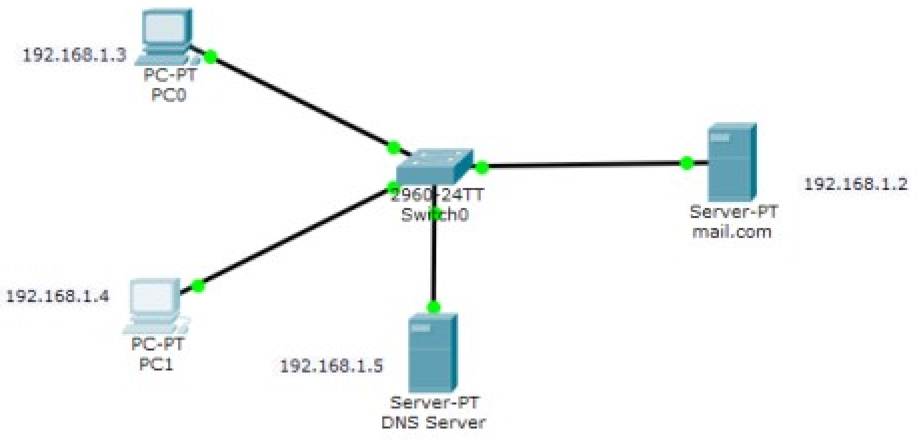
Mail Server

An *email server*, such as Gmail stores and sends email messages to email clients on request. We often send and receive emails on our mobile devices or computers. Have you ever imagined how this happens? Well, whenever you compose and send an email to another person, the message you send first goes to a mail server. It's the mail server which then sends the email when it is requested from the email client(e.g Gmail App) of the recipient's device.

So now, lets configure a mail server in Packet Tracer. And have in mind that although our main focus is configuring an email server, we'll still need services of a *DNS server* at one point.

Let's dive right in.

1. Build the network topology:



2.Configure IP addresses on the **PCs**, **DNS Server** and the M**ail Server**.

Mail Server IP address: 192.168.1.2/24

PC0 **IP address**: 192.168.1.3/24

PC1 **IP address:** 192.168.1.4/24

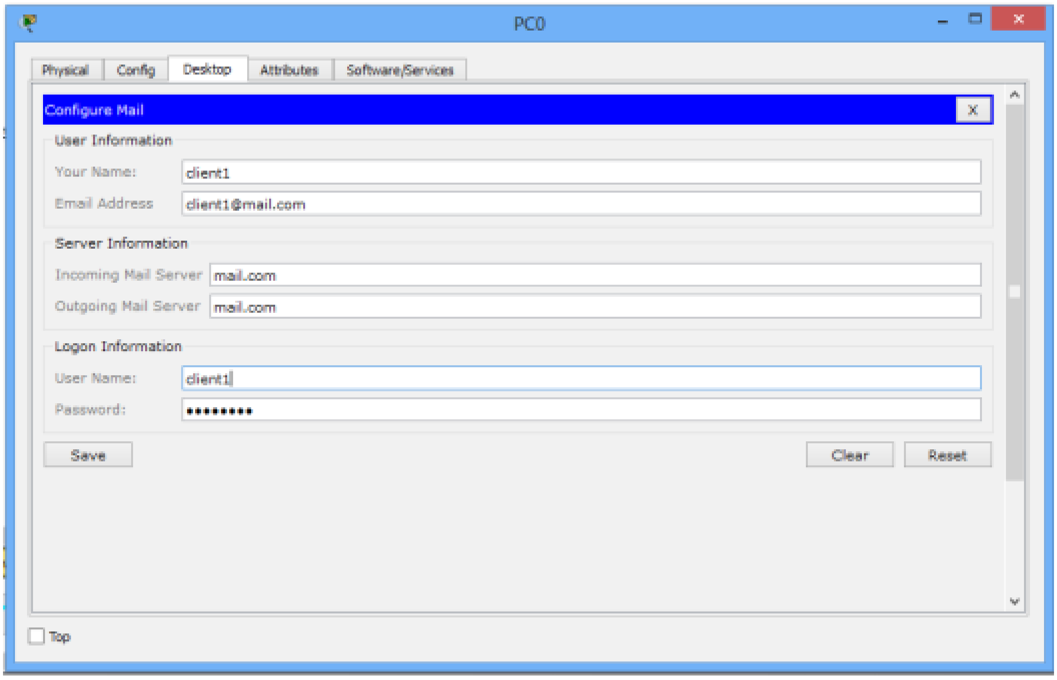
DNS server **IP address**: 192.168.1.5/24

3. Now configure **mail clients** on the **PCs** and **mail service** on the **generic server**.

Mail Clients:

Click on **PC0**. Go to its **Desktop** tab, and click on **Email.** Configure the email client by filling in the user, server and login information. Be sure to **Save.**

PC0:



