## Setting Up a Web Service

**Topology: A PC and a Web server**



Fig 1. A PC and a Web Server

## Task 1: Set up a web server with a basic website

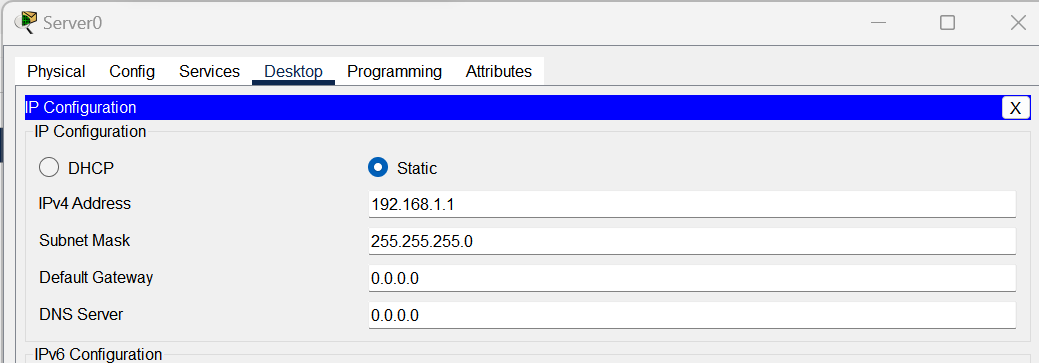


Fig 2. Configuration of the Web Server

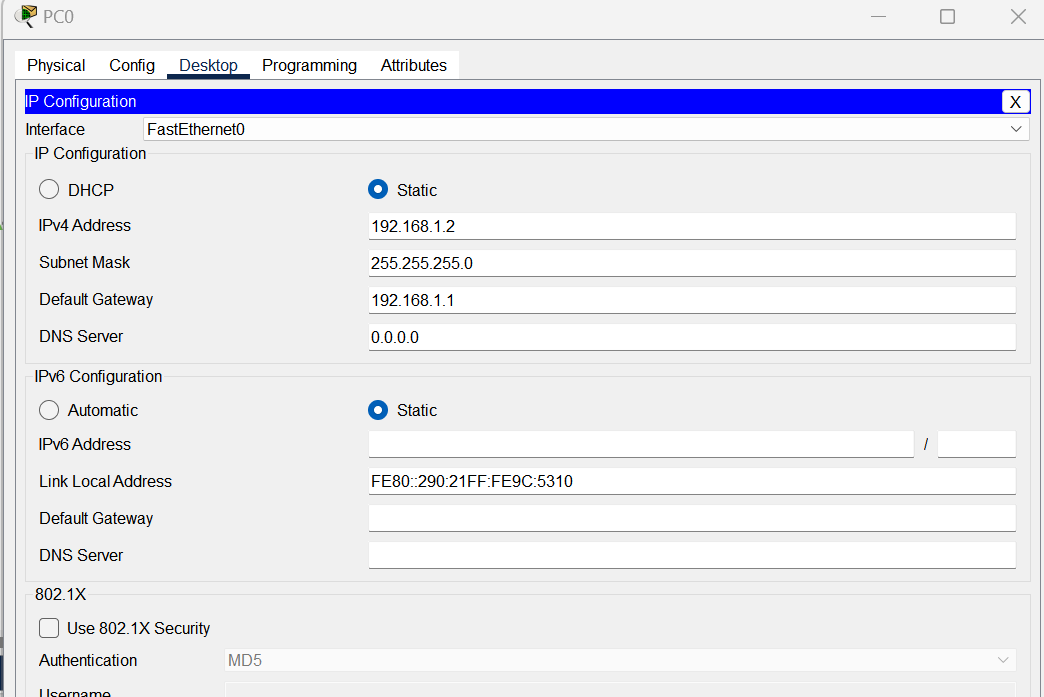


Fig 3. Configuration of the PC

**Step 1**: I opened my Cisco Packet Tracer project.

**Step 2**: In the workspace, I selected the web server device by clicking on it.

**Step 3**: Using the device details pane, I found the "Config" tab, which provides access to the device's configuration settings.

**Step 4**: In the "Config" tab, I located the section for configuring the IP address and subnet mask, typically labeled as "IP Configuration" or "Network Configuration."

**Step 5**: In the IP configuration section of the PC, I filled in the following details:

* IP Address: 198.162.1.2
* Subnet Mask: 255.255.255.0
* Default Gateway: 192.168.1.1

**Step 6**: I entered the desired IP address for the web server, ensuring it was within the same subnet as the PC. For instance:

* IP Address: 192.168.1.1
* Subnet Mask: 255.255.255.0

**Step 7**: If necessary, I configured the default gateway and DNS server settings for external connectivity. If not needed, I left these fields empty.

