TRƯỜNG ĐẠI HỌC SƯ PHẠM HÀ NỘI

KHOA CÔNG NGHỆ THÔNG TIN



**BÁO CÁO BÀI TẬP**

**TÌM HIỂU SSH VÀ SỬ DỤNG FILEZILLA UPLOAD CODE LÊN HOST**

**GV : Nguyễn Thị Thùy Liên.**

***Nhóm 1 : Khổng Đình Oai***

***Nguyễn Mạnh Dũng***

***Phan Bảo Thái***

***Trịnh Khánh Linh***

***Bùi Thị Thúy***

1. **SSH**
2. **SSH là gì**

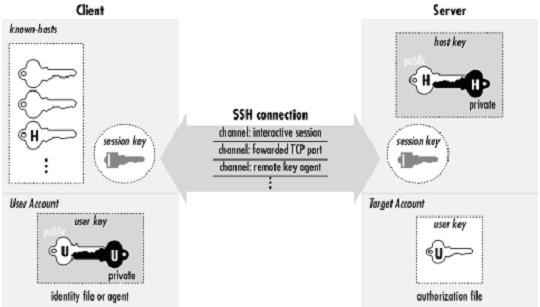
SSH (**Secure Shell**) là một giao thức dùng để thiết lập kết nối mạng một cách bảo mật và an toàn.

* Nó hoạt động theo chồng giao thức TCP/IP
* Có thể thiết lập một đường hầm (tunneling) an toàn
* SSH Client (OpenSSH,...) dùng để giao tiếp giữa client với Server có sử dụng các cơ chế mã hóa để đảm bảo tiêu chí an toàn thông tin thay thế cho các phương pháp cũ về trước như Telnet, rlogin,...

1. **Các đặc điểm của SSH**

* **Tính bí mật** (Privacy) của dữ liệu thông qua việc mã hoá mạnh mẽ
* **Tính toàn vẹn** (integrity) của thông tin truyền, đảm bảo chúng không bị   
  biến đổi.
* **Chứng minh xác thực** (authentication) nghĩa là bằng chứng để nhận dạng   
  bên gửi và bên nhận
* **Giấy phép** (authorization) :dùng để điều khiển truy cập đến tài khoản.
* **Chuyển tiếp** (forwarding) hoặc tạo đường hầm (tunneling) để mã hoá   
  những phiên khác dựa trên giao thức TCP/IP

1. **Cấu Trúc của SSH**

[](https://camo.githubusercontent.com/9f7873c8cc766c8650a32912022d0bddbbbc5429/687474703a2f2f646174612e73696e687669656e69742e6e65742f696d6773382f53696e685669656e49542e4e65742d2d2d535348312e4a5047)

**Server**

* Một chương trình cho phép đi vào kết nối SSH với một bộ máy, trình bày xác thực, cấp phép, … Trong hầu hết SSH bổ sung của Unix thì server thường là sshd.

**Client**

* Một chương trình kết nối đến SSH server và đưa ra yêu cầu như là “log me in” hoặc “copy this file”. Trong SSH1, SSH2 và OpenSSH, client chủ yếu là ssh và scp.

**Session**

* Một phiên kết nối giữa một client và một server. Nó bắt đầu sau khi client xác thực thành công đến một server và kết thúc khi kết nối chấm dứt. Session có thể được tương tác với nhau hoặc có thể là một chuyến riêng.

**Key**

* Một lượng dữ liệu tương đối nhỏ, thông thường từ mười đến một hoặc hai ngàn bit. Tính hữu ích của việc sử dụng thuật toán ràng buộc khoá hoạt động trong vài cách để giữ khoá: trong mã hoá, nó chắc chắn rằng chỉ người nào đó giữ khoá (hoặc một ai có liên quan) có thể giải mã thông điệp, trong xác thực, nó cho phép bạn kiểm tra trễ rằng người giữ khoá thực sự đã kí hiệu vào thông điệp.
* Có hai loại khóa: khoá đối xứng hoặc khoá bí mật và khoá bất đối xứng hoặc khóa công khai. Một khoá bất đối xứng hoặc khoá công khai có hai phần: thành phần công khai và thàn phần bí mật.

