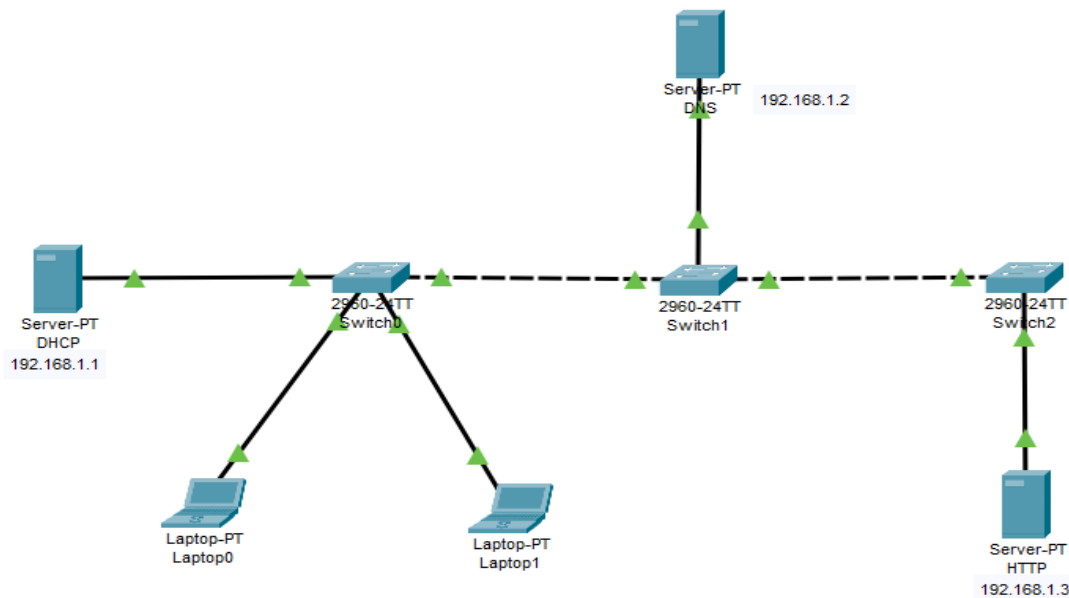


## REPORT

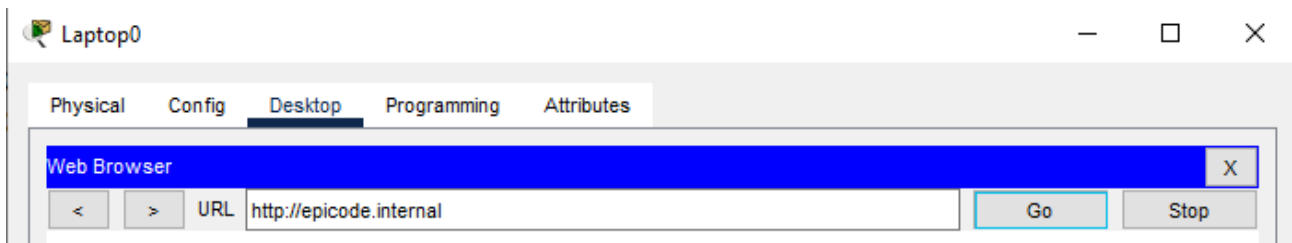
Usando il programma packet tracer, simuliamo la rete di 2 computer portatili, 1 server DHCP, 1 server DNS e 1 server HTTP, come nella rappresentazione sotto.



