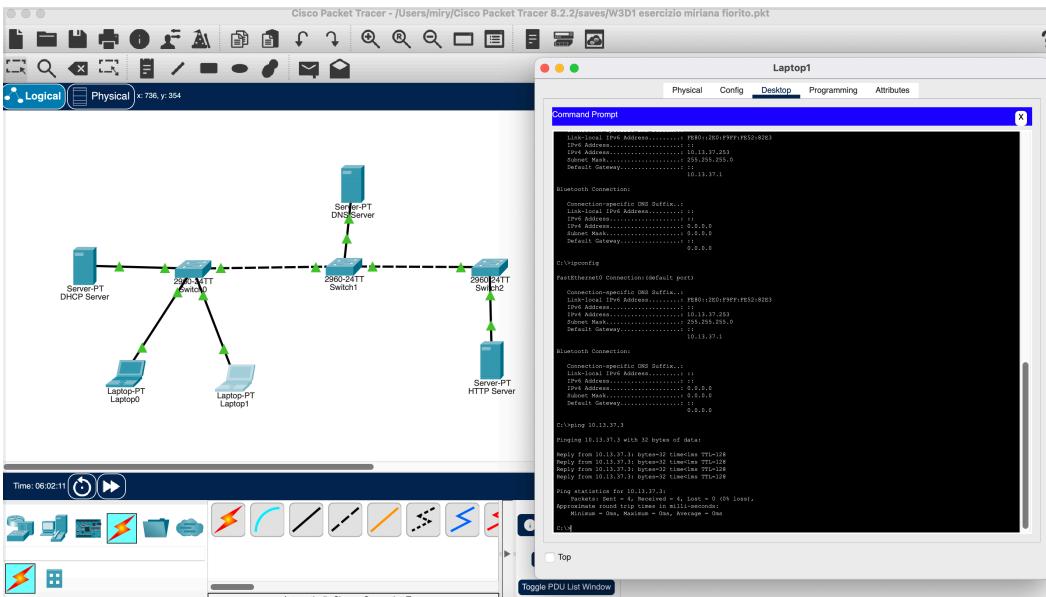
#### Laptop0 - Prompt - ipconfig



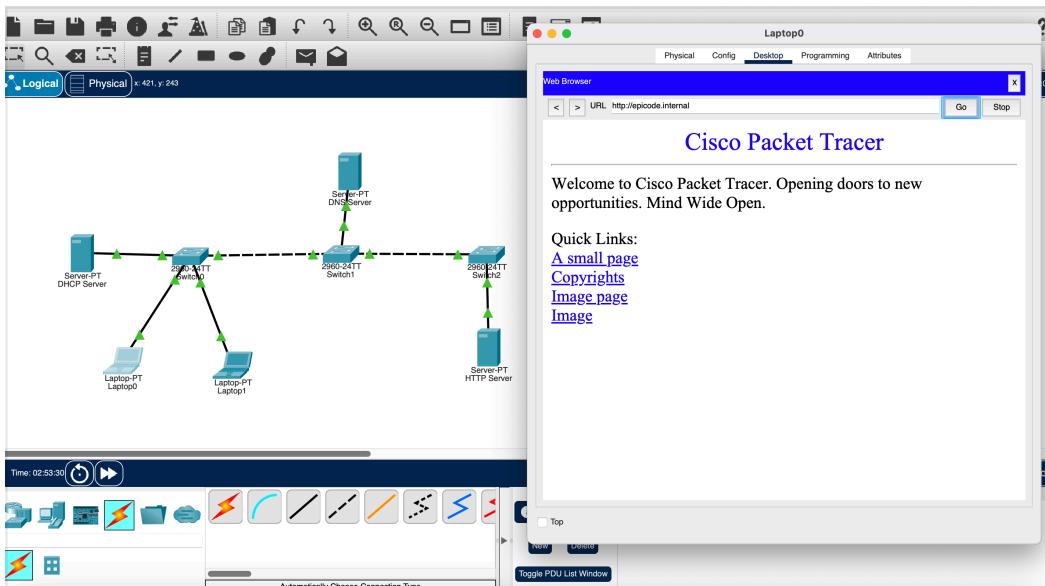
#### Laptop1 - Prompt - ipconfig



Ho anche effettuato un ping verso il server 10.13.37.3 come ulteriore verifica



Poi sono passata al browser del client e ho provato a digitare episode.internal nella barra degli indirizzi. Il browser ha inviato una richiesta al server DNS per sapere a quale IP corrispondeva quel nome, e il DNS ha risposto con l'indirizzo del server HTTP. Infine, il client ha contattato il server HTTP e la pagina web di esempio si caricava: segno che anche il DNS funzionava perfettamente.



Questo esercizio è stato utile anche per capire cosa succede dietro le quinte quando un client digita un nome in un browser: prima chiede al server DNS l'indirizzo IP corrispondente, poi usa quell'indirizzo per raggiungere il server e ottenere la pagina desiderata.

#### Facoltativo

Per concludere, ci stato chiesto di immaginare un sistema di videosorveglianza IP in una LAN e descrivere come i diversi livelli ISO/OSI collaborano per trasmettere le immagini dalle telecamere al server di registrazione.

Possiamo raccontarlo così:

tutto parte dal livello fisico, che fatto di cavi, switch e segnali elettrici portano i dati fisicamente da un punto all'altro. Poi interviene il livello di collegamento dati, che usa gli indirizzi MAC per identificare i dispositivi sulla rete e assicurarsi che i frame arrivino a destinazione.

Il livello di rete, invece, si occupa di instradare i pacchetti grazie agli indirizzi IP. Saliamo al livello trasporto, dove entra in gioco TCP, che garantisce che i dati arrivino completi e in ordine.

Il livello sessione mantiene la connessione attiva tra la telecamera e il server per tutto il tempo della registrazione. Il livello presentazione si occupa di convertire i dati in un formato che il server può comprendere. Infine, il livello applicazione gestisce le richieste delle telecamere e le risposte del server, permettendo di visualizzare e salvare le immagini.

In questo modo, passo dopo passo, abbiamo configurato una rete più complessa e capito come i diversi componenti e livelli lavorano insieme per farla funzionare.