Configure Mail

User Information

Your Name: client1

Email Address: client1@mail.com

Server Information

Incoming Mail Server: mail.com

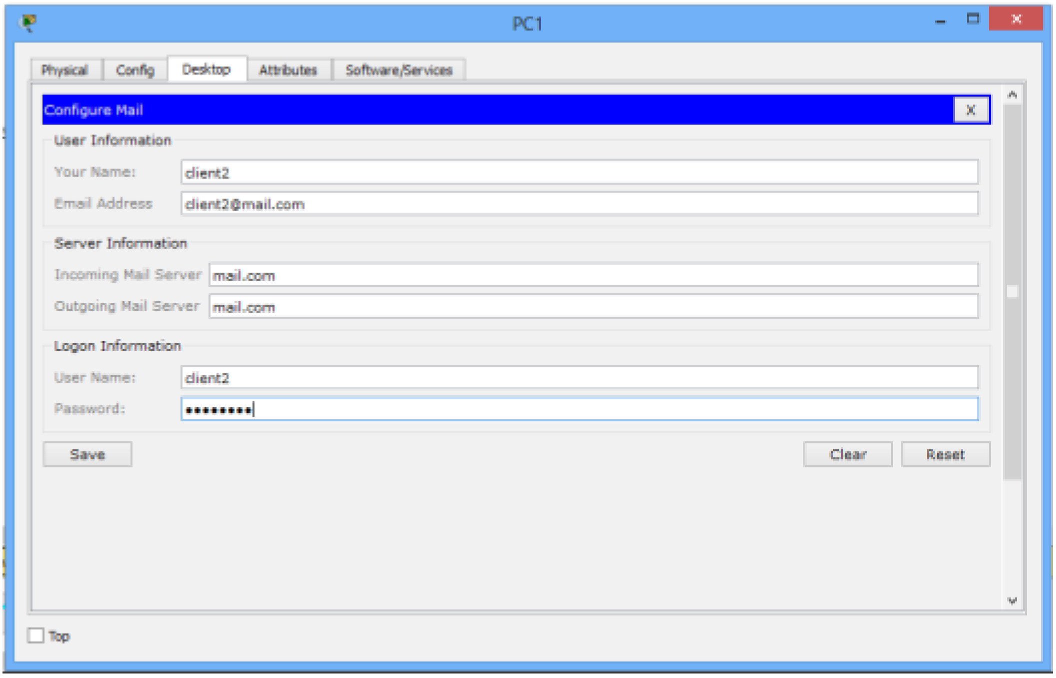
Outgoing Mail Server: mail.com

Logon Information

User Name: client1

Password: •••••••••

Configure mail client on PC1 in a similar way we did for PC1.



Configure Mail

User Information

Your Name: client2

Email Address: client2@mail.com

Server Information

Incoming Mail Server: mail.com

Outgoing Mail Server: mail.com

Logon Information

User Name: client2

Password: ••••••••

PC1:

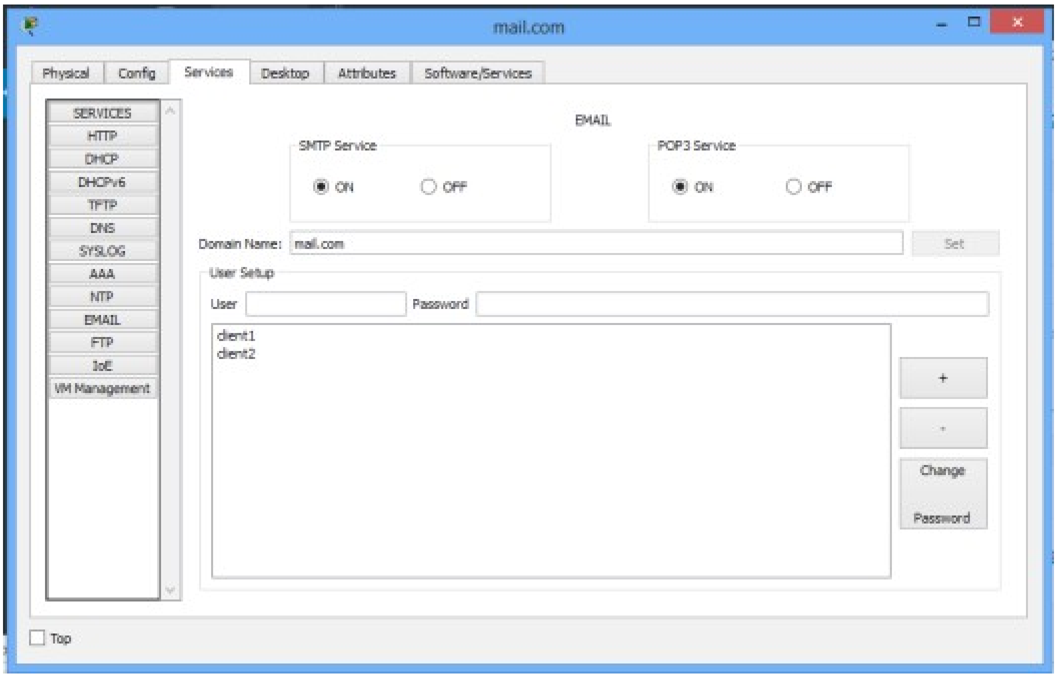
Next, we’ll configure the **email server.**

To do this, click on the server, then click **Services** tab, pick **email** server from the menu.

Provide the **Domain name** of the server then click on **Set** to set it. In this example I’ve used the name ‘mail.com’ .

Proceed and add **users** and provide their **passwords**. I have two email clients(users) with usernames ‘client1‘ and ‘client2‘ with a common password **‘**adminkim‘

After entering a username and password, click on **Add(+)** to add the user to the server. You can optionally remove a user by clicking on **Remove (-).** You can change a user’s password by clicking on **change password.**



mail.com

EMAIL

SMTP Service: ● ON ○ OFF

POP3 Service: ● ON ○ OFF

Domain Name: mail.com

User Setup

User: Password:

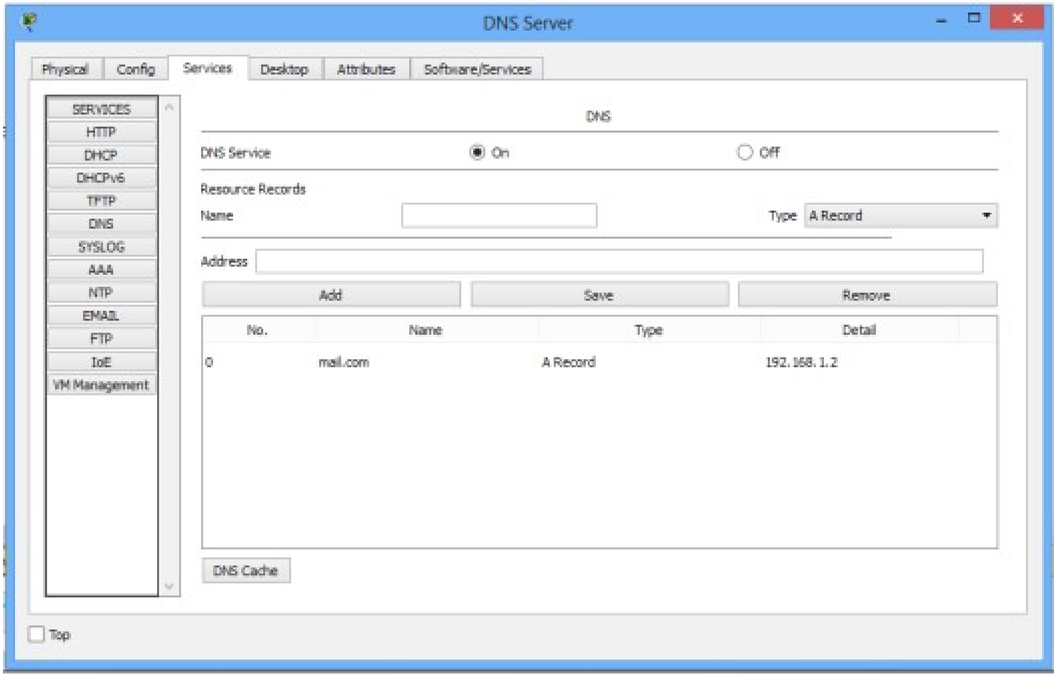
client1
client2

Try to relate this process to what happens when you register an email account with a mail service provider(mail server) like Gmail. The processes appear to agree, isn’t it?

Now, notice that we set a **domain name** for the email server. For that reason, we should have a **DNS server** that will resolve this domain name (plus other domain names if there were) to an IP address.

So let’s configure a DNS server.

Click DNS server, click **Services** tab, then pick **DNS**. Turn the service **ON.** Set name-address pairs and add them to the server. You can view the DNS entry below:



DNS

DNS Service ● On ○ Off

Resource Records

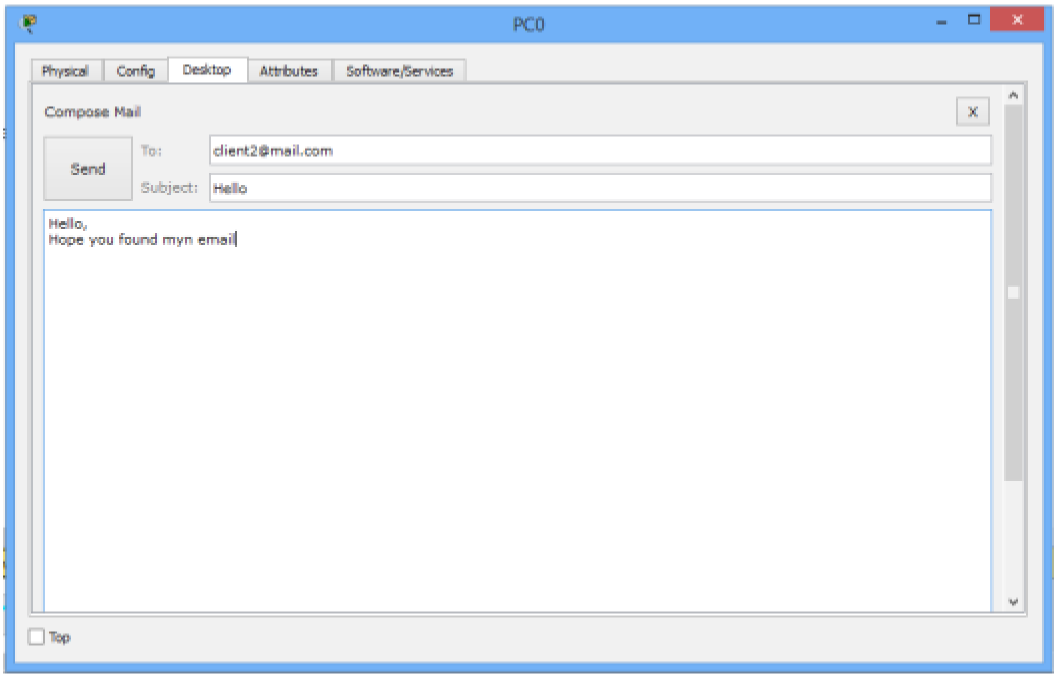
Name

Type A Record

Address

No.	Name	Type	Detail
0	mail.com	A Record	192.168.1.2

4. Lastly test the email service. Go to **PC0 email** client, **compose** an email and **send** its to **PC1** email address (client2@mail.com).



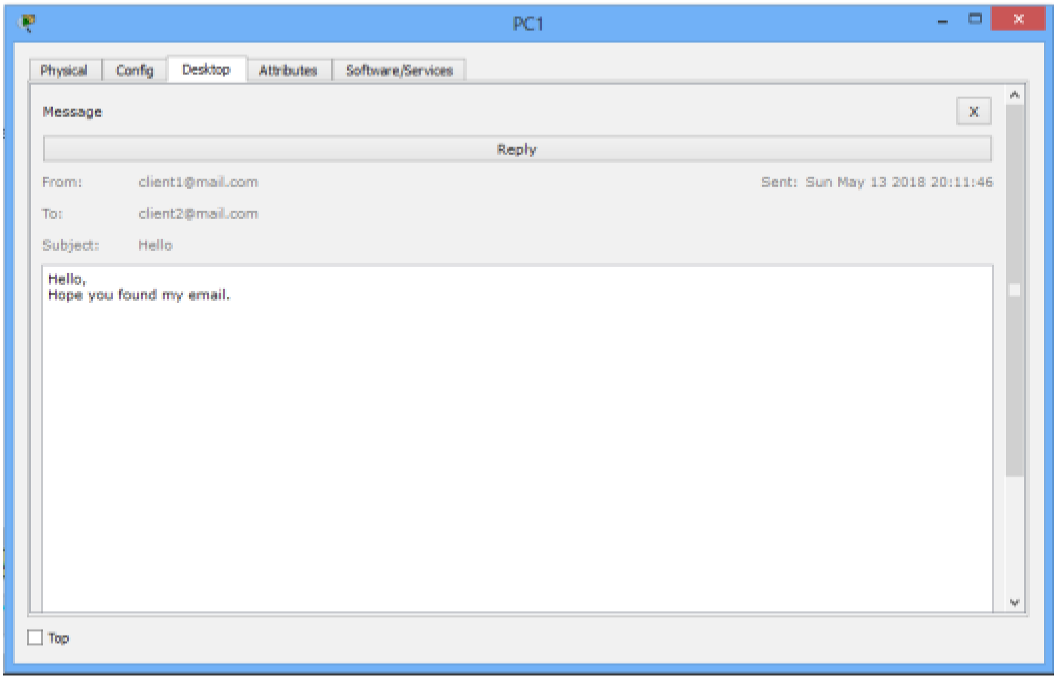
Compose Mail

To: client2@mail.com

Subject: Hello

Hello,
Hope you found myn email

Try to see whether the email from **PC0** is received on **PC1**. On the **email** client of PC1, click on **Receive.**