**Step 8**: After entering the information, I saved the configuration.

**Step 9**: I closed the configuration window, finalizing the IP address and subnet mask setup for the web server.

This configuration was crucial to ensure that the web server could communicate effectively with the PC in the network.

# Task 2: From the PC, use a browser to access the website

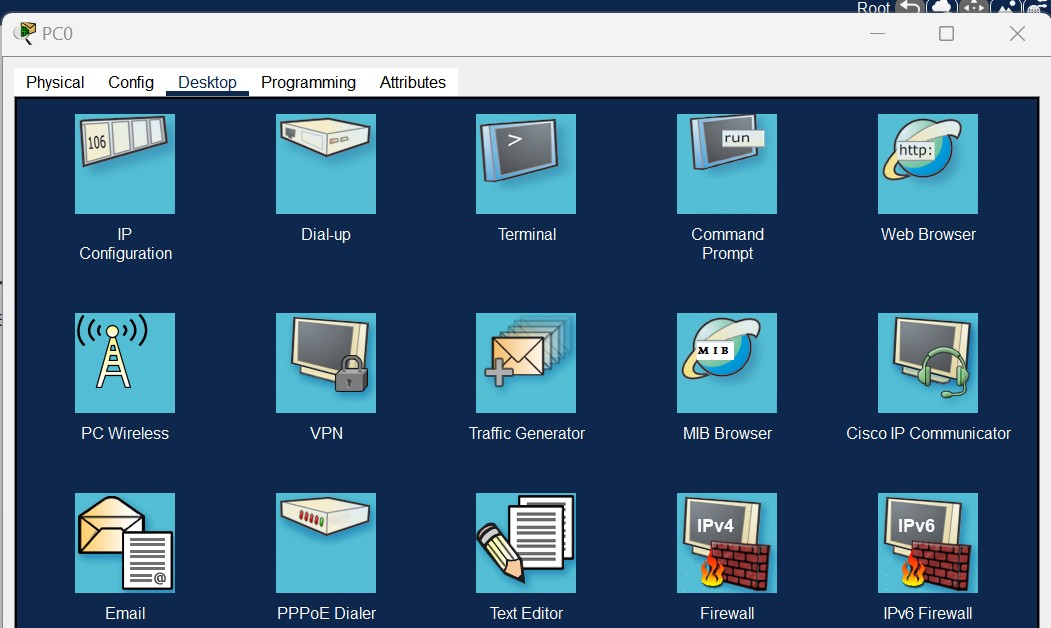


Fig 4. Finding the Web Broswer in PC0

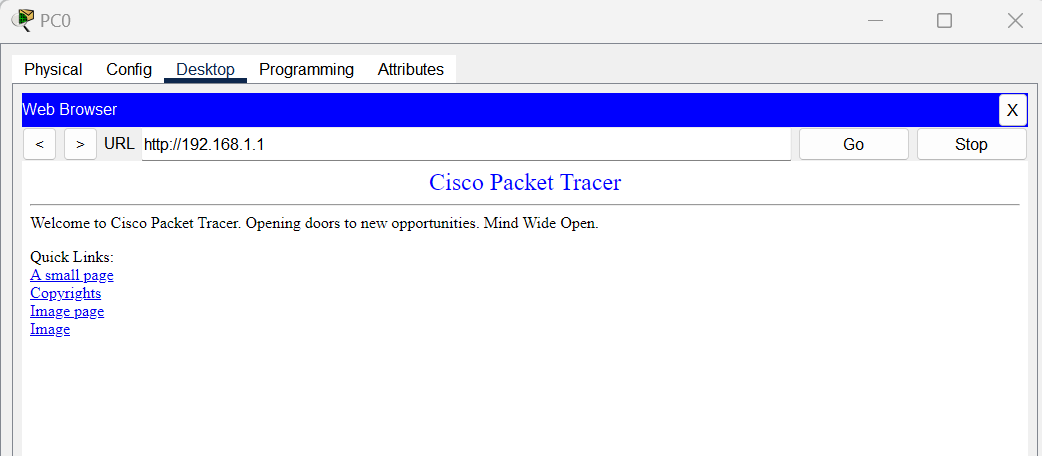


Fig 5. Entering the Web Server’s address

**Step 1**: Network Configuration I ensured that both the PC and web server had their IP addresses correctly configured within the same subnet to enable communication.