|  |  |
| --- | --- |
| Loại key | Mô tả |
| User key | Là một thực thể tồn tại lâu dài, là khoá bất đối xứng sử dụng bởi client như một sự chứng minh nhận dạng của user (một người dùng đơn lẻ có thể có nhiều khoá) |
| Host key | Là một thực thể tồn tại lâu dài, là khoá bất đối xứng sử dụng bới server như sự chứng minh nhận dạng của nó, cũng như được dùng bởi client khi chứng minh nhận dạng host của nó như một phần xác thực đáng tin. Nếu một bộ máy chạy một SSH server đơn, host key cũng là cái duy nhất để nhận dạng bộ máy đó. Nếu bộ máy chạy nhiều SSH server, mỗi cái có thể có một host key khác nhau hoặc có thể dùng chung. Chúng thường bị lộn với server key. |
| Session key | Là một giá trị phát sinh ngẫu nhiên, là khoá đối xứng cho việc mã hoá truyền thông giữa một SSH client và SSH server. Nó được chia ra làm 2 thành phần cho client và server trong một loại bảo bật trong suốt quá trình thiết lập kết nối SSH để kẻ xấu không phát hiện được nó. |
| Key generator | Một chương trình tạo ra những loại khoá lâu dài( user key và host key) cho SSH. SSH1, SSH2 và OpenSSH có chương trình ssh-keygen |
| Known hosts database | Là một chồng host key. Client và server dựa vào cơ sở dữ liệu này để xác thực lẫn nhau. |
| Agent | Một chương trình lưu user key trong bộ nhớ. Agent trả lời cho yêu cầu đối với khoá quan hệ hoạt động như là kí hiệu một giấy xác thực nhưng nó không tự phơi bày khoá của chúng. Nó là một đặc điểm rất có ích. SSH1, SSH2 và OpenSSH có agent ssh-agent và chương trình ssh-add để xếp vào và lấy ra khoá được lưu. |
| Signer | Một chương trình kí hiệu gói chứng thực hostbased. |
| Random seed | Một dãy dữ liệu ngẫu nhiên đựoc dùng bởi các thành phần SSH để khởi chạy phần mềm sinh số ngẫu nhiên |
| Configuration file | Một chồng thiết lập để biến đổi hành vi của một SSH client hoặc SSH server. Không phải tất cả thành phần đều được đòi hỏi trong một bản bổ sung của SSH. Dĩ nhiên những server, client và khoá là bắt buộc nhưng nhiều bản bổ sung không có agent và thậm chí vài bản không có bộ sinh khoá. |

**4. Cơ chế hoạt động**

**Một phiên làm việc của SSH2 đều phải trải qua 4 bước**

* Thiết lập kết nối ban đầu (SSH-TRANS)
* Tiến hành xác thực (SSH-AUTH)
* Mở phiên kết nối để thực hiện các dịch vụ (SSH-CONN)
* Chạy các ứng dụng SSH (Có thể là SSH-SFTP, SCP)

**SSH-TRANS**: là khối xây dựng cơ bản cung cấp kết nối ban đầu, ghi chép giao thức, xác thực server, mã hóa cơ bản và bảo toàn dữ liệu. Sau khi thiết lập kết nối, client có một kết nối độc lập và bảo mật

Sau đó, client dùng SSH-AUTH để xác thực đến server. SSH-AUTH yêu cầu một phương thức: Public key với thuật toán DSS. Ngoài ra, sử dụng mật khẩu và hostbased

Sau khi xác thực, SSH client yêu cầu SSH-CONN để cung cấp một kênh riêng biệt qua SSH-TRANS

Ngoài ra, còn cung cấp các dịch vụ như Remote Login and Command Execution, agent fowarding, files transfer, TCP port fowarding, X fowarding,...

Cuối cùng, một ứng dụng có thể sử dụng SSH-SFTP hoặc SCP truyền file hoặc thao tác remote từ xa

**5. Một số thuật toán trong SSH**

**Thuật toán Public-keys (Khóa công khai)**

* RSA: là thuật toán mã hóa bất đối xứng, dùng cho mã hóa và chữ ký
* DSA: dùng chữ ký số
* Thuật toán thỏa thuận Diffie-Hellman: cho phép 2 bên lấy được khóa được chia sẻ trên một kênh mở

**Thuật toán Private-keys (Khóa bí mật)**

* AES: là một thuật toán mã hóa khối, chiều dài có thể là 128 đến 256bit.
* DES: là một thuật toán mã hóa bảo mật
* 3DES: cải tiến của DES, tăng độ dài của khóa để đạt độ bảo mật cao hơn
* RC4: Kiểu mã hóa nhanh, nhưng kém bảo mật
* Blowfish: là một thuật toán mã hóa miễn phí, có tốc độ mã hóa nhanh hơn DES, nhưng chậm hơn RC4. Độ dài của key từ 32 đến 448bit.

**Hàm băm (HASH)**

* CRS-32: Băm dữ liệu nhưng không mã hóa. Chỉ sử dụng để kiểm tra tính toàn vẹn của gói tin, tránh thay đổi thông tin trên đường truyền
* MD5: Hàm băm có độ an toàn cao vì được mã hóa dữ liệu, với chiều dài là 128bit.
* SHA-1: Một cải tiến của MD5, với chiều dài là 160bit.