From: client1@mail.com
Sent: Sun May 13 2018 20:11:46
To: client2@mail.com
Subject: Hello

Hello,
Hope you found my email.

If everything is well set up, the email from **PC0** will be well received on **PC1**.

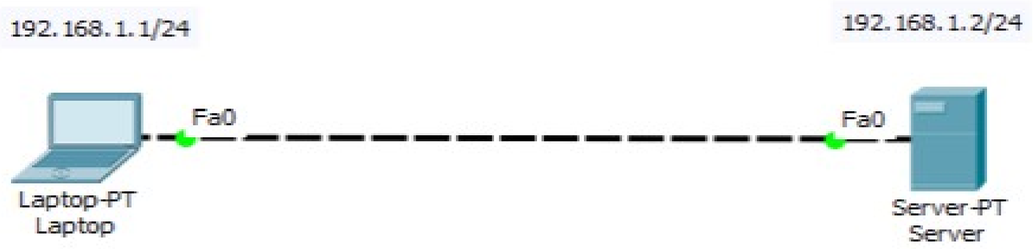
File Transfer Protocol (FTP)

The **File Transfer Protocol (FTP)** is a standard network protocol used for the transfer of computer files between a client and server on a computer network.

FTP employs a **client-server** architecture whereby the client machine has an **FTP client** installed and establishes a connection to an **FTP server** running on a remote machine. After the connection has been established and the user is successfully authenticated, the data transfer phase can begin.

Let’s now do FTP configuration in Packet Tracer:

1.Build the network topology.



2. Configure static IP addresses on the Laptop and the server.

Laptop: **IP address:** 192.168.1.1 **Subnet Mask:** 255.255.255.0

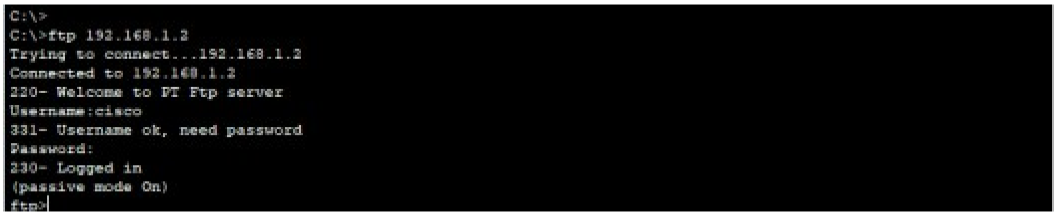
Server: **IP address:** 192.168.1.2 **Subnet Mask:** 255.255.255.0

3. Now try using an **FTP client** built in the Laptop to send files to an **FTP server** configured in the Server.

From the Laptop’s command prompt, FTP the server using the server IP address by typing:

```
ftp 192.168.1.2
```

Provide the **username**(cisco) and **password**(cisco) [which are the defaults] for ftp login.



```
C:\>
C:\>ftp 192.168.1.2
Trying to connect...192.168.1.2
Connected to 192.168.1.2
220- Welcome to PT Ftp server
Username:cisco
331- Username ok, need password
Password:
230- Logged in
(passive mode On)
ftp>
```

You are now in the FTP prompt .

PC0 has an **FTP client** which can be used to read, write, delete and rename files present in the FTP server.

The **FTP server** can be used to read and write configuration files as well as IOS images. Additionally, the FTP server also supports file operations such rename, delete and listing directory.

With that keep in mind, we can do something extra. So let’s do this:

4. **Create** a file in the Laptop then **upload** it to the server using **FTP**.

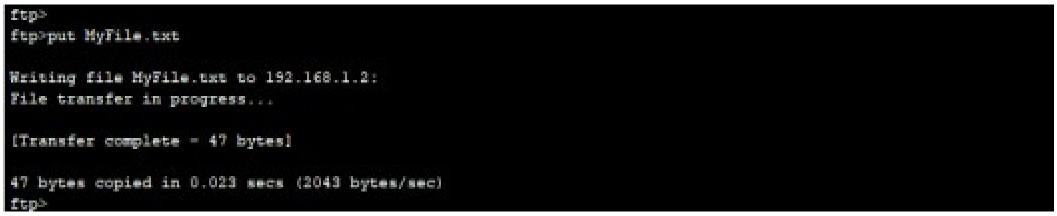
To do this, open the **Text Editor** in the Laptop, create a file and give it your name of choice.

Type any text in the editor then **save** your file. e.g. myFile.txt.