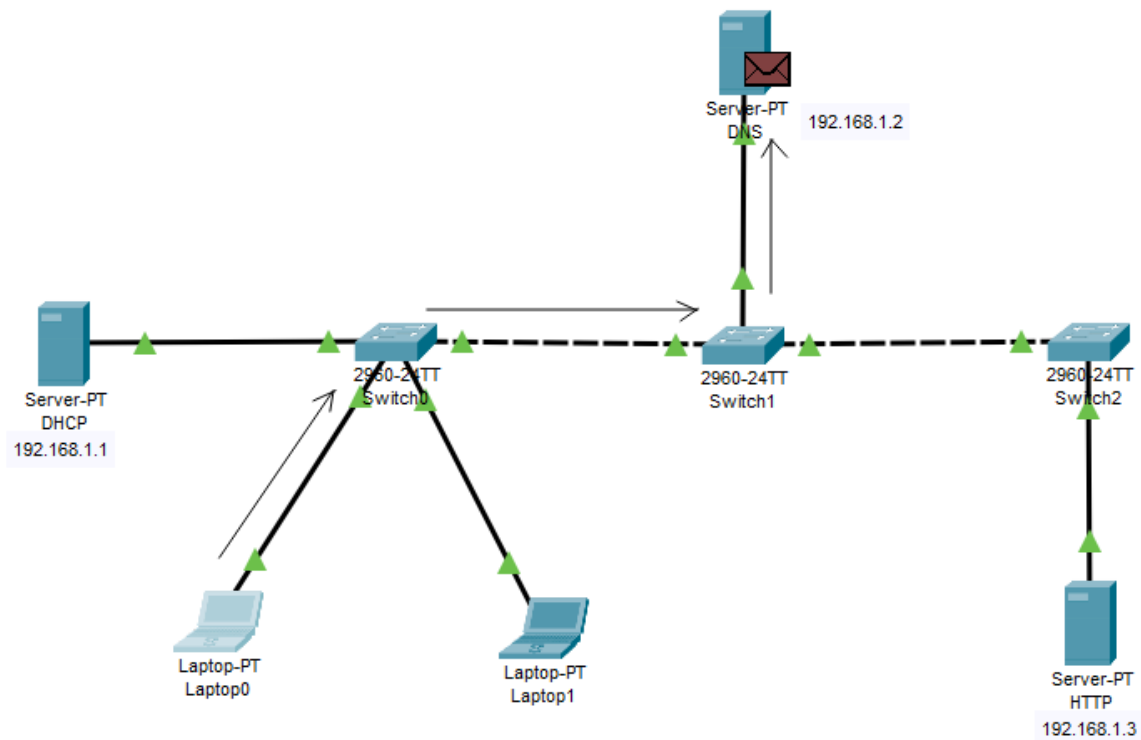
Oggi al posto di dare un indirizzo IP statico ai 2 client, li abbiamo connessi al server Dynamic Host Configuration Protocol (DHCP) "192.168.1.1" che non è altro che un protocollo di gestione della rete che assegna automaticamente gli indirizzi IP.

Abbiamo configurato anche il server Domain Name System (DNS) "192.168.1.2" che ha la funzione di gestire la mappatura tra nomi e numeri, una specie di "rubrica telefonica", dove abbiamo configurato il nostro sito "epicode.internal" al indirizzo del server HTTP "192.168.1.3"

Per accedere al sito, dal quale non sappiamo l'indirizzo IP, pero sappiamo il DNS ovvero "epicode.internal" e questo ci basta per inviare la richiesta al server ci basta scriverlo nel browser:

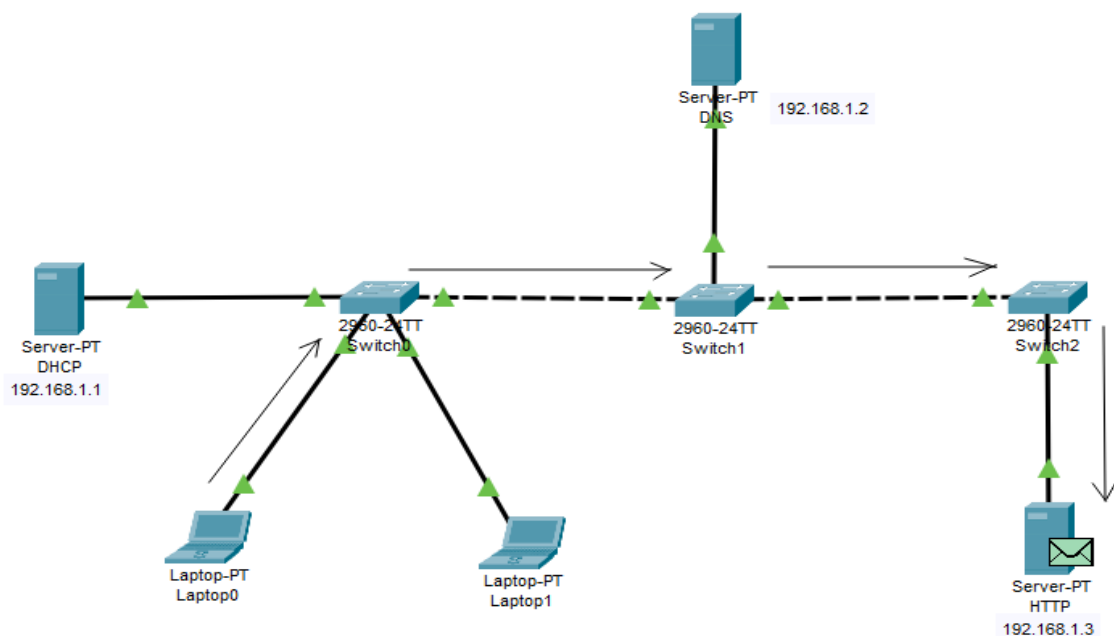


La nostra macchina andrà a fare la richiesta al server DNS per l'IP del nostro sito.