**Step 2**: Start the PC If the PC wasn't already powered on, I activated it by clicking on it and using the "Power" button in the "Physical" tab.

**Step 3**: Open the Web Browser. I accessed the PC's desktop by clicking on it and selecting the "Desktop" tab.

**Step 4**: Launch the Web Browser Within the PC's desktop, I opened the web browser by clicking its icon.

**Step 5**: Access the Web Server In the web browser's address bar, I typed the web server's IP address: "[http://192.168.1.1](http://192.168.1.1/)," and pressed Enter.

This action sent an HTTP request from the PC to the web server. If the configuration was correct, I could view the web server's hosted website. The key was to ensure the IP address in the browser matched the web server's configuration.

# Task 3: Observe the TCP three-way handshake and the HTTP GET request

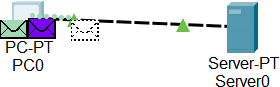


Fig 6. Three-way handshake

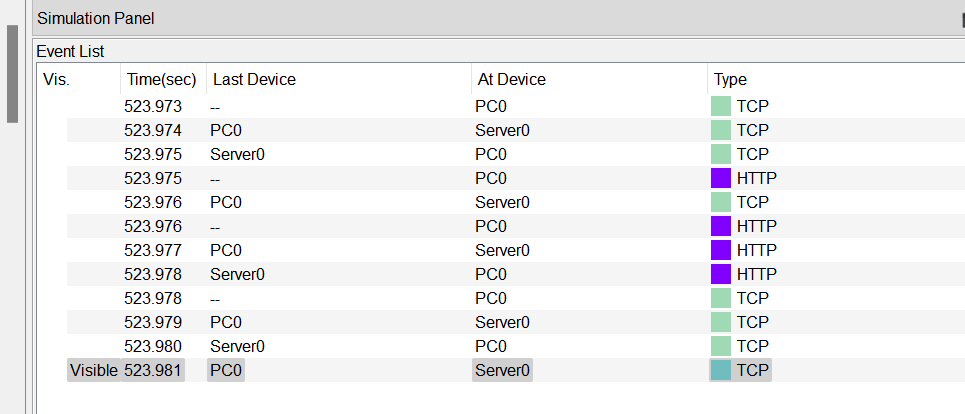


Fig 7. Stimulation Analysis

I initiated a three-way handshake to establish a connection between my PC (PC0) and the web server (Server0) in the Cisco Packet Tracer simulation.

I started the process by sending a signal to the server, essentially saying, "Hey, can we communicate?" This signal is using the "TCP" protocol. This step is like extending a hand for a handshake in real life, signifying my intent to connect.

In response, Server0 acknowledged my request. It replied, "Yes, let's communicate," and it also asked me to confirm that I was ready to proceed. It sent back a signal with both the "TCP SYN" and "ACK" flags. This is like the other person shaking your hand and asking, "Are you ready?"

I confirmed that I was ready to communicate by sending another signal with the "TCP ACK" flag. It's like shaking the other person's hand and saying, "Confirmed, let's communicate."

This process is known as the "three-way handshake," and it ensures that both sides are ready to establish a connection and start communicating over the TCP protocol.

## Analysis:

**TCP Three-Way Handshake - Secure Start:**

* + The TCP three-way handshake is like a secure greeting between devices. It ensures that both sides are ready to communicate. The first step (SYN) is akin to extending a hand for a handshake. The second step (SYN-ACK) is the other

party saying, "Let's communicate!" The final step (ACK) is like confirming, "Yes, let's talk." This handshake establishes a reliable connection.

## Web Server - Content Provider:

* + Web servers are like friendly waiters at a restaurant. When you ask for a web page, they fetch it and serve it to you. They handle your request and make sure you get the web content you want.

## Challenges and Solutions in Web Communication:

* + In web conversations, hiccups can occur. Slow connections or data loss can be problematic. To fix these issues, we can identify and address them. It's a bit like ensuring a smooth and uninterrupted phone call.

## Stateful vs. Stateless Protocols - Remembering the Past:

* + Some protocols remember what you've talked about in previous conversations (stateful). Others don't remember anything (stateless). It's similar to conversations where some people recall what you discussed earlier, while others start fresh every time.

## HTTPS - Keeping Secrets:

* + HTTPS is like speaking in a secret code to keep your online conversations private. It encrypts your data to protect it from prying eyes. It's like sealing a letter in an envelope to keep its contents confidential.

Each of these aspects plays a crucial role in how web services work and how they ensure reliability, security, and effective communication. Understanding them is key to maintaining a smooth and secure web experience.