5. Now upload the file from the Laptop to the server using FTP. (An FTP connection has to be started first. But this is what we’ve done in step 3)

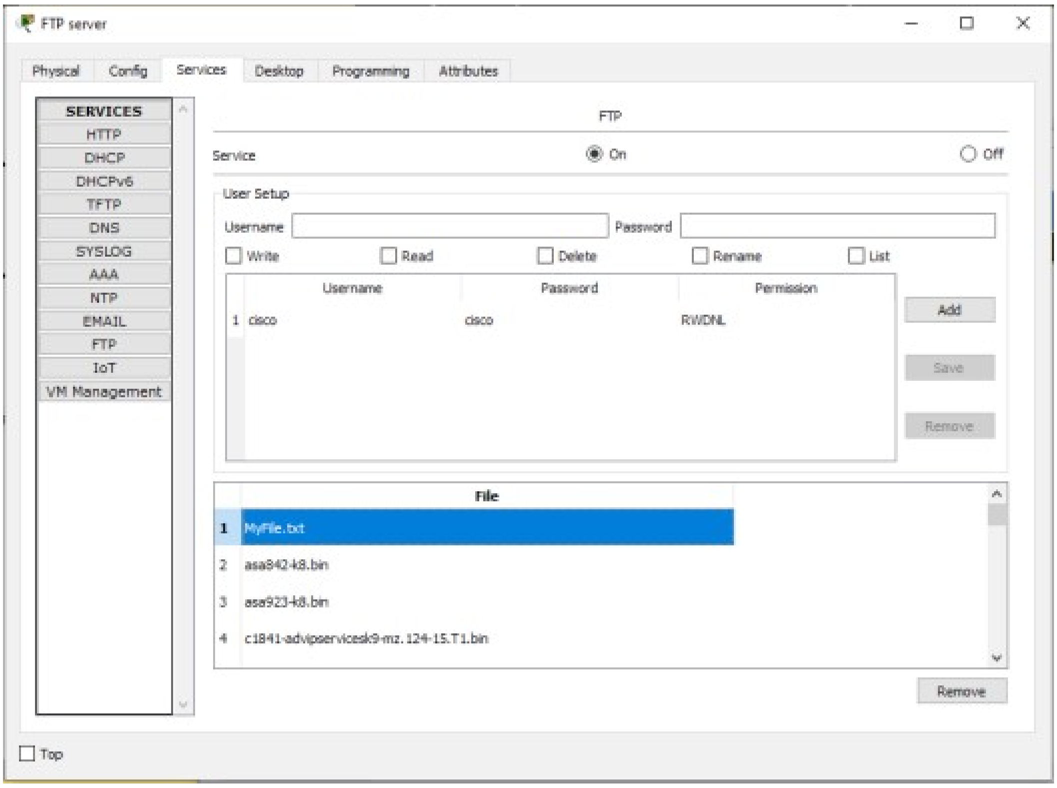
So to do an FTP upload, we’ll type:

```
 put MyFile.txt
```



```
ftp>
ftp>put MyFile.txt

Writing file MyFile.txt to 192.168.1.2:
File transfer in progress...

[Transfer complete - 47 bytes]

47 bytes copied in 0.023 secs (2043 bytes/sec)
ftp>
```

6. Once file upload is successful, go to the Server **FTP directory** to verify if the file sent has been received . To do this, go to **Server-> Services->FTP**. Here look for MyFile.txt sent from the laptop.



FTP

Service ● On ○ Off

User Setup

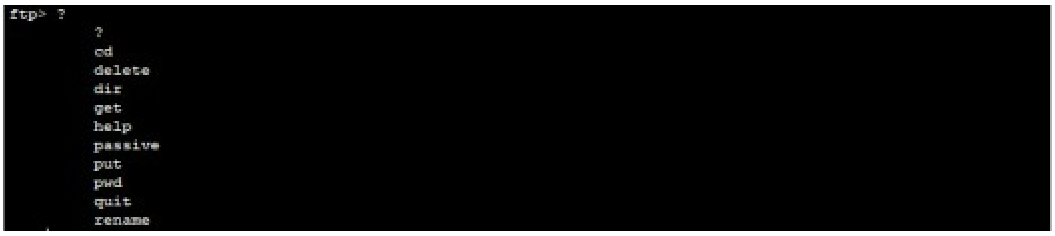
Username Password

□ Write □ Read □ Delete □ Rename □ List

	Username	Password	Permission
1	cisco	cisco	RWDNL

	File
1	MyFile.txt
2	asa842-k8.bin
3	asa923-k8.bin
4	c1841-advipservicesk9-mz.124-15.T1.bin

Something extra: To check **other FTP commands** supported by the FTP client running on the Laptop(or PC), you can use a question mark (**?**) on the Laptop's command prompt as shown below:



```
ftp> ?
         ?
         cd
         delete
         dir
         get
         help
         passive
         put
         pwd
         quit
         rename
```

You can see the put command that we used to upload our file to the FTP server. Other commands listed include:

get-used to get(download) a file from the server.

For example: get MyFile.txt

delete– to delete a file in the FTP directory with the server

For example: delete MyFile.txt

Rename– used to Rename a file

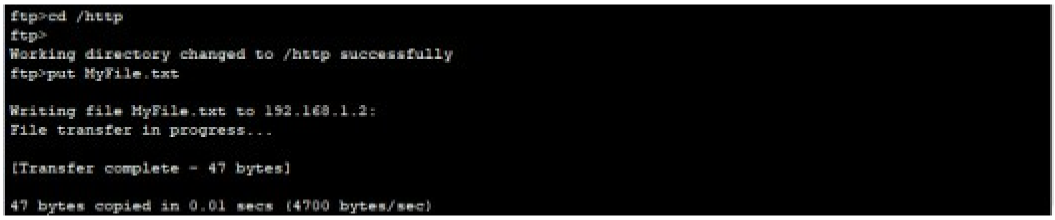
cd – used to change directory.

For example, we can open an **HTTP directory** in the server by typing: cd /http. This will change the current directory from FTP directory to HTTP directory

Once the http directory is open, you can upload a file to the HTTP server. You're now uploading a file to an HTTP folder(directory) using FTP.