Una volta ricevuto la richiesta, il server DNS controllerà la “ribbrica” dei indirizzi cercando quello che concede con l'IP richiesto, e invierà una risposta al nostro client.

Con la risposta del DNS il nostro computer invia la richiesta di comunicare con il server al indirizzo IP giusto.



Raggiunto il server HTTP, il pacchetto di richiesta inviato dal client verrà ricevuto, controllato se si trova al posto giusto, e inizierà il “tree-way-handshake” cioè il computer client dal quale abbiamo inviato la richiesta di connessione con server, ha mandato una SYN, che sarebbe una flag del protocollo TCP che si occupa del trasporto di informazione, nel nostro caso ha impostato un sequence number 0.

PDU Information at Device: HTTP

OSI Model   Inbound PDU Details   Outbound PDU Details

At Device: HTTP  
Source: Laptop0  
Destination: 192.168.1.3

**In Layers**

Layer7
Layer6
Layer5
<b>Layer 4: TCP Src Port: 1030, Dst Port: 80</b>
Layer 3: IP Header Src. IP: 192.168.1.5, Dst. IP: 192.168.1.3
Layer 2: Ethernet II Header 000A.F3B9.4390 >> 000D.BD98.67A9
Layer 1: Port FastEthernet0

**Out Layers**

Layer7
Layer6
Layer5
Layer 4: TCP Src Port: 80, Dst Port: 1030
Layer 3: IP Header Src. IP: 192.168.1.3, Dst. IP: 192.168.1.5
Layer 2: Ethernet II Header 000D.BD98.67A9 >> 000A.F3B9.4390
Layer 1: Port(s): FastEthernet0

1. The device receives a TCP SYN segment on server port 80.  
2. Received segment information: the sequence number 0, the ACK number 0, and the data length 24.  
3. TCP retrieves the MSS value of 1460 bytes from the Maximum Segment Size Option in the TCP header.  
4. The connection request is accepted.  
5. The device sets the connection state to SYN\_RECEIVED.

Challenge Me   << Previous Layer   Next Layer >>

A questo punto il server andrà a rispondere inviando al client il pacchetto con i flag SYN e ACK abilitati, questa volta il numero di sequenza sarà un numero casuale, mentre l'ACK sarà uguale al numero di sequenza ricevuto dal client aggiungendo 1.

Insieme al SYN e ACK il server invia anche altre informazioni come il port del indirizzo sul quale può conettersi il client, la grandezza del pacchetto di risposta.

PDU Information at Device: HTTP

OSI Model   Inbound PDU Details   Outbound PDU Details

At Device: HTTP  
Source: Laptop0  
Destination: 192.168.1.3

In Layers	Out Layers
Layer7	Layer7
Layer6	Layer6
Layer5	Layer5
Layer 4: TCP Src Port: 1030, Dst Port: 80	Layer 4: TCP Src Port: 80, Dst Port: 1030
Layer 3: IP Header Src. IP: 192.168.1.5, Dest. IP: 192.168.1.3	Layer 3: IP Header Src. IP: 192.168.1.3, Dest. IP: 192.168.1.5
Layer 2: Ethernet II Header 000A.F3B9.4390 >> 000D.BD98.67A9	Layer 2: Ethernet II Header 000D.BD98.67A9 >> 000A.F3B9.4390
Layer 1: Port FastEthernet0	Layer 1: Port(s): FastEthernet0

1. TCP accepts a window size up to 16384 bytes.
2. TCP adds Maximum Segment Size Option to the TCP SYN-ACK header with Maximum Segment Size equal to 536 bytes.
3. The device sends a TCP SYN+ACK segment.
4. Sent segment information: the sequence number 0, the ACK number 1, and the data length 24.

Challenge Me   << Previous Layer   Next Layer >>

La risposta torna dal client con la conferma del avvenuta iniziativa del “Three-way-handshake” e con delle informazioni del SYN e ACK inviate dal server.

Ancora una volta il computer client invia TCP ACK per completare la sincronizzazione, modificando il sequence number inviato dal server in ACK e aggiungendo il valore di 1.

Raggiunto il server HTTP, il server apre il pacchetto stabilendo la connessione TCP con il client.

26/10/2022

Filip S.