**6. Tham Khảo**

<http://sinhvienit.net/forum/chuyen-de-tim-hieu-ssh.10058.html>

1. **CÔNG CỤ FILEZILLA**
2. **Khái niệm**

FileZilla(tên gọi đầy đủ FileZilla Client) là phần mềm mã nguồn mở hỗ trợ upload file, dữ liệu lên server một cách nhanh chóng thông qua giao thức FTP. Đồng thời, còn giúp người dùng quản lý file, tìm kiếm file từ xa, chỉnh sửa file từ xa, đối chiếu thư mục một cách dễ dàng.

1. **Tính năng của FZ**

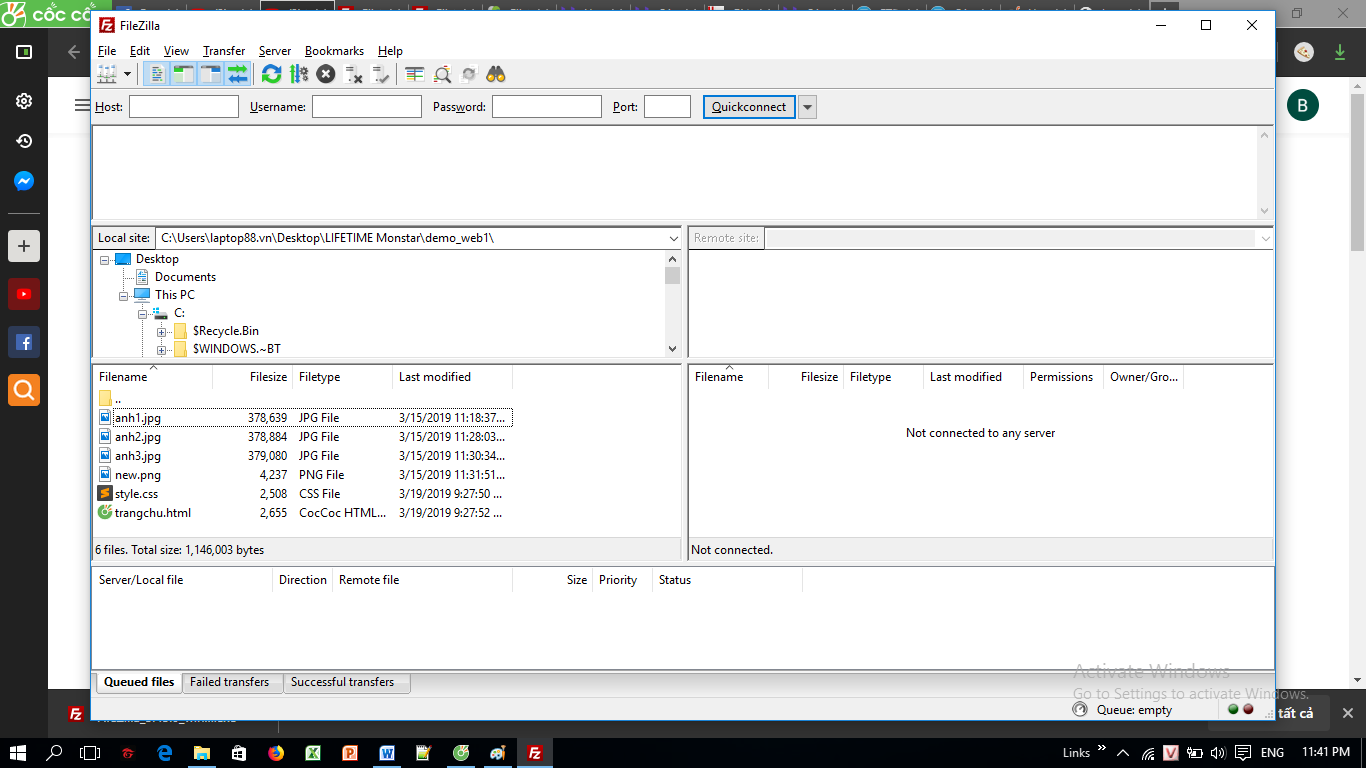
* Dễ sử dụng
* Hỗ trợ các giao thức chuyển đổi file FTP, SSH, SFTP
* Ứng dụng đa nền tảng, chạy trên cả Windows, Linux, BSD, Mac OS X
* Hỗ trợ IPv6
* Có sẵn trong nhiều ngôn ngữ
* Hỗ trợ khôi phục và chuyển đổi các file có dung lượng lớn hơn 4 GB
* Giao diện người dùng đơn giản và trực quan
* Cung cấp các bookmark
* Hỗ trợ tính năng kéo và thả
* Tìm kiếm tập tin từ xa
* Lọc tên file
* Đối chiếu thư mục
* Trình thuật sĩ cấu hình mạng
* Chỉnh sửa file từ xa
* Hỗ trợ HTTP/1.1, SOCKS5 và FTP-Proxy
* Giới hạn tốc độ truyền tải file
* Đồng bộ hóa việc duyệt thư mục