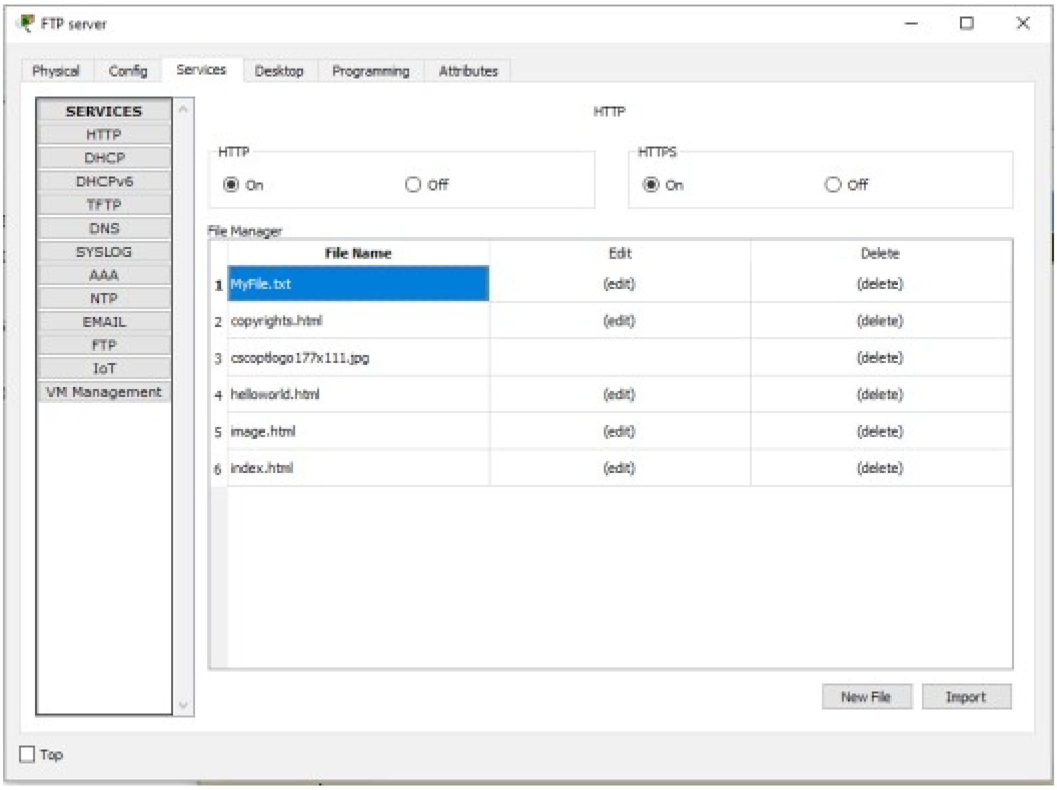
For example: put MyFile.txt

To see this working, let’s **open** an **HTTP directory** and upload(**put**) a file to it using FTP:



```
ftp>cd /http
ftp>
Working directory changed to /http successfully
ftp>put MyFile.txt

Writing file MyFile.txt to 192.168.1.2:
File transfer in progress...

[Transfer complete - 47 bytes]

47 bytes copied in 0.01 secs (4700 bytes/sec)
```

You can now check up in the **HTTP directory** in the server and verify that the file uploaded from the Laptop(MyFile.txt) is well received:



HTTP

HTTP: ● On ○ Off

HTTPS: ● On ○ Off

File Manager

	File Name	Edit	Delete
1	MyFile.txt	(edit)	(delete)
2	copyrights.html	(edit)	(delete)
3	cscoptlogo177x111.jpg		(delete)
4	helloworld.html	(edit)	(delete)
5	image.html	(edit)	(delete)
6	index.html	(edit)	(delete)

Notice that we are uploading files to an HTTP Server directory using File Transfer Protocol.(FTP). This is what actually happens when you use an **FTP client** such as FileZilla client to upload files to a website. In our case here, we are using an FTP client **built-in** the Laptop.

This may interest you: *T*he first FTP client applications were command-line programs developed before operating systems had graphical user interfaces, and are still shipped with most Windows and Linux operating systems. (Actually this is what we have been using this far). Many FTP clients(e.g. FileZilla) and automation utilities have since been developed for desktops, servers, mobile devices, and hardware. FTP has also been incorporated into productivity applications, such as HTML *editors*.

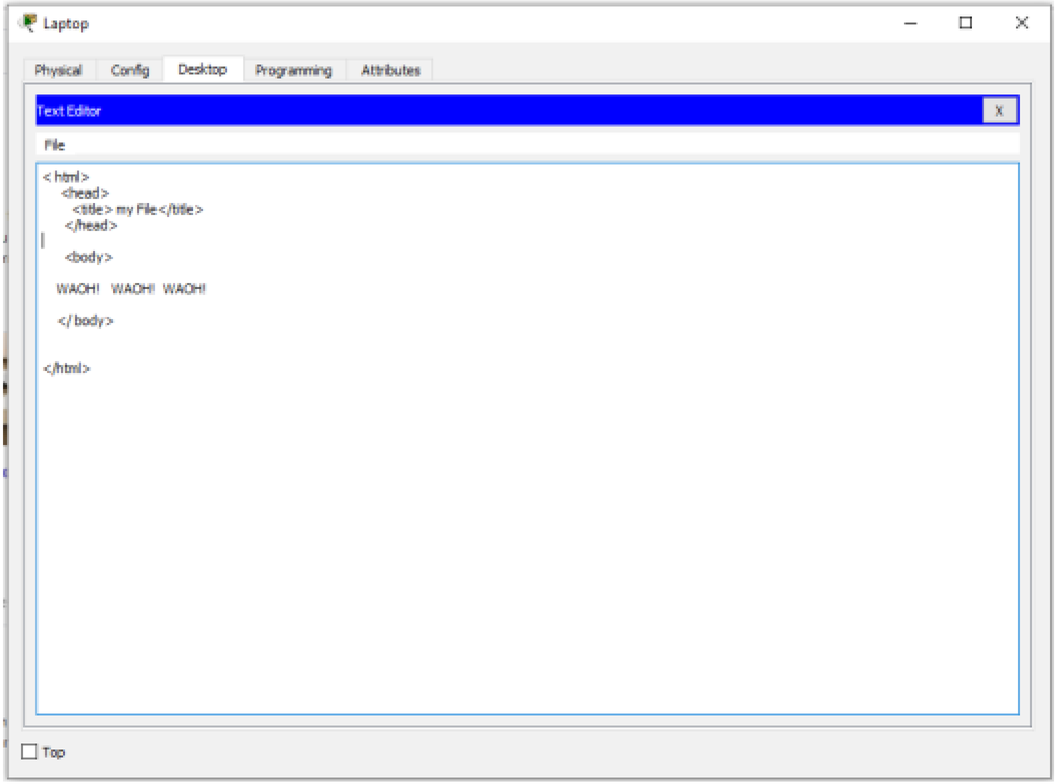
Well done for reading this topic up to this point! You now have more than a foundation regarding working with FTP to upload, download, delete, rename…files.