1. **Hướng dẫn sử dụng**
2. **Tải và cài đặt**

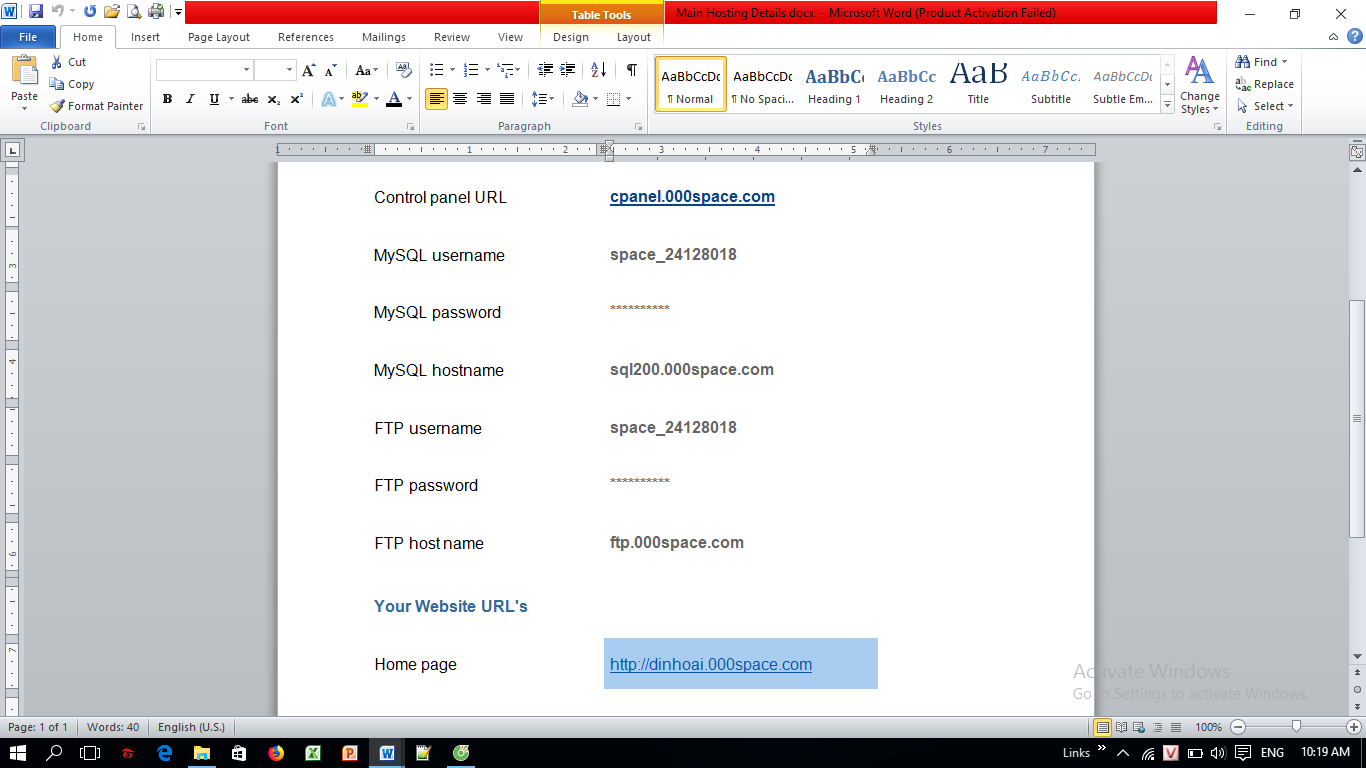
B1: Bạn vào link https://filezilla-project.org/download.php để down các phiên bản mới nhất.

B2: Sau khi down load về bạn tiến hành cài đặt filezilla vào máy tính của bạn.