If you’re okay so far, then let’s do something even more interesting…

We’ll **create** an html file in our Laptop, **upload** it to HTTP server directory using FTP, then try to **access** the file from the Laptop’s browser.

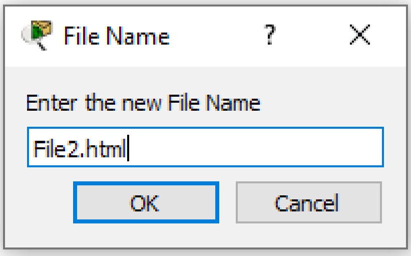
So psych up and let’s move on!

On the Laptop, open the **text editor**, then type some markup(html) and save the file with the extension *.html*. See all this below:



```
< html>
   <head>
     <title> my File</title>
    </head>

    <body>

  WAOH!  WAOH!  WAOH!

  </ body>


</html>
```

Save your file as an html file like this:

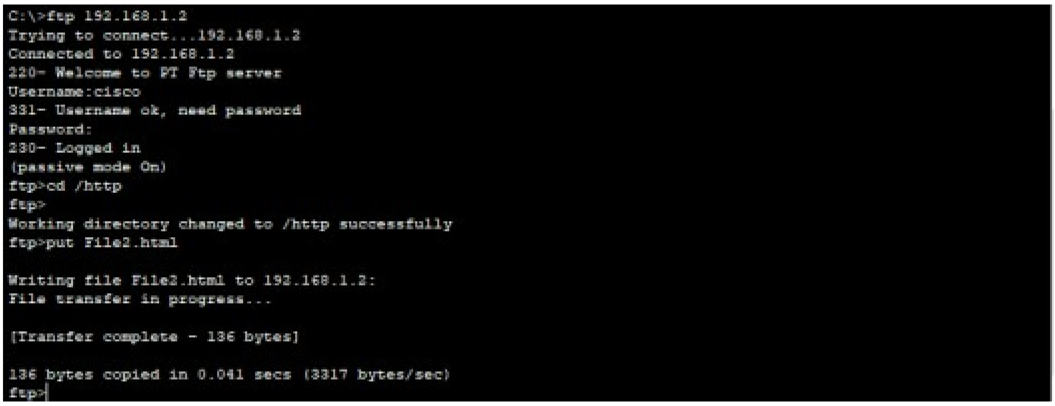


Enter the new File Name

File2.html

Now upload the file(File2.html) to the HTTP server using FTP. This is easy. We’ve already done it previously!

If you’re already in the HTTP directory, you just need to type: put File2.html. If no, first ftp the server(ftp 192.168.1.2), provide the login username(cisco) and password(cisco); change the current directory to HTTP(cd /http) , and finally upload the html file onto the HTTP directory(put File2.html)

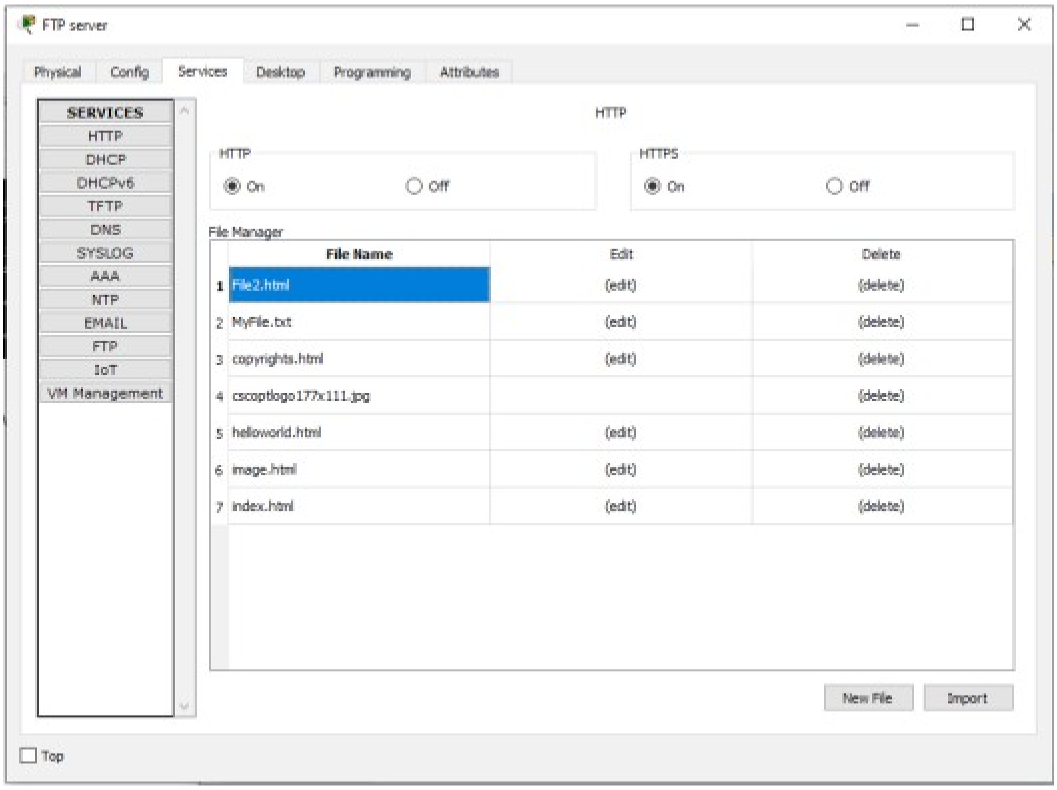


```
C:\>ftp 192.168.1.2
Trying to connect...192.168.1.2
Connected to 192.168.1.2
220- Welcome to PT Ftp server
Username:cisco
331- Username ok, need password
Password:
230- Logged in
(passive mode On)
ftp>cd /http
ftp>
Working directory changed to /http successfully
ftp>put File2.html

Writing file File2.html to 192.168.1.2:
File transfer in progress...

[Transfer complete - 136 bytes]

136 bytes copied in 0.041 secs (3317 bytes/sec)
ftp>
```

Moving on…

Check whether the html file uploaded has been received in the HTTP directory:

Go to **Server->Services-> HTTP**. Then look up for the file in the File Manager.