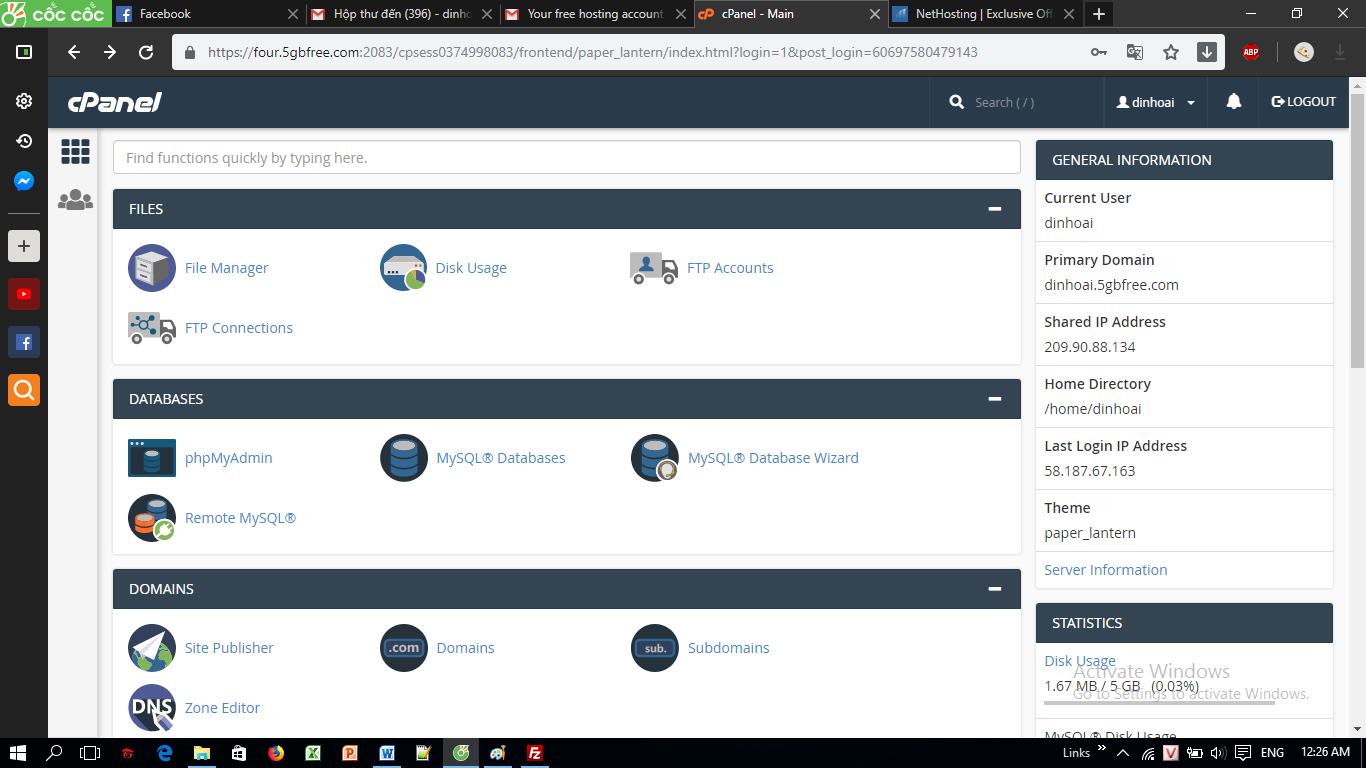
B3: Khởi động filezilla client và sử dụng:



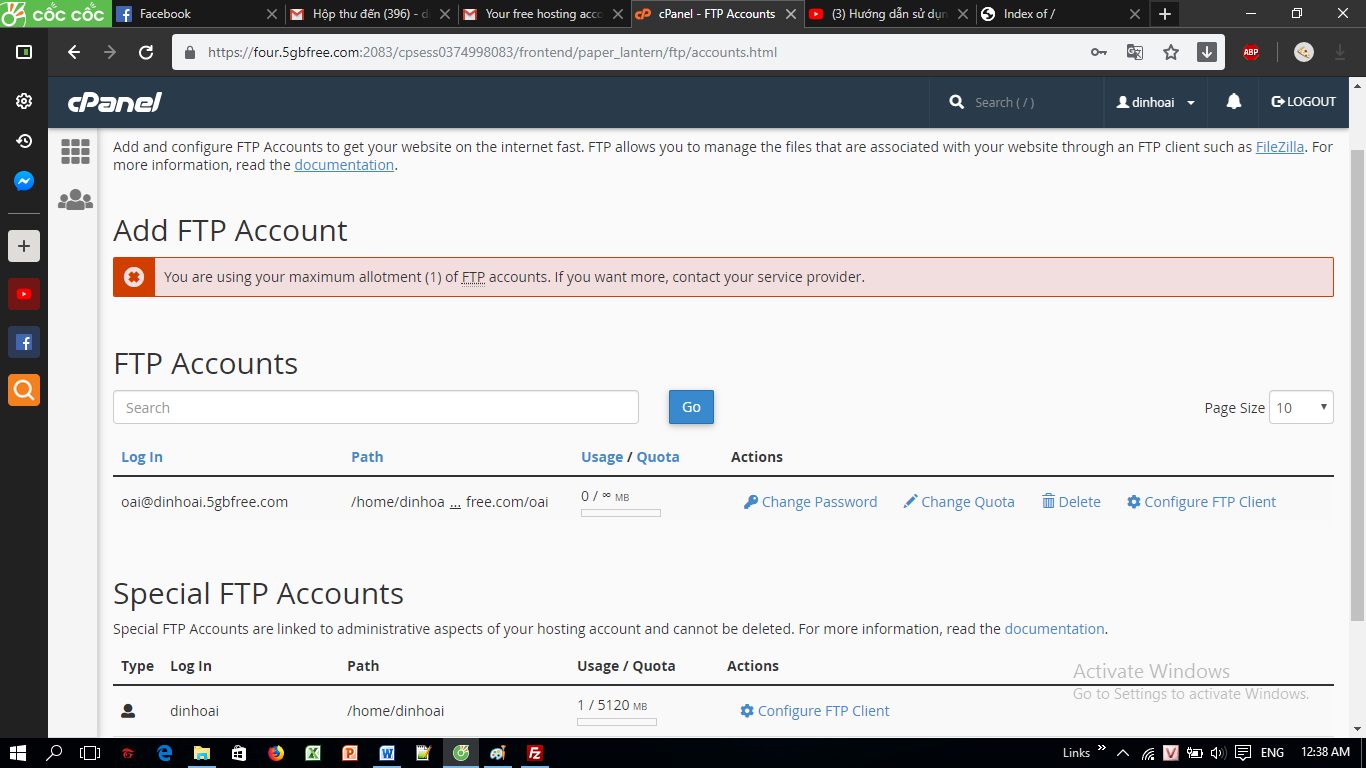
+ Tiếp theo ta cần một tạo một hosting ( Có thể thuê hoặc sử dụng một host free) : truy cập vào [http://.000space.com](http://dinhoai.000space.com/) sau đó click vào Signup For Free Hosting để đăng kí nhận host miễn phí . Sau khi nhập thông tin đăng kí mở mail xác nhận bằng việc click vào đường dẫn mà trang web gửi về ta sẽ được cung cấp một địa chỉ để vào cpanel , một tài khoản ,…



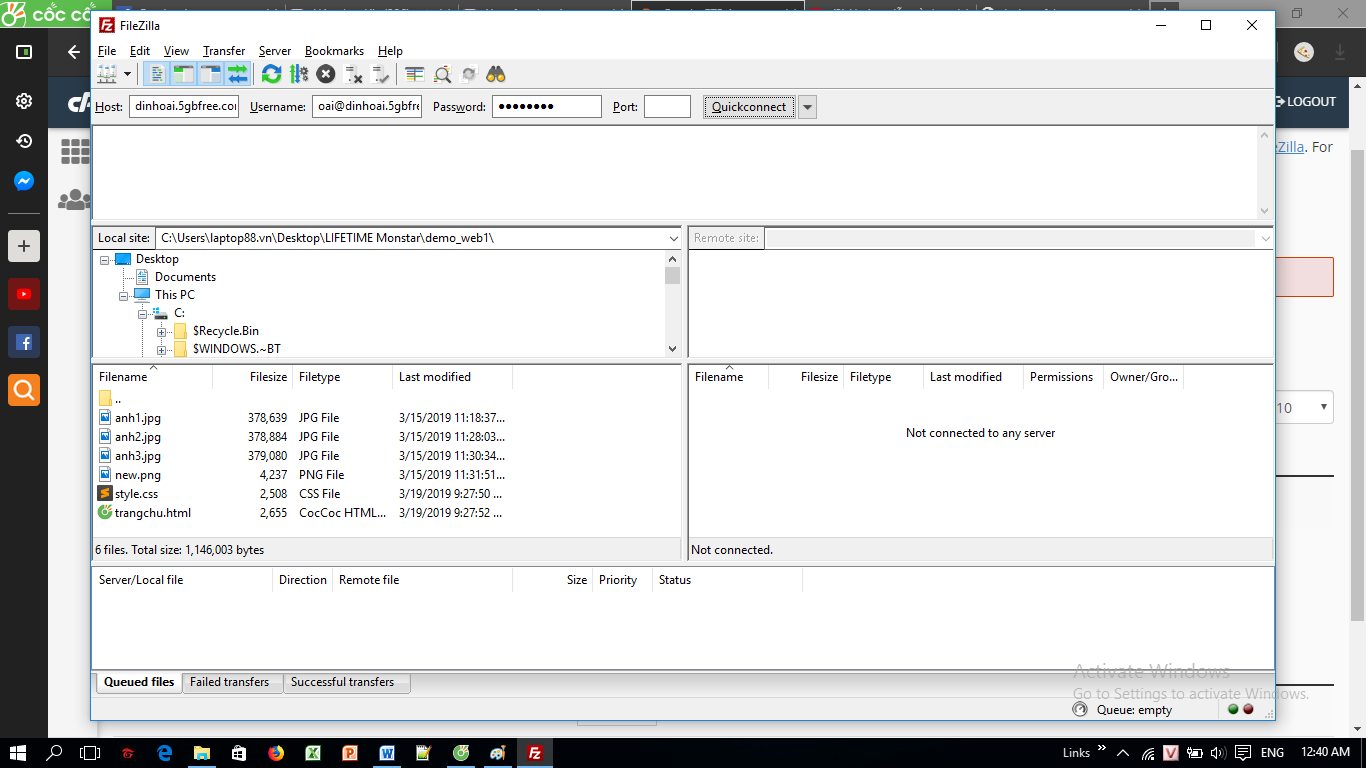
+ Đăng nhập vào Cpanel của địa chỉ nhà cung cấp dịch vụ cấp cho bạn tạo tài khoản FTP Account (trang web vừa tạo host trên cung cấp luôn cho chúng ta FTP user nên ta ko cần tạo).