HTTP

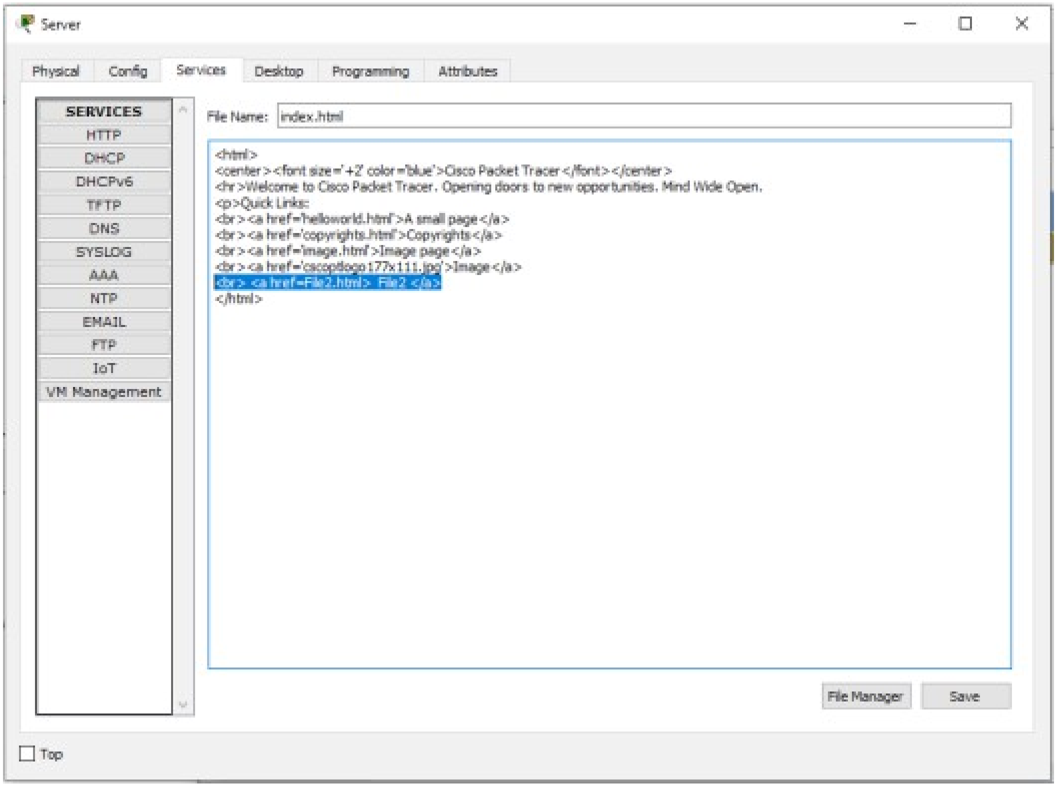
HTTP: ● On ○ Off

HTTPS: ● On ○ Off

File Manager

	File Name	Edit	Delete
1	File2.html	(edit)	(delete)
2	MyFile.txt	(edit)	(delete)
3	copyrights.html	(edit)	(delete)
4	cscoptlogo177x111.jpg		(delete)
5	helloworld.html	(edit)	(delete)
6	image.html	(edit)	(delete)
7	index.html	(edit)	(delete)

Now edit index.html file in the **HTTP directory** so as to include a link to File2 that we’ve just uploaded. This will make File2 accessible from the Laptop’s browser. To do this, locate index.html then click **edit**. Proceed to edit it as shown below. Then **save** and accept overwrite.

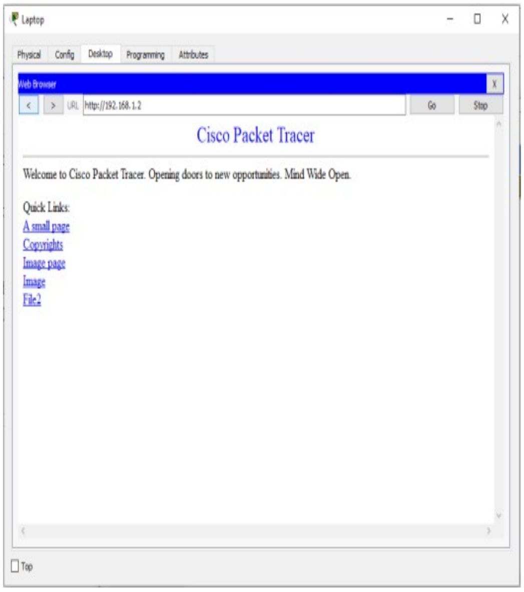


File Name: index.html

```
<html>
<center><font size='+2' color='blue'>Cisco Packet Tracer</font></center>
<hr>Welcome to Cisco Packet Tracer. Opening doors to new opportunities. Mind Wide Open.
<p>Quick Links:
<br><a href='helloworld.html'>A small page</a>
<br><a href='copyrights.html'>Copyrights</a>
<br><a href='image.html'>Image page</a>
<br><a href='cscoptlogo177x111.jpg'>Image</a>
<br> <a href=File2.html> File2 </a>
</html>
```

Finally, try to access the newly uploaded file from the Laptop's browser.

So go to the Laptop's browser and access the server using the server's IP address. By doing this, the browser is making an http request to the server. The server will respond to the Laptop with the index.html file containing a **link** to File2 which we've uploaded from the Laptop using FTP.



URL http://192.168.1.2

Cisco Packet Tracer

Welcome to Cisco Packet Tracer. Opening doors to new opportunities. Mind Wide Open.

Quick Links:
A small page
Copyrights
Image page
Image
File2

Click **File2** link to view the contents of the file in the browser.