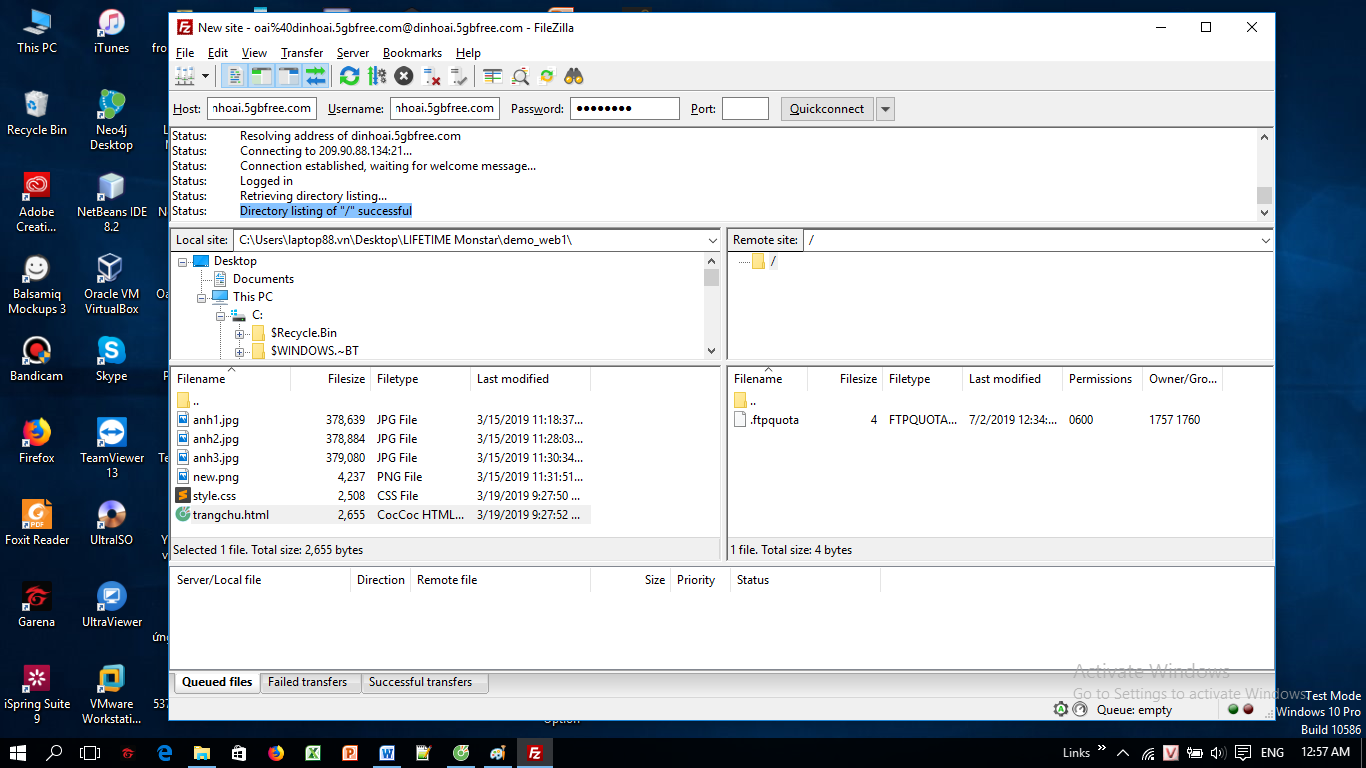
+ Để tạo tài khoản FTP account ta click vào FTP account sau đó điền thông tin cần tạo và click create ta sẽ tạo thành công user như hình sau



+ Tiếp theo ta sử dụng địa chỉ host, user , password tạo bên trên và điền vào FileZilla để kết nối tới host (địa chỉ mặc định của cổng Port là 21 và ta không cần điền) .



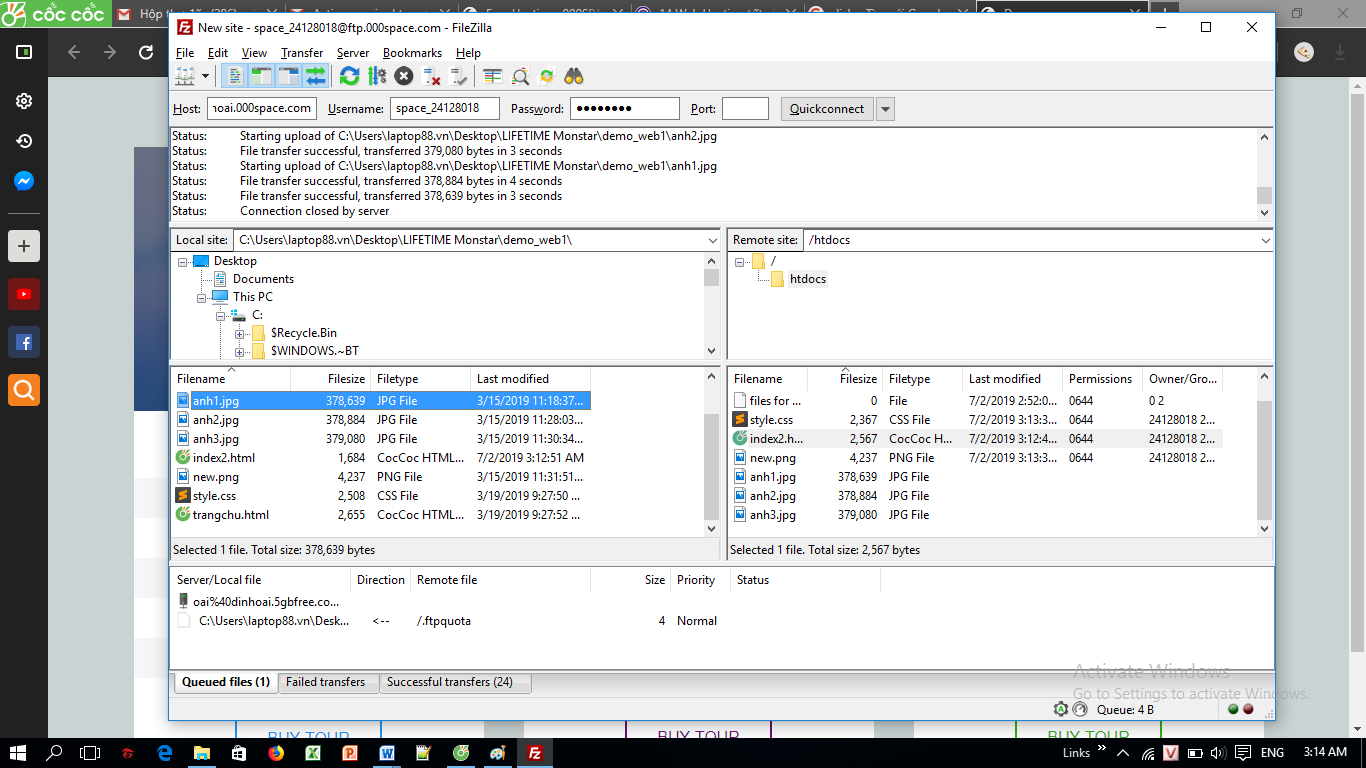
+ Sau khi click Quickconnect : Nếu trạng thái báo thành công ta đã hoàn thành quá trình kết nối FileZilla đến host.(Như hình dưới đây)



+ Nếu không kết nối được theo cách trên ta có thể thực hiện chọn File => Site Manager => New Site => điền đầy đủ thông tin Host , Logon Type : Normal ,

Điền user , password . Lưu ý chọn Encrytion là : Only use plain FTP ( **Đối với các phiên bản FileZilla 3.xx**)

+ Sau khi hoàn thành kết nối ta có thể trực tiếp kéo thả , upload các file , folder code một cách dễ dàng.



Và kết quả sau khi upload code ta được như hình dưới đây :

